A nonlinear model predictive control framework using reference generic terminal ingredients - extended version

Johannes Köhler[1], Matthias A. Müller[2], Frank Allgöwer[1]

Abstract—In this paper, we present a quasi infinite horizon nonlinear model predictive control (MPC) scheme for tracking of generic reference trajectories. This scheme is applicable to nonlinear systems, which are locally incrementally stabilizable. For such systems, we provide a reference generic offline procedure to compute an incrementally stabilizing feedback with a continuously parameterized quadratic quasi infinite horizon terminal cost. As a result we get a nonlinear reference tracking MPC scheme with a valid terminal cost for general reachable reference trajectories without increasing the online computational complexity. As a corollary, the terminal cost can also be used to design nonlinear MPC schemes that reliably operate under online changing conditions, including unreachable reference signals. The practicality of this approach is demonstrated with a benchmark example.

This paper is an extended version of the accepted paper [1], and contains additional details regarding *robust* trajectory tracking (App. B), continuous-time dynamics (App. C), output tracking stage costs (App. D) and the connection to incremental system properties (App. A).

Index Terms—Nonlinear model predictive control, Constrained control, Reference tracking, Incremental Stability

I. INTRODUCTION

Model Predictive Control (MPC) [2] is a well established control method, that computes the control input by repeatedly solving an optimization problem online. The main advantages of MPC are the ability to cope with general nonlinear dynamics, hard state and input constraints, and the inclusion of performance criteria. In MPC (theory), recursive feasibility and closed-loop stability of a desirable setpoint are usually ensured by including suitable terminal ingredients (terminal set and terminal cost) in the optimization problem [3].

In many applications, the control goal goes beyond the stabilization of a pre-determined setpoint. These practical challenges include tracking of changing reference setpoints, stabilization of dynamic trajectories, output regulation and general economic optimal operation. There exist many promising ideas to tackle these issues in MPC, for example by simultaneously optimizing an artificial reference [4], [5], [6], [7],

[1]Johannes Köhler and Frank Allgöwer are with the Institute for Systems Theory and Automatic Control, University of Stuttgart, 70550 Stuttgart, Germany. (email:{johannes.koehler, frank.allgower}@ist.uni-stuttgart.de).

[2]Matthias A. Müller is with the Institute of Automatic Control, Leibniz University Hannover, 30167 Hannover, Germany. (email:mueller@irt.uni-hannover.de).

[8], [9], [10]. However, most of these approaches are limited in some form to linear systems and/or setpoint stabilization. The computation of suitable terminal ingredients seems to be a bottleneck for the practical extension of these methods to nonlinear systems and dynamic trajectories. We bridge this gap, by providing a reference *generic* offline computation for the terminal ingredients. Thus, we can provide practical schemes for nonlinear systems subject to changing operating conditions.

Related work

For linear stabilizable systems, a terminal set and terminal cost can be computed based on the linear quadratic regulator (LQR) and the maximal output admissible set [11]. For the purposes of stabilizing a given setpoint, a suitable design procedure for nonlinear systems with a stabilizable linearization has been provided in [12], [2].

In practice, the setpoint to be stabilized can change and thus procedures independent of the setpoint are necessary. In [13], the issue of finding a setpoint independent terminal cost has been investigated based on the concept of pseudo linearizations. While in principle very appealing, the computation of such a pseudo linearization for general nonlinear systems seems unpractical. In [14], a locally stabilizing controller is assumed and the terminal cost and constraints are defined implicitly based on the infinite horizon tail cost. The main drawback of this method is the implicit description of the terminal cost, which can significantly increase the online computational demand. In [6] the feasible setpoints are partitioned into disjoint sets and for each such set a fixed stabilizing controller and terminal cost are designed using the methods in [15], [16] based on a local linear time-varying (LTV) system description. This method is mainly limited to systems with a one dimensional steady-state manifold, due to the otherwise complex and difficult partitioning. In addition, the piece-wise definition can also lead to numerical difficulties since the terminal cost is not differentiable with respect to the setpoint.

There are many applications in which we want to stabilize some dynamic trajectory or periodic orbit. The nonlinear system along this trajectory can be locally approximated with an LTV system. In [17], this is used to compute a (time-varying) terminal cost for asymptotically constant trajectories. In [18] periodic trajectories are considered and a (periodic) terminal cost is computed based on linear matrix inequalities

(LMIs). A significant practical restriction for these methods is the fact that the offline computation is accomplished for a *specific* (a priori known) trajectory.

In general, the existing procedures to compute terminal ingredients for MPC are mainly focused on computing a terminal cost for a *specific* reference point or reference trajectory. Thus, online changes in the setpoint or trajectory cannot be handled directly and necessitate repeated offline computations.

Contribution

In this work, we provide a reference *generic* offline procedure to compute a parameterized terminal cost. This procedure is applicable to both setpoint or trajectory stabilization. The feasibility of this approach requires local incremental stabilizability of the nonlinear dynamics. The existing design procedures [12], [17], [18] use the linearization around the considered setpoint or trajectory to locally establish properties of the nonlinear systems. In a similar spirit, we consider the linearization of the nonlinear system dynamics around all possible points in the constraint set and describe the dynamics analogous to quasi-linear parameter-varying (LPV) systems. With this description, we formulate the desired properties on the linearized dynamics and provide suitable LMIs to compute the parameter dependent terminal cost and controller. In closed-loop operation we have a quadratic terminal cost with an ellipsoidal terminal constraint directly available. This provides a generalization of the offline computations in [12], [17], [18] to *generic* references. We employ the proposed method in an evasive maneuver test for a car and show that the design of suitable reference generic terminal ingredients can significantly improve the control performance compared to MPC schemes with terminal equality constraints or without terminal constraints.

Given these terminal ingredients, we can extend existing tracking MPC schemes, such as [4], [5], [6], [7], [8], [9], [10] to nonlinear system dynamics and optimal periodic operation, which is a fundamental step towards practical nonlinear MPC schemes. In particular, we provide a nonlinear periodic tracking MPC scheme for exogenous output signals as an extension to [4], [5], [6].

Outline

The remainder of this paper is structured as follows: Section II presents the reference tracking MPC scheme based on the proposed parameterized terminal ingredients. Section III provides a constructive procedure to design parametric terminal ingredients independent of the considered reference. Section IV shows how the resulting parameterized terminal ingredients can be used to extend existing MPC schemes for changing operation conditions to nonlinear system dynamics and periodic operation. Section V shows the practicality of this procedure with numerical examples. Section VI concludes the paper. In the appendix, these results are extended to *robust* trajectory tracking (App. B), continuous-time dynamics (App. C), and output tracking stage costs (App. D). In addition, the connection between the generic terminal ingredients and incremental system properties is discussed (App. A).

II. REFERENCE TRACKING MODEL PREDICTIVE CONTROL

A. Notation

The quadratic norm with respect to a positive definite matrix $Q = Q^\top$ is denoted by $\|x\|_Q^2 = x^\top Q x$. The minimal and maximal eigenvalue of a symmetric matrix $Q = Q^\top$ is denoted by $\lambda_{\min}(Q)$ and $\lambda_{\max}(Q)$, respectively. The identity matrix is $I_n \in \mathbb{R}^{n \times n}$. The interior of a set $\mathcal{X}$ is denoted by $\mathrm{int}(\mathcal{X})$. The vertices of a polytopic set Θ are denoted by $\theta_i \in \mathrm{Vert}(\Theta)$.

B. Setup

We consider the following nonlinear discrete-time system

$$x(t+1) = f(x(t), u(t)) \tag{1}$$

with the state $x \in \mathbb{R}^n$, control input $u \in \mathbb{R}^m$, and time step $t \in \mathbb{N}$. The extension of the following derivation to continuous-time dynamics is detailed in Appendix C. We impose pointwise in time constraints on the state and input

$$(x(t), u(t)) \in \mathcal{Z}, \tag{2}$$

with some compact[1] set $\mathcal{Z}$. We consider the following assumption regarding the reference signal $r = (x_r, u_r) \in \mathbb{R}^{n+m}$.

Assumption 1. *The reference signal r satisfies $r(t) \in \mathcal{Z}_r$, $\forall t \geq 0$, with some set $\mathcal{Z}_r \subseteq int(\mathcal{Z})$. Furthermore, the evolution of the reference signal is restricted by $r(t+1) \in \mathcal{R}(r(t))$, with $\mathcal{R}(r) = \{(x_r^+, u_r^+) \in \mathcal{Z}_r | \ x_r^+ = f(x_r, u_r)\}$.*

This assumption characterizes that the reference trajectory r is reachable, i.e., follows the dynamics f and lies (strictly) in the constraint set $\mathcal{Z}$. If the reference trajectory is not reachable it is possible to enforce these constraints on an artificial reference trajectory which can be included in the MPC optimization problem, compare Section IV.

Remark 1. *The set $\mathcal{R}(r)$ can be modified to incorporate additional incremental input constraints $\|u_r(t+1) - u_r(t)\|_\infty \leq \epsilon$. Setpoints are included as a special case, with $\mathcal{R}(r) = r$ and the steady-state manifold $\mathcal{Z}_r$.*

C. Terminal cost and terminal set

Denote the tracking error by $e_r(t) = x(t) - x_r(t)$. The control goal is to stabilize the tracking error $e_r(t) = 0$ and achieve constraint satisfaction $(x(t), u(t)) \in \mathcal{Z}$, $\forall t \geq 0$. To this end we define the quadratic reference tracking stage cost

$$\ell(x, u, r) = \|x - x_r\|_Q^2 + \|u - u_r\|_R^2, \tag{3}$$

with positive definite weighting matrices Q, R.

Remark 2. *The extension to an tracking stage cost $\ell(x, u, r) = \|h(x, u) - h(x_r, u_r)\|_{S(r)}^2$ with some output $y = h(x, u)$ and a positive definite weighting matrix S is discussed in the Appendix D.*

As discussed in the introduction, we need suitable terminal ingredients to ensure stability and recursive feasibility for the closed-loop system.

---

[1] The derivations can be extended to time-varying constraint sets $\mathcal{Z}(t)$ and dynamics $f(x, u, t)$. The consideration of non-compact constraint sets may require additional uniformity conditions on the nonlinear dynamics.

Assumption 2. *There exist matrices $K_f(r) \in \mathbb{R}^{m \times n}$, $P_f(r) \in \mathbb{R}^{n \times n}$ with $c_l I_n \leq P_f(r) \leq c_u I_n$, a terminal set $\mathcal{X}_f(r) = \{x \in \mathbb{R}^n | \ V_f(x,r) \leq \alpha\}$ with the terminal cost $V_f(x,r) = \|x - x_r\|_{P_f(r)}^2$, such that the following properties hold for any $r \in \mathcal{Z}_r$, any $x \in \mathcal{X}_f(r)$ and any $r^+ \in \mathcal{R}(r)$*

$$V_f(x^+, r^+) \leq V_f(x,r) - \ell(x, k_f(x,r), r), \qquad (4a)$$

$$(x, k_f(x,r)) \in \mathcal{Z}, \qquad (4b)$$

with $x^+ = f(x, k_f(x,r))$, $k_f(x,r) = u_r + K_f(r) \cdot (x - x_r)$ and positive constants c_l, c_u, α.

For $r = r^+ = 0$ this reduces to the standard conditions in [12]. For a given trajectory r, this implies time-varying terminal ingredients, compare [17], [18]. Designing suitable[2] terminal ingredients that satisfy this assumption is the main contribution of this paper and is discussed in more detail in the Section III.

Remark 3. *Assumption 1 implies that the reference $r(t)$ is contained within a control invariant subset $\mathcal{Z}_\infty \subseteq \mathcal{Z}_r$. Thus, Assumption 2 could be relaxed, such that the conditions (4) only need to be satisfied for points $r \in \mathcal{Z}_\infty$. The exact characterization of the set $\mathcal{Z}_\infty$ is, however, challenging and thus we consider the stricter[3] conditions as formulated in Assumption 2.*

D. Preliminary results

Denote the reference r over the prediction horizon N by $r(\cdot|t) \in \mathbb{R}^{(n+m) \times (N+1)}$ with $r(k|t) = r(t+k)$, $k = 0, \ldots, N$. Given a predicted state and input sequence $x(\cdot|t) \in \mathbb{R}^{n \times N+1}$, $u(\cdot|t) \in \mathbb{R}^{m \times N}$ the tracking cost with respect to the reference $r(\cdot|t)$ is given by

$$J_N(x(\cdot|t), u(\cdot|t), r(\cdot|t)) := \sum_{k=0}^{N-1} \ell(x(k|t), u(k|t), r(k|t)) + V_f(x(N|), r(N|t)).$$

The MPC scheme is based on the following (standard) MPC optimization problem

$$V(x(t), r(\cdot|t)) = \min_{u(\cdot|t)} J_N(x(\cdot|t), u(\cdot|t), r(\cdot|t)) \qquad (5a)$$

$$\text{s.t. } x(k+1|t) = f(x(k|t), u(k|t)), \qquad (5b)$$

$$x(0|t) = x(t), \qquad (5c)$$

$$(x(k|t), u(k|t)) \in \mathcal{Z}, \qquad (5d)$$

$$x(N|t) \in \mathcal{X}_f(r(N|t)). \qquad (5e)$$

The solution to this optimization problem are the value function V and the optimal input trajectory $u^*(\cdot|t)$. In closed-

loop operation we apply the first part of the optimized input trajectory to the system, leading to the following closed loop

$$x(t+1) = f(x(t), u^*(0|t)) = x^*(1|t), \quad t \geq 0. \qquad (6)$$

The following theorem summarizes the standard theoretical properties of the closed-loop system (6).

Theorem 1. *Let Assumptions 1 and 2 hold. Assume that Problem (5) is feasible at $t = 0$. Then Problem (5) is recursively feasible and the tracking error $e_r = 0$ is (uniformly) exponentially stable for the resulting closed-loop system (6).*

Proof. This theorem is a straight forward extension of standard MPC results in [20], compare also [17]. Given the optimal solution $u^*(\cdot|t)$, the candidate sequence

$$u(k|t+1) = \begin{cases} u^*(k+1|t) & k \leq N-2 \\ k_f(x^*(N|t), r(N|t)) & k = N-1 \end{cases}, \qquad (7)$$

is a feasible solution to (5a) and implies

$$V(x(t+1), r(\cdot|t+1)) \leq V(x(t), r(\cdot|t)) - \ell(x(t), u(t), r(t)). \qquad (8)$$

Compact constraints in combination with the quadratic terminal cost imply

$$\|x(t) - x_r(t)\|_Q^2 \leq V(x(t), r(\cdot|t)) \leq c_v \|x(t) - x_r(t)\|_Q^2,$$

for some $c_v \geq 1$. Uniform exponential stability follows from standard Lyapunov arguments using the value function V. $\square$

This theorem shows that if we can design suitable terminal ingredients (Ass. 2), the closed-loop tracking MPC has all the (standard) desirable properties. In Section IV we discuss how this can be extended to more general tracking problems. This scheme can be easily modified to ensure robust reference tracking using the method in [21], for details see Appendix B and the numerical example in Section V.

Remark 4. *A powerful alternative to the proposed quasi-infinite horizon reference tracking MPC scheme would be a reference tracking MPC scheme without terminal ingredients [22] ($V_f(x,r) = 0$, $\mathcal{X}_f(r) = \mathcal{X}$). If it is possible to design terminal ingredients (Ass. 2), the value function of such an MPC scheme without terminal constraints is locally bounded by $V(x(t), r(\cdot|t)) \leq \gamma \ell(x,u,r)$, with a suitable constant γ, compare [22, Prop. 2]. Thus, an MPC scheme without terminal constraints enjoys similar closed-loop properties to Theorem 1, provided a sufficiently large prediction horizon N is used, compare [22, Thm. 2]. One of the core advantages of including suitably designed terminal ingredients is that we can implement the MPC scheme with a short prediction horizon N. On the other hand, if the reference is not reachable (Ass. 1), MPC schemes without terminal constraints can still be successfully applied [22, Thm. 4], which is in general not the case for MPC schemes with terminal constraints.*

III. REFERENCE GENERIC OFFLINE COMPUTATIONS

This section provides a reference *generic* offline computation to design terminal ingredients for nonlinear reference tracking MPC. In Lemma 1 we provide sufficient conditions

---

[2]In principle, this assumption can always be satisfied with a terminal equality constraint $\mathcal{X}_f(r) = x_r$. However, this can lead to numerical problems, and decrease performance and robustness of the MPC scheme. In addition, tracking schemes such as [4], [6], [19], typically require a non-vanishing terminal set size α to ensure exponential stability, compare Section IV.

[3]If there exists a fixed constant T_0, such that $r(t+k) \in \mathcal{Z}_r$, $\forall k \in [0, T_0]$, implies $r(t) \in \mathcal{Z}_\infty$, then the conditions in Assumption 2 are not stricter. However, if we use a convex overapproximation (Prop. 1) and/or parameterize the matrices P_f, K_f, then this may introduce additional conservatism.

for the terminal ingredients based on properties of the linearization. Then, two approaches based on LMI computations are described to compute the terminal ingredients, based on Lemma 2 and Proposition 1. After that, a procedure to obtain a non conservative terminal set size α is discussed. Finally, the overall offline procedure is summarized in Algorithm 2. For the special case of setpoint tracking, existing methods are discussed in relation to the proposed procedure. In Appendix C and D, these results are extended to continuous-time dynamics and output tracking stage costs, respectively.

A. Sufficient conditions based on the linearization

We denote the Jacobian of f evaluated around an arbitrary point $r \in \mathcal{Z}_r$ by

$$A(r) = \left[\frac{\partial f}{\partial x}\right]\bigg|_{(x,u)=r}, \quad B(r) = \left[\frac{\partial f}{\partial u}\right]\bigg|_{(x,u)=r}. \quad (9)$$

The following lemma establishes local incremental properties of the nonlinear system dynamics based on the linearization.

Lemma 1. *Suppose that f is twice continuously differentiable. Assume that there exists a matrix $K_f(r) \in \mathbb{R}^{m \times n}$ and a positive definite matrix $P_f(r) \in \mathbb{R}^{n \times n}$ continuous in r, such that for any $r \in \mathcal{Z}_r$, $r^+ \in \mathcal{R}(r)$, the following matrix inequality is satisfied*

$$(A(r) + B(r)K_f(r))^\top P_f(r^+)(A(r) + B(r)K_f(r)) \quad (10)$$
$$\leq P_f(r) - (Q + K_f(r)^\top R K_f(r)) - \epsilon I_n$$

with some positive constant ϵ. Then there exists a sufficiently small constant α, such that P_f, K_f satisfy Assumption 2.

Proof. The proof is very much in line with the result for setpoints in [12], [2]. First we show satisfaction of the decrease condition (4a) and then constraint satisfaction (4b).
Part I: Denote $\Delta x := x - x_r$ and $\Delta u := K_f(r)\Delta x$. Using a first order Taylor approximation at $r = (x_r, u_r)$, we get

$$f(x, k_f(x, r)) = f(x_r, u_r) + A(r)\Delta x + B(r)\Delta u + \Phi_r(\Delta x),$$

with the remainder term Φ_r. The terminal cost satisfies

$$V_f(x^+, r^+) = \|f(x,u) - f(x_r, u_r)\|^2_{P_f(r^+)}$$
$$= \|(A(r) + B(r)K_f(r))\Delta x + \Phi_r(\Delta x)\|^2_{P_f(r^+)}$$
$$\leq \|(A(r) + B(r)K_f(r))\Delta x\|^2_{P_f(r^+)} + \|\Phi_r(\Delta x)\|^2_{P_f(r^+)}$$
$$+ 2\|\Phi_r(\Delta x)\|_{P_f(r^+)}\|(A(r) + B(r)K_f(r))\Delta x\|_{P_f(r^+)}$$
$$\overset{(10)}{\leq} V_f(x, r) - \epsilon\|\Delta x\|^2 - \ell(x, k_f(x,r), r) + \|\Phi_r(\Delta x)\|^2_{P_f(r^+)}$$
$$+ 2\|\Phi_r(\Delta x)\|_{P_f(r^+)}\|(A(r) + B(r)K_f(r))\Delta x\|_{P_f(r^+)}. \quad (11)$$

Using the continuity of $P_f(r)$, $K_f(r)$ and the compactness of the constraint set $\mathcal{Z}_r$, there exist finite constants

$$c_u = \max_{r \in \mathcal{Z}_r} \lambda_{\max}(P_f(r)), \quad c_l = \min_{r \in \mathcal{Z}_r} \lambda_{\min}(P_f(r)), \quad (12)$$

$$k_u = \max_{r \in \mathcal{Z}_r} \|K_f(r)\|, \quad (13)$$

$$c_{u,2} = \max_{r \in \mathcal{Z}_r} \lambda_{\max}(P_f(r) - (\epsilon I + Q + K_f(r)^\top R K_f(r))).$$

Suppose that the remainder term Φ_r is locally Lipschitz[4] continuous in the terminal set with a constant $L_{\Phi,\alpha}$ satisfying

$$\|\Phi_r(\Delta x)\| \leq L_{\Phi,\alpha}\|\Delta x\|,$$
$$L_{\Phi,\alpha} \leq L_\Phi^* := \sqrt{\frac{c_{u,2} + \epsilon}{c_u}} - \sqrt{\frac{c_{u,2}}{c_u}}. \quad (14)$$

Then we have

$$\|\Phi_r(\Delta x)\|^2_{P_f(r^+)}$$
$$+ 2\|\Phi_r(\Delta x)\|_{P_f(r^+)}\|(A(r) + B(r)K_f(r))\Delta x\|_{P_f(r^+)}$$
$$\overset{(10)(12)(14)}{\leq} \left(L^2_{\Phi,\alpha}c_u + 2L_{\Phi,\alpha}\sqrt{c_u}\sqrt{c_{u,2}}\right)\|\Delta x\|^2$$
$$= \left(c_u\left(L_{\Phi,\alpha} + \sqrt{\frac{c_{u,2}}{c_u}}\right)^2 - c_{u,2}\right)\|\Delta x\|^2 \overset{(14)}{\leq} \epsilon\|\Delta x\|^2,$$

which in combination with (11) implies the desired inequality (4a). Twice continuous differentiability of f in combination with compactness of $\mathcal{Z}$ implies that there exists some constant T with

$$\|\Phi_r(\Delta x)\| \leq T\left(\|\Delta x\|^2 + \|\Delta u\|^2\right) \overset{(13)}{\leq} T(1 + k_u^2)\|\Delta x\|^2.$$

Using $\|\Delta x\| \leq \sqrt{\frac{\alpha}{c_l}}$ from the terminal constraint, we get (14) for all $\alpha \leq \alpha_1$ with

$$\alpha_1 := c_l\left(\frac{L_\Phi^*}{T(1 + k_u^2)}\right)^2. \quad (15)$$

Part II: Constraint satisfaction: The terminal constraint $\|\Delta x\|^2_{P_f(r)} \leq \alpha$ in combination with (12), (13) implies

$$(\Delta x, \Delta u) \in \mathcal{B}(\alpha) = \left\{z \in \mathbb{R}^{n+m} \mid \|z\|^2 \leq \frac{\alpha}{c_l}\left(1 + k_u^2\right)\right\}.$$

Given $\mathcal{Z}_r \subseteq \text{Int}(\mathcal{Z})$, there exists a small enough α_2 such that

$$(x,u) = r + (\Delta x, \Delta u) \subseteq \mathcal{Z}_r \oplus \mathcal{B}(\alpha) \subseteq \mathcal{Z}, \; \forall \alpha \leq \alpha_2. \quad (16)$$
$$\square$$

As a summary, given matrices P_f, K_f satisfying (10), we can compute a local Lipschitz bound (14), which in turn implies a maximal terminal set size α_1. Similarly, the constraint sets $\mathcal{Z}$ and $\mathcal{Z}_r$ in combination with K_f, P_f imply an upper bound α_2 to ensure constraint satisfaction. Then Assumption 2 is satisfied for any $\alpha \leq \min\{\alpha_1, \alpha_2\}$. This result is an extension of [12], [2] to arbitrary dynamic references.

B. Quasi-LPV based procedure

Lemma 1 states that matrices satisfying inequality (10) also satisfy Assumption 2 with a suitable terminal set size α. In the following, we formulate computationally tractable optimization problems to compute matrices that satisfy the conditions in Lemma 1. The following Lemma transforms the conditions in (10) to be linear in the arguments.

---

[4] In line with existing procedures [12], we first deriving a sufficient local Lipschitz bound L_Φ^* and then obtain a local region α_1 (15). Alternatively, it is possible to directly use the quadratic bound $\|\Phi_r(\Delta x)\| \leq c\|\Delta x\|^2$ and work with higher order terms to obtain α_1, compare [22, Prop. 1].

Lemma 2. *Suppose that there exists matrices $X(r)$, $Y(r)$ continuous in r, that satisfy the constraints in (19) for all $r \in \mathcal{Z}_r$, $r^+ \in \mathcal{R}(r)$. Then $P_f = X^{-1}$, $K_f = YP_f$ satisfy (10).*

Proof. The proof is standard, compare [23] and Lemma 6 in the Appendix. $\square$

The optimization problem (19) is convex, linear in X, Y and minimizes the worst-case terminal cost $P_f(r) \leq X_{\min}^{-1}$. So far, the result is only conceptual, since (19) is an infinite programming problem (infinite dimensional optimization variables with infinite dimensional constraints). In particular, we need a finite parameterization of X, Y and the infinite constraints need to be converted into a finite set of sufficient constraints.

Remark 5. *One solution to this problem would be sum-of-squares (SOS) optimization [24]. Assuming A, B are polynomial, consider matrices X, Y polynomial in r (with a specified order d) and ensure that the matrix in (19) is SOS. A similar approach is suggested in [25] to find a control contraction metric (CCM) for continuous-time systems (which is a strongly related problem). This approach is not pursued here since most systems require a polynomial of high order to approximate the nonlinear dynamics and the computational complexity grows exponentially in n^d, thus prohibiting the practical application. The connection between CCM and LPV gain-scheduling design is discussed in [26].*

We approach this problem from the perspective of quasi-LPV systems and gain-scheduling [27]. First, write the Jacobian (9) as

$$A(r) = A_0 + \sum_{j=1}^{p} \theta_j(r)A_j, \; B(r) = B_0 + \sum_{j=1}^{p} \theta_j(r)B_j, \; (17)$$

with some nonlinear (continuously differentiable) parameters $\theta \in \mathbb{R}^p$. This can always be achieved with $p \leq n(n+m)$. We impose the same structure on the optimization variables with

$$X(r) = X_0 + \sum_{j=1}^{p} \theta_j(r)X_j, \; Y(r) = Y_0 + \sum_{j=1}^{p} \theta_j(r)Y_j. \; (18)$$

Remark 6. *For input affine systems of the form $f(x,u) = f_x(x) + Bu$, the Jacobian (17) and correspondingly the parameters θ_i only depend on x_r. Thus, the resulting terminal ingredients are solely parameterized by the state x_r.*

Using the parameterization (17)-(18), (19) contains only a finite number of optimization variables, but still needs to be verified for all $r \in \mathcal{Z}_r$, $r^+ \in \mathcal{R}(r)$. There are two options to deal with this: convexifying the problem or gridding the constraint set.

1) Convexify: In order to convexify (19), we match the constraint sets $\mathcal{Z}_r$, $\mathcal{R}(r)$ on the reference r to polytopic constraint sets Θ, Ω on the parameters θ. The polytopic sets Θ, $\Omega(\theta)$ need to satisfy

$$\begin{aligned} \theta(r) &\in \Theta, \quad \forall r \in \mathcal{Z}_r, \\ \theta(r^+) &\in \Omega(\theta(r)), \quad \forall r^+ \in \mathcal{R}(r). \end{aligned} \quad (21)$$

Computing a set Θ, such that $\theta(r) \in \Theta$ for all $r \in \mathcal{Z}_r$ can be achieved by considering a hyperbox $\Theta = \{\theta \in \mathbb{R}^p | \; \theta_i \in [\underline{\theta}_i, \overline{\theta}_i]\}$. For Ω, a simple approach is $\Omega(\theta) = \{\theta\} \oplus \Omega$, where Ω is a hyperbox that encompasses the maximal change in the parameters θ in one time step, i.e. $\Omega = \{\Delta\theta \in \mathbb{R}^p | \; \Delta\theta_i \in [\underline{v}_i, \overline{v}_i]\}$. We denote the joint polytopic constraint set by

$$(\theta, \theta^+) \in \overline{\Theta} = \{(\theta, \theta^+) \in \Theta \times \Theta | \; \theta^+ \in \{\theta\} \oplus \Omega\}, \quad (22)$$

which consists of 6^p vertices. The following proposition provides a simple convex procedure to compute a terminal cost, by solving a finite number of LMIs.

Proposition 1. *Suppose that there exist matrices X_i, Y_i, Λ_i, $X_{\min}$ that satisfy the constraints in (20). Then the matrices*

$$P_f(r) = X^{-1}(r), \quad K_f(r) = Y(r)P_f(r),$$

satisfy (10), with X, Y according to (18).

Proof. Due to Lemma 2, it suffices to show that $X(r)$, $Y(r)$ satisfy the constraints in (19). Due to the definition of the set $\overline{\Theta}$ (22) and $\Lambda_i \geq 0$, any solution that satisfies the constraints (20b) over all $(\theta, \theta^+) \in \overline{\Theta}$, also satisfies the constraints (19) for all $r \in \mathcal{Z}_r$, $r^+ \in \mathcal{R}(r)$. It remains to show that it suffices to check the inequality on the vertices of the constraint set $\overline{\Theta}$. This last result is a consequence of multi-convexity [28, Corollary 3.2]. In particular, if a function f is multi-concave along the edges of the constraint set $\overline{\Theta}$, then it attains its minimum at a vertex of $\overline{\Theta}$ and thus it suffices to verify (20b) over the vertices of $\overline{\Theta}$. The edges of $\overline{\Theta}$ (22) are characterized by $\{\theta_i, \; \theta_i^+, \; \theta_i^+ - \theta_i\}$, $i = 1, \ldots, p$. A function is multi-concave if the second derivative w.r.t. these directions is negative-semi-definite, compare [28, Corollary 3.4]. Similar to [28, Corollary 3.5], the additional constraint (20d) ensures that the function is multi-concave. Thus, it suffices to verify inequality (20b) on the vertices of the constraint set $\overline{\Theta}$. $\square$

Remark 7. *The result in Proposition 1 remains valid, if the set $\overline{\Theta}$ in (22) is replaced by the set $\overline{\Theta} = \Theta \times (\Theta \oplus \Omega)$. This set has only 4^p vertices and the induced conservatism of this approximation is negligible if Ω is small compared to Θ.*

2) Gridding: A common heuristic to ensure that parameter dependent LMIs such as (19) hold for all (r, r^+) is to consider the constraints on sufficiently many sample points in the constraint set, compare e.g. [28, Sec. 4.2]. Due to continuity, the constraint is typically satisfied on the full constraint set if it holds on a sufficiently fine grid. For this method it is crucial that satisfaction of (4a) is verified by using a fine grid (compare Algorithm 1).

The gridding consists of a grid over all possible state and input combinations (r, r^+), i.e., all considered points satisfy

$$r, \; r^+ \in \mathcal{Z}_r, \quad r^+ \in \mathcal{R}(r), \quad \mathcal{R}(r^+) \neq \emptyset. \quad (23)$$

For the simple structure $\mathcal{R}(r)$ in Assumption 1 this can be achieved by gridding r, computing $x_r^+ = f(x_r, u_r)$, and considering all u_r^+, such that $(x_r^+, u_r^+) \in \mathcal{Z}_r$ and $(f(x_r^+, u_r^+), \tilde{u}_r) \in \mathcal{Z}_r$ with some $\tilde{u}_r$. This approach does not introduce additional conservatism, but is computationally

$$\min_{X(r),Y(r),X_{\min}} \; -\log\det X_{\min} \tag{19a}$$

$$\text{s.t.} \begin{pmatrix} X(r) & X(r)A(r)^\top + Y(r)^\top B(r)^\top & (Q+\epsilon)^{1/2}X(r) & (R^{1/2}Y(r))^\top \\ * & X(r^+) & 0 & 0 \\ * & * & I & 0 \\ * & * & * & I \end{pmatrix} \geq 0, \tag{19b}$$

$$X_{\min} \leq X(r), \tag{19c}$$

$$\forall r \in \mathcal{Z}_r, \; r^+ \in \mathcal{R}(r). \tag{19d}$$

$$\min_{X_i,Y_i,\Lambda_i,X_{\min}} \; -\log\det X_{\min} \tag{20a}$$

$$\text{s.t.} \begin{pmatrix} X(\theta) & X(\theta)A(\theta)^\top + Y(\theta)^\top B(\theta)^\top & (Q+\epsilon)^{1/2}X(\theta) & (R^{1/2}Y(\theta))^\top \\ * & X(\theta^+) & 0 & 0 \\ * & * & I & 0 \\ * & * & * & I \end{pmatrix} - \begin{pmatrix} \sum_{i=1}^p \theta_i^2 \Lambda_i & 0 \\ 0 & 0 \end{pmatrix} \geq 0, \tag{20b}$$

$$X_{\min} \leq X(\theta), \quad \forall (\theta,\theta^+) \in \text{Vert}(\overline{\Theta}), \tag{20c}$$

$$\begin{pmatrix} 0 & (A_iX_i + B_iY_i)^\top \\ (A_iX_i + B_iY_i) & 0 \end{pmatrix} - \Lambda_i \leq 0, \quad \Lambda_i \geq 0, \quad i = 1,\dots,p. \tag{20d}$$

challenging for high dimensional systems. As discussed in Remark 1 we can include additional constraints on the reference, which makes the offline computation less conservative. If some parameters, e.g. u_r, enter the LMIs affinely and are subject to polytopic constraints, it suffices to consider the vertices of the corresponding constraint set.

The advantage of the convex procedure (compared to the gridding) is that it typically scales better with the system dimension. This comes at the cost of additional conservatism due to the construction of the set $\overline{\Theta}$ and the additional multi-convexity constraint (20d). The computational demand can be reduced by considering (block-)diagonal multipliers $\Lambda_i = \lambda_i I$. It can often be beneficial to consider a combination of the two approaches, i.e. grid in some dimensions and conservatively convexify in others. The advantages and applicability of both approaches are explored in more detail in the numerical examples in Section V.

The main result is that we can formulate the offline design procedure similar to the gain scheduling synthesis of (quasi)-LPV systems and thus can draw on a well established field to formulate[5] offline LMI procedures, compare [27].

C. Non-conservative terminal set size α

The terminal set size α derived in Lemma 1 can be quite conservative. In the following we illustrate how a non conservative value α can be computed (given P_f and K_f).

1) Constraint satisfaction - α_2: Assume that we have polytopic constraints of the form $\mathcal{Z} = \{r = (x,u)|L_r r \leq l\}$.

The constant α_2, with the property that $\alpha \leq \alpha_2$ implies constraint satisfaction (4b), can be computed with

$$\alpha_2 := \max_\alpha \; \alpha \tag{24}$$

$$\text{s.t.} \; \|P_f(r)^{-1/2}\begin{pmatrix} I_n & K_f^\top(r) \end{pmatrix}L_{r,j}^\top\|^2 \alpha \leq (l_j - L_{r,j}r)^2,$$

$$\forall r \in \mathcal{Z}_r, \quad j = 1,\dots n_z.$$

This problem can be efficiently solved by girdding the constraint set $\mathcal{Z}_r$, solving the resulting linear program (LP) for each point r and taking the minimum. In the special case that P_f, K_f are constant this reduces to one small scale LP.

2) Local Stability - α_1: Determining a non-conservative constant α_1, related to the local Lyapunov function V_f can be significantly more difficult. For comparison, in the setpoint stabilization case a non-convex optimization problem is formulated to check whether (4a) holds for a specific value of α_1, compare [12, Rk. 3.1]. In a similar fashion, we consider the following algorithm[6] to determine whether (4a) holds for all $\alpha \leq \alpha_1$:

Algorithm 1 Offline computation - Local stability α_1

   Given a candidate constant α_1:
   Grid: Select (r,r^+) satisfying (23)
1: Evaluate $P_f(r), P_f(r^+), K_f(r)$ using (18).
2: Generate random vectors Δx_i: with $\|\Delta x_i\|_{P_f(r)}^2 \leq \alpha_1$.
3: Check if $x_i = x_r + \Delta x_i$ satisfies (4a).

Starting with $\alpha_1 = \alpha_2$, the value α_1 is iteratively decreased until all considered combination (r,r^+,x_i) satisfy (4a).

The overall offline procedure to compute the terminal ingredients (Ass. 2) is summarized as follows:

---

[5]If the parameters θ_i are chosen based on a vertex representation ($\theta_i \geq 0, \sum_{i=1}^p \theta_i = 1$) the multi-convexity condition (20d) can be replaced by positivity conditions of the polynomials, compare for example [29]. In [30] a convexification with an additional matrix is considered. More elaborate methods to formulate LPV synthesis with finite LMIs can be found in [31].

[6]Algorithm 1 can be thought of as a sampling based strategy to solve this non-convex optimization problem considered in [12, Rk. 3.1]. Using standard convex solvers, like sequential quadratic programming (SQP), yield a faster solution, but can get stuck in local minima. This is dangerous for this problem, since the local minima correspond to values α that do not satisfy Assumption 2. Alternatively, nonlinear Lipschitz-like bounds can be used to reduce the conservatism, compare [32] (which, however, also use sampling).

Algorithm 2 Offline computation

1: Define θ corresponding to the linearization (17).
2: LMI computation using gridding or convexification:
   Convex: Determine hyperbox sets Θ, Ω satisfying (21).
      Solve (20) using $\overline{\Theta}$ according to (22) or Remark 7.
   Gridding: Select (r_i, r_i^+) satisfying (23).
      Solve (19) for all $(r_i, \ r_i^+)$.
3: Compute size of the terminal set $\alpha = \min\{\alpha_1, \alpha_2\}$:
   a):compute α_1 using Algorithm 1 (or (15)),
   b):compute α_2 using (24) (or (16)).

The presented offline procedure is considerably more involved than for example the computation for one specific setpoint [12]. We emphasize that this procedure only has to be completed once and we need *no* repeated offline computations to account for changing operation conditions. Furthermore, the applicability to nonlinear systems with the corresponding computational effort offline is detailed with numerical examples in Section V.

D. Setpoint tracking

Now we discuss setpoint tracking, which is included in the previous derivation as a special case with $\mathcal{Z}_r$ such that $(x_r, u_r) \in \mathcal{Z}_r$ implies $x_r = f(x_r, u_r)$ and $\mathcal{R}(r) = r$. Note, that both presented approaches significantly simplify in this case. For the gridding approach it suffices to grid along the steady-state manifold $\mathcal{Z}_r$ which is typically low dimensional. In the convex approach (Prop. 1) we have $\theta^+ = \theta$ and thus we only consider the 2^p vertices of Θ.

Compared to the dynamic reference tracking problem, the problem of tracking a setpoint has received a lot of attention in the literature and many solutions have been suggested.

One of the first attempts to solve this issue is the usage of a pseudo linearization in [13]. There, a nonlinear state and input transformation is sought, such that the linearization of the transformed system around the setpoints is constant and thus constant terminal ingredients can be used. This approach seems unpractical, since there is no easy or simple method to compute such a pseudo linearization.

In [6], [15], [16] the steady-state manifold $\mathcal{Z}_r$ is partitioned into sets. In each set the nonlinear system is described as an LTV system and a constant terminal cost and controller are computed. Correspondingly, in closed-loop operation under changing setpoints [6] the terminal cost matrix P_f is piecewise constant. This might cause numerical problems in the optimization, since the cost is not differentiable with respect to the reference r. Furthermore, the (manual) partitioning of the steady-state manifold seems difficult for general MIMO systems (if the dimension of the steady-state manifold is larger than one). In comparison, Algorithm 2 yields continuously parameterized terminal ingredients, thus avoiding the need for user defined partitioning and piece-wise definitions.

In [9, Remark 8] it was proposed to compute a continuously parameterized controller $K_f(r)$ by analytically using a pole-

placement formula and solving the corresponding Lyapunov[7] equation to obtain $P_f(r)$. The resulting terminal ingredients are quite similar to the proposed ones. However, this procedure cannot be directly translated into a simple optimization problem and might hence not be tractable.

IV. NONLINEAR MPC SUBJECT TO CHANGING OPERATION CONDITIONS

Many control problems are more general than the reference tracking considered in Section II. One challenge includes tracking and output regulation with exogenous signals in order to accommodate online changing operation conditions. For this set of problems, the reference r might not satisfy Assumption 1 (due to sudden changes and unreachable signals), compare [4], [5], [6]. More generally, the minimization of a possibly online changing and non-convex economic cost is a (non-trivial) control problem which is often encountered, compare [7], [8], [9], [10]. One promising method to solve these problems is the simultaneous optimization of an artificial reference, as done in [4], [5], [6], [7], [8], [9], [10]. Compared to a standard reference tracking MPC formulation such as (5), these schemes ensure recursive feasibility despite changes in exogenous signals (such as the desired output reference or the economic cost). In this section, we show how the reference generic terminal ingredients can be used to design nonlinear MPC schemes that reliably operate under changing operating conditions, as an extension and combination of the ideas in [4], [5], [6], [7], [8], [9], [10]. In particular, we present a scheme that exponentially stabilizes the periodic trajectory which best tracks an exogenous output signal. The extension of the economic MPC schemes [7], [8], [9], [10] to periodic artificial trajectories based on the reference generic terminal ingredients is beyond the scope of this work and part of current research.

A. Nonlinear periodic tracking MPC subject to changing exogenous output references

We assume that at time t an exogenous T-periodic output reference signal $y_e(\cdot|t) \in \mathbb{R}^{p \times T}$ is given. For some T-periodic reference $r(\cdot|t) = (x_r(\cdot|t), u_r(\cdot|t)) \in \mathbb{R}^{(n+m) \times T}$, we define the tracking cost with respect to this output signal y_e by

$$J_T(r(\cdot|t), y_e(\cdot|t)) = \sum_{j=0}^{T-1} \| \underbrace{h(r(j|t))}_{=y_r(j|t)} - y_e(j|t)\|^2,$$

with a bounded nonlinear output function $h : \mathcal{Z}_r \to \mathbb{R}^p$. The objective is to stabilize the feasible T-periodic reference trajectory r, that minimizes J_T. In [4], [6] the issue of stabilizing the optimal setpoint for piece-wise constant output signals has been investigated. In [5] periodic trajectories have been considered for the special case of linear systems. By

---

[7]In [9], the terminal cost V_f is computed for a (differentiable) economic stage cost $\ell(x, u)$ (not necessarily quadratic), compare also [7]. The computation of the terminal cost is decomposed into a linear and quadratic term, compare [33]. Computing the quadratic term of this economic terminal cost is equivalent to computing a quadratic terminal cost for a quadratic stage cost (Ass 2).

combining these methods with the proposed terminal ingredients, we can design a nonlinear MPC scheme that stabilizes the optimal periodic[8] trajectory for periodic output reference signals, compare [19]. The scheme is based on the following optimization problem

$$W_T(x(t), y_e(\cdot|t)) \tag{25a}$$
$$= \min_{u(\cdot|t), r(\cdot|t)} J_N(x(\cdot|t), u(\cdot|t), r(\cdot|t)) + J_T(r(\cdot|t), y_e(\cdot|t))$$

$$\text{s.t. } x(k+1|t) = f(x(k|t), u(k|t)), \quad x(0|t) = x(t), \tag{25b}$$
$$(x(k|t), u(k|t)) \in \mathcal{Z}, \quad x(N|t) \in \mathcal{X}_f(r(N|t)), \tag{25c}$$
$$r(j+1|t) \in \mathcal{R}(r(j|t)) \subseteq \mathcal{Z}_r, \tag{25d}$$
$$r(l+T|t) = r(l|t), \; l = 0, \ldots, \max\{0, N-T\}, \tag{25e}$$
$$j = 0, \ldots, T-1, \quad k = 0, \ldots, N-1.$$

This scheme is recursively feasible, independent of the output reference signal y_e. Furthermore, if the exogenous signal y_e is T-periodic the closed-loop system is stable. Additionally, if a convexity and continuity condition on the set of feasible periodic orbits and the output function h is satisfied [19, Ass. 5], then the optimal reachable periodic trajectory is (uniformly) exponentially stable for the resulting closed-loop system. Thus, the terminal ingredients enable us to implement a nonlinear version of the tracking scheme in [4], [5], that ensures exponential stability of the optimal (periodic) operation. More details on the theoretical properties and numerical examples can be found in [19]. Although the consideration of general non-periodic trajectories is still an open issue, we conjecture that the approach can be extended to any class of finitely parameterized reference trajectories.

V. Numerical examples

The following examples show the applicability of the proposed method to nonlinear systems and the closed-loop performance improvement when including suitable terminal ingredients. We first illustrate the basic procedure at the example of a periodic reference tracking task for a continuous stirred-tank reactor (CSTR). Then we demonstrate the advantages of using suitable terminal ingredients with (robust) trajectory tracking and an evasive maneuver test for a car. Additional examples, including tracking of periodic output signals (Sec. IV-A) with a nonlinear ball and plate system can be found in [19].

In the following examples, the offline computation is done with an Intel Core i7 using the semidefinite programming (SDP) solver SeDuMi-1.3 [34] and the online optimization is done with CasADi [35]. The offline computation can be done using both the discrete-time formulation (Sec. III) or the continuous-time formulation (Appendix C). Hence, we also compare the performance of these different formulations.

A. Periodic reference tracking - CSTR

System model: We consider a continuous-time model of a continuous stirred-tank reactor (CSTR)

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{pmatrix} = \begin{pmatrix} 1 - x_1 - 10^4 x_1^2 \exp(\frac{-1}{x_3}) - 400 x_1 \exp(\frac{-0.55}{x_3}) \\ 10^4 x_1^2 \exp(\frac{-1}{x_3}) - x_2 \\ u - x_3 \end{pmatrix},$$

where x_1, x_2, x_3 correspond to the concentration of the reaction, the desired product, waste product and u is related to the heat flux through the cooling jacket, compare [36], [37, Sec. 3.4]. The constraints are

$$\mathcal{Z}_r = [0.05, 0.45] \times [0.05, 0.15] \times [0.05, 0.2] \times [0.059, 0.439],$$
$$\mathcal{Z} = [0, 1]^3 \times [0.049, 0.449].$$

The discrete-time model is defined with explicit Runge-Kutta discretization of order 4 and a sampling time[9] of $h = 0.01$.

For this system, periodic operation is economically beneficial, compare [36]. Thus, we consider the problem of tracking reachable periodic reference trajectories r (Assumption 1), corresponding to the economic operation of the plant.

Offline computations: In the following, we illustrate the reference generic offline computation for this system. We consider the standard quadratic tracking stage cost with $Q = I_3$, $R = 10$ and use $\epsilon = 0.1$.

For the continuous-time system, the Jacobian (9) contains four nonlinear terms, yielding the parameters

$$\theta_1(x) = 400 \exp(-0.55/x_3), \quad \theta_2(x) = 2 \cdot 10^4 x_1 \exp(-1/x_3),$$
$$\theta_3(x) = 10^4 (x_1/x_3)^2 \exp(-1/x_3),$$
$$\theta_4(x) = 400 \cdot 0.55 x_1/(x_3^2) \exp(-0.55/x_3).$$

The input u_r enters the LMIs affinely. Thus, we only consider the two vertices of u_r and grid (x_1, x_3) using 10^2 points.

For the discrete-time system, the explicit description of the nonlinear dynamics f and the corresponding Jacobian $A(r)$, $B(r)$ is complex. Thus, we directly define the non-constant[10] components of the Jacobian A, B as the parameters $\theta \in \mathbb{R}^6$. We compute the hyperbox sets $\Theta, \Omega \subseteq \mathbb{R}^6$ satisfying (21) numerically. For the discrete-time convex approach the polytopic description $\overline{\Theta}$ (22) and the hyperbox description $\Theta \times \Omega$ (Remark 7) are considered. For the gridding, $(x_1, x_3, u_r, u_r^+) \in \mathbb{R}^4$ is gridded using 10^4 points, of which approximately 8.000 satisfy the conditions (23) and are considered in the optimization problem (19).

The computational demand and the performance of the different methods are detailed in Table I. As expected, the gridding approach yields the smallest and least conservative terminal cost. For this example, the convex discrete-time approach (Prop. 1) seems less favorable, which is mainly due to the simple description of the parameters. Due to the small sampling time h and correspondingly small set Ω, the more detailed description $\overline{\Theta}$ only marginally improves the performance but significantly increases the offline computational

---

[8] In the case of setpoint tracking ($T = 1$), the MPC scheme reduces to [6]. As discussed in Section III-D, the proposed procedure can be used to design suitable terminal ingredients for setpoints.

[9] In [37, Sec. 3.4] a sampling time of $h = 0.1$ is used. However, with the considered fourth order explicit Runge-Kutta discretization, a sampling time of $h = 0.1$ does not preserve stability of the continuous-time system.

[10] The derivatives $\partial f_3/\partial r$, $\partial f_1/\partial x_2$, and $\partial f_2/\partial x_2$ are constant.

demand. Furthermore, the continuous-time formulation can be computed more efficiently. We note, that the parameters Q, R, h, are chosen, such that the continuous-time control law is also stabilizing for the discrete-time implementation. In particular, if R is decreased or h increased, the terminal ingredients based on the continuous-time formulation do not satisfy Assumption 2 with a piece wise constant input. Such considerations are not necessary for the discrete-time formulation, compare Remark 14 in Appendix C.

In the following, we consider the discrete-time terminal ingredients based on Lemma 2. Computing $\alpha_2 = 0.02$ using (24) requires 30 s. Executing Algorithm 1 to ensure that $\alpha = 0.02$ is valid takes 10 min using $2 \cdot 20^4 \cdot 100 = 3.2 \cdot 10^7$ samples.

In Figure 1 we can see an exemplary periodic trajectory and the corresponding terminal set[11]. The period length is $T = 1144$, which corresponds to $11.44\ s$, compare [37, Sec. 3.4].
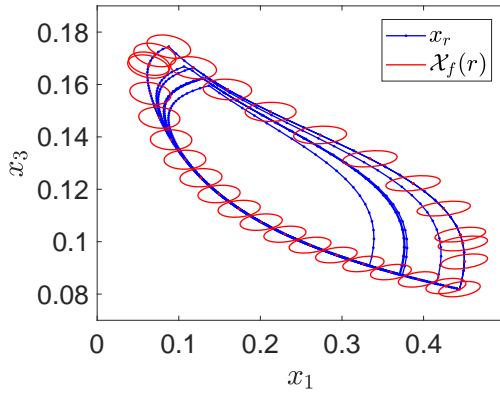

Fig. 1. Periodic trajectory - CSTR: Reference trajectory r (blue) with terminal sets $\mathcal{X}_f(r)$ (red ellipses).

We wish to emphasize that this offline computation is only done once and requires no explicit knowledge of the specific trajectory or its period length T. This is in contrast to the existing methods, such as [17], [18] which would compute terminal ingredients for a specific reference trajectory and thus could not deal with online changing operation conditions (e.g. due to changes in the price signal [10]).

B. Automated driving - robust reference tracking

The following example shows the applicability of the proposed procedure to nonlinear robust reference tracking and demonstrates the performance improvement of including suitable terminal ingredients.

System model: We consider a nonlinear kinematic bicycle model of a car

$$\dot{z}_1 = v\cos(\psi + \beta), \quad \dot{z}_2 = v\sin(\psi + \beta),$$
$$\dot{\psi} = v/l_r \sin(\beta), \quad \dot{v} = a, \quad \dot{\delta} = u_\delta,$$
$$\beta = \tan^{-1}\left(\frac{l_r}{l_f + l_r}\tan(\delta)\right),$$
$$x = [z_1, z_2, \psi, v, \delta]^\top \in \mathbb{R}^5, \quad u = [a, u_\delta]^\top \in \mathbb{R}^2,$$

[11]If α would be recomputed for the specific trajectory r, we would get $\alpha = 0.1$. This conservatism is a result of the fact, that the previously computed value α needs to be valid for *every* reachable reference trajectory (Ass. 1).

with the position z_i, the inertial heading ψ, the velocity v, the front steering angle δ, the acceleration a and the change in the steering angle u_δ. The model constants $l_f = 1.4$ and $l_r = 1.5$ represent the distance of the center of mass to the front and rear axle. More details on kinematic bicycle models can be found in [38]. The (non-compact) constraint sets are given by

$$\mathcal{Z}_r = \{v \in [10, 50], a \in [-1, 1], \delta \in [-0.4, 0.4], u_\delta \in [-3, 3]\},$$
$$\mathcal{Z} = \{v \in [5, 55], a \in [-2, 2], \delta \in [-0.5, 0.5], u_\delta \in [-6, 6]\}.$$

Offline computations: We consider the stage cost $Q = I_5$, $R = I_2$ and $\epsilon = 0.1$ and use an Euler discretization with the step size $h = 2ms$. Computing the linearization (9) and using a quasi-LPV parameterization (17) results in $\theta \in \mathbb{R}^8$, where the parameters θ consist of trigonometric functions in Ψ, δ and are linear in the velocity v.

For this example, the convex approach (Prop. 1) is not feasible, since the simple and conservative hyperbox[12] description $\theta \in \Theta$ includes linearized dynamics which are not stabilizable.

For the gridding, we consider both the discrete-time and a continuous-time formulation (compare Appendix C). In the continuous-time formulation a and u_δ enter the LMIs affinely. Thus, we only consider the $2^2 = 4$ vertices of (a, u_δ) and grid (ψ, v, δ) using 10^3 points. For the discrete-time formulation (19) the LMIs are not affine in u_δ and thus we grid $(\psi, v, \delta, u_\delta)$ using $10^3 \cdot 5$ points and consider the two vertices of a. The dimensions of the corresponding LMI-blocks are $(2n + m) \times (2n + m) = 12 \times 12$ and $(3n + m) \times (3n + m) = 17 \times 17$, respectively. The following table captures the weighting of the terminal cost and the computational effort of the proposed approach.

Method	Continuous-Time (Lemma. 4)	Discrete-time (Lemma. 2)
#LMIs-blocks	$10^3 \cdot 2^2 = 4 \cdot 10^3$	$10^3 \cdot 5 \cdot 2 = 10^4$
comp. time	14 min	33 min
$\max_{r \in \mathcal{Z}_r} \lambda_{\max}(P_f(r))$	$8.0 \cdot 10^4 \cdot h$	$8.4 \cdot 10^4$

Remark 8. *For the considered example and parameters, the continuous-time terminal cost is also valid for a zero-order hold discrete-time implementation with $h = 2\ ms$. This is in general not the case. For example if $R = 10^{-4}$ or $h = 10\ ms$ is chosen, the terminal ingredients based on the continuous-time offline optimization are not stabilizing for the discrete-time system. If the continuous-time offline procedure is used, the computation (and thus verification) of α for the discrete-time system using Algorithm 1 is crucial. This issue is also discussed in Remark 14 of Appendix C.*

In the following, we only consider the discrete-time terminal ingredients based on Lemma 2. Executing Algorithm 1 to ensure that $\alpha_1 = 10^4$ is valid takes 25 min using $20^3 \cdot 10^2 \cdot 100 = 8 \cdot 10^7$ samples.

Robust trajectory tracking - Evasive maneuver test: In order to demonstrate the applicability of the proposed tracking MPC scheme, we consider an evasive maneuver test (compare ISO norm 3888-2 [39]). In this scenario a car is driving with
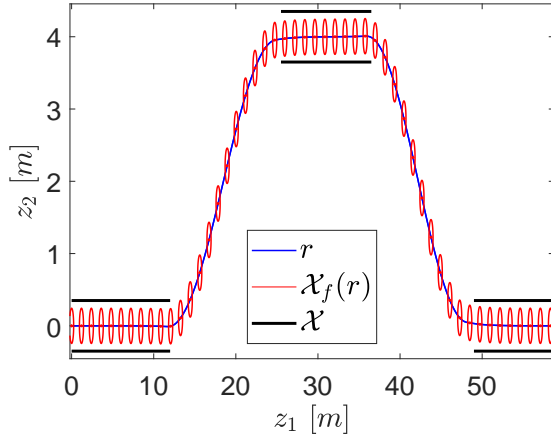
[12]This description does not take into account that $\sin(\psi + \beta)$ and $\cos(\psi + \beta)$ cannot be zero simultaneously. This issue can be circumvented by considering a more detailed description of Θ, e.g. using coupled ellipsoidal constraints.

Method	Continuous-Time		Discrete-time		
	Gridding (Lemma. 4)	Convex (Prop. 5)	Gridding (Lemma. 2)	Convex (Prop. 1)	
#LMIs	200	$4^4 = 256$	7994	$4^6 = 4096$	$6^6 = 46.656$
computational time	12 s	10 s	783 s	356 s	3h 18min
$\max_{r\in\mathcal{Z}_r}\lambda_{\max}(P_f(r))$	$3.3\cdot10^3\cdot h$	$8.3\cdot10^4\cdot h$	$3.5\cdot10^3$	$4\cdot10^7$	$3.8\cdot10^7$

TABLE I

COMPUTATIONAL DEMAND AND CONSERVATISM OF DIFFERENT OFFLINE COMPUTATIONS - CSTR.

$v = 20 \ m/s$ and performs two consecutive lane changes to simulate the avoidance of a possible obstacle. The basic setup, with a feasible reference trajectory r, additional path constraints[13] $\mathcal{X}$ and the terminal set (projected on $z_1 \times z_2$) can be seen in Figure 2. The terminal set size is restricted by the input constraint on u_δ and the path constraint $\mathcal{X}$, yielding the terminal set size $\alpha = \alpha_2 \approx 10^2$. For comparison, we also computed a terminal cost for this specific *given* trajectory based on an LTV description [17]. The generic offline computation results in a roughly five times larger terminal cost, which gives an indication of the conservatism.

with $\epsilon = 2.5\cdot10^{-4}$. The resulting robust tracking MPC scheme guarantees (uniform) practical exponential stability and robust constraint satisfaction, for details see Appendix B and [21].

We simulated the closed-loop MPC using random disturbances $\|w(t)\| = \hat{w}$ and compared the performance to MPC without terminal constraints ($V_f = 0$, UC, [22]) and MPC with terminal equality constraint ($\mathcal{X}_f(r) = x_r$, TEC). To enable a comparison of the computational demand we fixed the number of iterations in CasADi to 1 per time step, resulting in online computation time of approx 13 ms for all three approaches. The corresponding results can be seen in Figures 3 and 4.
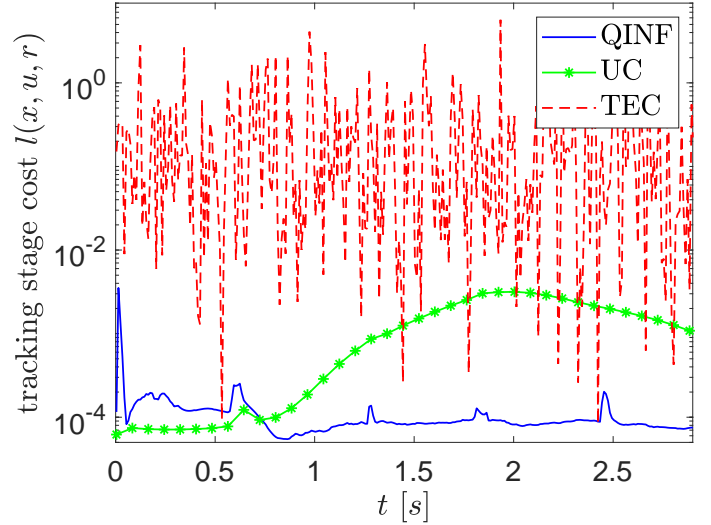

Fig. 2. Evasive maneuver test: Reference trajectory r (blue), terminal sets $\mathcal{X}_f(r)$ (red) and additional state constraints $\mathcal{X}$ (black).

Fig. 3. Evasive maneuver test: Closed-loop tracking stage cost for the proposed terminal constraint tracking MPC (blue,solid,QINF), a corresponding tracking MPC scheme without terminal constraints (green,dash-star,UC) and an MPC scheme with a terminal equality constraint (red,dashed,TEC)

In order to show that the proposed approach can be applied under realistic conditions, we consider additive disturbances $w(t) \in \mathbb{R}^n$ and a prediction horizon of $N = 10$. To ensure robust constraint satisfaction, we use the constraint tightening method proposed in [21], which is based on the achievable contraction[14] rate $\rho = 0.9995$. To ensure robust recursive feasibility, the terminal set needs to be robust positively invariant, which can be ensured for $\|w(t)\| \leq \hat{w} = 1.82 \cdot 10^{-5} = 9.1 \cdot 10^{-3}h$, compare (29) in Proposition 4 of Appendix B. The constraints are tightened over the prediction horizon with a scalar using the method in [21]

$$(x(k|t), u(k|t)) \in (1 - \epsilon_k)\mathcal{Z}, \quad \epsilon_k = \epsilon\frac{1 - \rho^k}{1 - \rho},$$

The closed-loop performance (as measured by the tracking stage cost [15]) of UC and TEC are 10 and 3.000 times larger than the proposed scheme with the terminal cost (QINF), compare Figure 3. Specifically, the MPC without terminal constraints (UC) has a significant (growing) tracking error in the position (see Figure 4), since the UC with a short horizon typically leads to a slower convergence with smaller control action (as stability is not explicitly enforced). On the other side, the terminal equality constraint MPC (TEC) has large deadbeat like input oscillations, which is a result of the terminal constraint with the short prediction horizon. UC and

---

[13]Ideally, these constraints should restrict the overall position of the vehicle. For simplicity we treat them as (time-varying) polytopic constraints on z_2, that require the z_2 position to be within a margin of $\pm 35 \ cm$.

[14]This property is verified by computing a terminal cost, which is valid on the full constraint set $\mathcal{Z}$, compare Prop. 2 and App. B. Analogous to the computation of α, the numerical value of ρ can be ascertained using Alg. 1.

[15]If we ignore the input tracking stage cost and only consider $\|x - x_r\|_Q^2$ as the performance, then the TEC has only 13% of the tracking error of QINF and UC has 30-times the tracking error. If, for some reason, we would only be interested in the tracking error in the input $\|u - u_r\|_R^2$, then UC has only 48% of the error of QINF and TEC has $4.5 \cdot 10^3$ times the error of QINF.
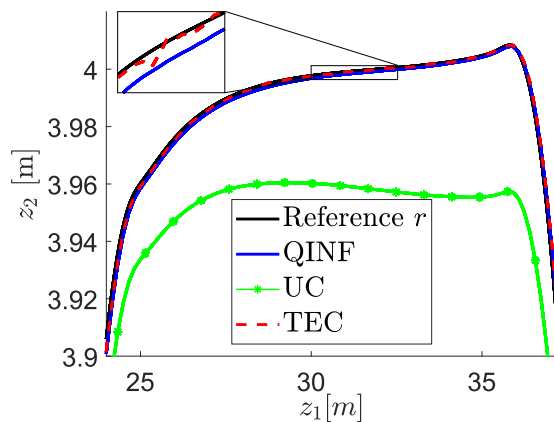
Fig. 4. Evasive maneuver test: Closed-loop trajectory of z_1, z_2 over the time interval $t \in [1.32s, 1.81s]$ with the reference r (black,solid), the MPC based on the proposed terminal ingredients (blue,solid,QINF), a corresponding tracking MPC scheme without terminal constraints (green,dash-star,UC) and an MPC scheme with a terminal equality constraint (red,dashed,TEC).

TEC achieve a similar performance to QINF with $N = 10$, if the prediction horizon[16] is increased to $N = 23$ and $N = 59$, respectively. This increases the online computational demand compared to QINF by 100% and 300%, respectively.

The proposed MPC scheme robustly achieves a small tracking error with a short prediction horizon. This shows that including (suitable) terminal ingredients significantly reduces the tracking error and improves the closed-loop performance, as also articulated in [40].

VI. CONCLUSION

We have presented a procedure to compute terminal ingredients for nonlinear reference tracking MPC schemes offline. The main novelty in this approach is that the offline computation only needs to be done once, irrespective of the setpoint or trajectory to be stabilized. This is possible by computing parameterized terminal ingredients and approximating the nonlinear system locally as a quasi-LPV system, with the reference trajectory to be stabilized as the parameter. Furthermore, we have shown that the reference generic offline computation enables us to design nonlinear MPC schemes that ensure optimal periodic operation despite online changing operation conditions. We have demonstrated the applicability and advantages of the proposed procedure with numerical examples.

The extension of the proposed procedure to large scale nonlinear distributed systems using a seperable formulation is part of future work.

REFERENCES

[1] J. Köhler, M. A. Müller, and F. Allgöwer, "A nonlinear model predictive control framework using reference generic terminal ingredients," *IEEE Trans. Autom. Control*, 2020.
[2] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model Predictive Control: Theory, Computation, and Design*. Nob Hill Pub., 2017.
[3] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, pp. 789–814, 2000.
[4] D. Limon, I. Alvarado, T. Alamo, and E. F. Camacho, "MPC for tracking piecewise constant references for constrained linear systems," *Automatica*, vol. 44, pp. 2382–2387, 2008.
[5] D. Limon, M. Pereira, D. M. de la Peña, T. Alamo, C. N. Jones, and M. N. Zeilinger, "MPC for tracking periodic references," *IEEE Trans. Autom. Control*, vol. 61, pp. 1123–1128, 2016.
[6] D. Limon, A. Ferramosca, I. Alvarado, and T. Alamo, "Nonlinear MPC for tracking piece-wise constant reference signals," *IEEE Trans. Autom. Control*, vol. 63, pp. 3735–3750, 2018.
[7] L. Fagiano and A. R. Teel, "Generalized terminal state constraint for model predictive control," *Automatica*, vol. 49, pp. 2622–2631, 2013.
[8] M. A. Müller, D. Angeli, and F. Allgöwer, "Economic model predictive control with self-tuning terminal cost," *European Journal of Control*, vol. 19, pp. 408–416, 2013.
[9] ——, "On the performance of economic model predictive control with self-tuning terminal cost," *J. Proc. Contr.*, vol. 24, pp. 1179–1186, 2014.
[10] A. Ferramosca, D. Limon, and E. F. Camacho, "Economic MPC for a changing economic criterion for linear systems," *IEEE Trans. Autom. Control*, vol. 59, pp. 2657–2667, 2014.
[11] E. G. Gilbert and K. T. Tan, "Linear systems with state and control constraints: The theory and application of maximal output admissible sets," *IEEE Trans Autom Control*, vol. 36, pp. 1008–1020, 1991.
[12] H. Chen and F. Allgöwer, "A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability," *Automatica*, vol. 34, pp. 1205–1217, 1998.
[13] R. Findeisen, H. Chen, and F. Allgöwer, "Nonlinear predictive control for setpoint families," in *Proc. American Control Conf. (ACC)*, vol. 6, 2000, pp. 260–264.
[14] L. Magni and R. Scattolini, "On the solution of the tracking problem for non-linear systems with MPC," *Int. J. of systems science*, vol. 36, pp. 477–484, 2005.
[15] Z. Wan and M. V. Kothare, "Efficient scheduled stabilizing model predictive control for constrained nonlinear systems," *Int. J. Robust and Nonlinear Control*, vol. 13, pp. 331–346, 2003.
[16] ——, "An efficient off-line formulation of robust model predictive control using linear matrix inequalities," *Automatica*, vol. 39, pp. 837–846, 2003.
[17] T. Faulwasser and R. Findeisen, "A model predictive control approach to trajectory tracking problems via time-varying level sets of lyapunov functions," in *Proc. 50th IEEE Conf. Decision and Control (CDC), European Control Conf. (ECC)*, 2011, pp. 3381–3386.
[18] E. Aydiner, M. A. Müller, and F. Allgöwer, "Periodic reference tracking for nonlinear systems via model predictive control," in *Proc. European Control Conf. (ECC)*, 2016, pp. 2602–2607.
[19] J. Köhler, M. A. Müller, and F. Allgöwer, "MPC for nonlinear periodic tracking using reference generic offline computations," in *Proc. IFAC Conf. Nonlinear Model Predictive Control*, 2018, pp. 656–661.
[20] J. B. Rawlings and D. Q. Mayne, *Model predictive control: Theory and design*. Nob Hill Pub., 2009.
[21] J. Köhler, M. A. Müller, and F. Allgöwer, "A novel constraint tightening approach for nonlinear robust model predictive control," in *Proc. American Control Conf. (ACC)*, 2018, pp. 728–734.
[22] ——, "Nonlinear reference tracking: An economic model predictive control perspective," *IEEE Trans. Autom. Control*, vol. 64, pp. 254–269, 2019.
[23] S. P. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear matrix inequalities in system and control theory*. SIAM, 1994.
[24] P. A. Parrilo, "Semidefinite programming relaxations for semialgebraic problems," *Mathematical programming*, vol. 96, pp. 293–320, 2003.
[25] I. R. Manchester and J.-J. E. Slotine, "Control contraction metrics: Convex and intrinsic criteria for nonlinear feedback design," *IEEE Trans. Autom. Control*, vol. 62, pp. 3046–3053, 2017.
[26] R. Wang, R. Tóth, and I. R. Manchester, "A comparison of LPV gain scheduling and control contraction metrics for nonlinear control," in *Proc. 3rd IFAC Workshop on Linear Parameter Varying Systems (LPVS)*, 2019, pp. 44–49.
[27] W. J. Rugh and J. S. Shamma, "Research on gain scheduling," *Automatica*, vol. 36, pp. 1401–1425, 2000.
[28] P. Apkarian and H. D. Tuan, "Parameterized LMIs in control theory," *SIAM journal on control and optimization*, vol. 38, pp. 1241–1264, 2000.
[29] V. F. Montagner, R. C. Oliveira, V. J. Leite, and P. L. Peres, "Gain scheduled state feedback control of discrete-time systems with time-varying uncertainties: an LMI approach," in *Proc. 44th IEEE Conf. Decision and Control (CDC)*, 2005, pp. 4305–4310.

[16]For this second comparison, we did not limit the number of iterations for UC and TEC, since we were unable to achieve a similar performance with UC using only 1 iterations (which may be due to the lack of a good warmstart).

[30] W.-J. Mao, "Robust stabilization of uncertain time-varying discrete systems and comments on an improved approach for constrained robust model predictive control," *Automatica*, vol. 39, pp. 1109–1112, 2003.

[31] C. Scherer and S. Weiland, "Linear matrix inequalities in control," *Lecture Notes, Delft University, The Netherlands*, vol. 3, 2000.

[32] D. W. Griffith, L. T. Biegler, and S. C. Patwardhan, "Robustly stable adaptive horizon nonlinear model predictive control," *J. Proc. Contr.*, vol. 70, pp. 109–122, 2018.

[33] R. Amrit, J. B. Rawlings, and D. Angeli, "Economic optimization using model predictive control with a terminal cost," *Annual Reviews in Control*, vol. 35, pp. 178–186, 2011.

[34] J. F. Sturm, "Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones," *Optimization methods and software*, vol. 11, pp. 625–653, 1999.

[35] J. A. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi: a software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, vol. 11, no. 1, pp. 1–36, 2019.

[36] J. Bailey, F. Horn, and R. Lin, "Cyclic operation of reaction systems: Effects of heat and mass transfer resistance," *AIChE Journal*, vol. 17, pp. 818–825, 1971.

[37] T. Faulwasser, L. Grüne, and M. A. Müller, "Economic nonlinear model predictive control," *Foundations and Trends® in Systems and Control*, vol. 5, pp. 1–98, 2018.

[38] J. Kong, M. Pfeiffer, G. Schildbach, and F. Borrelli, "Kinematic and dynamic vehicle models for autonomous driving control design," in *IEEE Intelligent Vehicles Symposium (IV)*, 2015, pp. 1094–1099.

[39] "ISO 3888-2: Test track for a severe lane-change manoeuvre - Part 2: Obstacle avoidance," Berlin, Tech. Rep., 2011.

[40] D. Mayne, "An apologia for stabilising terminal conditions in model predictive control," *Int. J. Control*, vol. 86, no. 11, pp. 2090–2095, 2013.

[41] D. Angeli, "A Lyapunov approach to incremental stability properties," *IEEE Trans. Autom. Control*, vol. 47, pp. 410–421, 2002.

[42] D. N. Tran, B. S. Rüffer, and C. M. Kellett, "Incremental stability properties for discrete-time systems," in *Proc. 55th IEEE Conf. Decision and Control (CDC)*, 2016, pp. 477–482.

[43] W. Lohmiller and J.-J. E. Slotine, "On contraction analysis for non-linear systems," *Automatica*, vol. 34, pp. 683–696, 1998.

[44] F. Bayer, M. Bürger, and F. Allgöwer, "Discrete-time incremental ISS: A framework for robust NMPC," in *Proc. European Control Conf. (ECC)*, 2013, pp. 2068–2073.

[45] S. Yu, C. Böhm, H. Chen, and F. Allgöwer, "Robust model predictive control with disturbance invariant sets," in *Proc. American Control Conf. (ACC)*, 2010, pp. 6262–6267.

[46] S. Yu, C. Maier, H. Chen, and F. Allgöwer, "Tube MPC scheme based on robust control invariant set with application to lipschitz nonlinear systems," *Systems & Control Letters*, vol. 62, pp. 194–200, 2013.

[47] L. Chisci, J. A. Rossiter, and G. Zappa, "Systems with persistent disturbances: predictive control with restricted constraints," *Automatica*, vol. 37, pp. 1019–1028, 2001.

[48] M. A. Müller and K. Worthmann, "Quadratic costs do not always work in MPC," *Automatica*, vol. 82, pp. 269–277, 2017.

[49] A. Isidori, *Nonlinear Control Systems*. Springer, 2013.

[50] M. Hertneck, J. Köhler, S. Trimpe, and F. Allgöwer, "Learning an approximate model predictive controller with guarantees," *IEEE Control Systems Letters*, vol. 2, no. 3, pp. 543–548, 2018.

[51] J. Köhler, R. Soloperto, M. A. Müller, and F. Allgöwer, "A computationally efficient robust model predictive control framework for uncertain nonlinear systems," submitted to IEEE Transactions on Automatic Control, 2019, arXiv preprint arXiv:1910.12081.

[52] T. Faulwasser, "Optimization-based solutions to constrained trajectory-tracking and path-following problems," Ph.D. dissertation, Otto-von-Guericke-Universität Magdeburg, 2012.

APPENDIX

In Appendix A, the connection between *incremental system properties* and the considered reference generic terminal ingredients are discussed. In Appendix B, these incremental stability properties are used to extend the approach to *robust* reference tracking, by introducing a simple constraint tightening to ensure robust constraint satisfaction under additive disturbances. In Appendix C, the derivations for the reference generic offline computations (Prop. 1) are extended to *continuous-time* systems. In Appendix D, the procedure is extended to nonlinear *output tracking stage costs*, for both discrete-time and continuous-time systems.

A. (Local) Incremental exponential stabilizability

In the following we clarify the connection between incremental stabilizability properties and the terminal ingredients.

Definition 1. *A set of reference trajectories r specified by some dynamic inclusion $r(t + 1) \in \mathcal{R}(r(t))$ is locally incrementally exponentially stabilizable for the system* (1)*, if there exist constants $\rho \in (0, 1), M, c > 0$ and a control law $\kappa(x, r)$, such that for any initial condition satisfying $\|x(0) - x_r(0)\| \leq c$, the trajectory $x(t)$ with $x(t + 1) = f(x(t), \kappa(x(t), r(t)))$ satisfies $\|x(t) - x_r(t)\| \leq M\rho^t\|x(0) - x_r(0)\|, \forall t \geq 0$.*

This definition is closely related to the concept of universal exponential stabilizability [25], which characterizes the stabilizability of arbitrary trajectories in continuous-time. One of the core differences in the definitions is the treatment of constraints, i.e. we study stabilizability of classes of trajectories r that satisfy certain constraints, compare Assumption 1 and Remark 1. This difference is crucial when discussing local versus global stabilizability and constrained control.

The following proposition shows that the conditions in Lemma 1 directly imply local incremental exponential stabilizability of the reference trajectory.

Proposition 2. *Suppose that there exist matrices $P_f(r)$, $K_f(r)$ that satisfy the conditions in Lemma 1. Then the control law $k_f(x, r) = u_r + K_f(r)(x - x_r)$ locally incrementally exponentially stabilizes any reference r satisfying Assumption 1.*

Proof. The following proof follows the arguments of [22, Prop. 1,2]. For any $\|x(0) - x_r(0)\| \leq c$ with $c = \sqrt{\alpha/c_u}$, we have $x(0) \in \mathcal{X}_f(r)$, with α, c_u according to Lemma 1. Thus, the terminal cost $V_f(x, r)$ is a local incremental Lyapunov function that satisfies

$$V_f(x(t+1), r(t+1)) \leq \rho^2 V_f(x(t), r(t)), \ \rho^2 = 1 - \frac{\lambda_{\min}(Q)}{c_u},$$

and thus

$$\|x(t) - x_r(t)\| \leq \rho^t M\|x_r(0) - x(0)\|, \quad M = \sqrt{c_u/c_l}.$$

$\square$

Remark 9. *This result establishes local incremental stabilizability with the incremental Lyapunov function $V_f(x, r)$ based on properties of the linearization, compare [22, Prop. 1]. This system property is a natural extension of previous works on*

incremental stability and corresponding incremental Lyapunov functions, see [41], [42], [22, Ass. 1]. This property implies stabilizability of $(A(r), B(r))$ around any (fixed) steady-state $r^+ = r$, but it does not necessarily imply stabilizability of $(A(r), B(r))$ for arbitrary $r \in \mathcal{Z}_r$, as $P_f(r)$ might decrease along the trajectory.*

For continuous-time systems, an analogous result exists based on contraction metrics and universal stabilizability [25].

The following proposition shows that in the absence of constraints we recover non-local results similar to [25].

Proposition 3. *Consider $\mathcal{Z}_r = \mathcal{Z} = \mathbb{R}^{n+m}$. Suppose that there exist matices $P_f(r)$, $K_f(r)$ that satisfy the conditions in Lemma 1. Assume further that $c_l I \leq P_f(r) \leq c_u I$ for all $r \in \mathbb{R}^{n+m}$ with some constants c_l, c_u and $K_f(r) = K(x_r)$. Then any reference r satisfying Assumption 1 is exponentially incrementally stabilizable with the control law*

$$\kappa(x, r) = K(x)x - K(x_r)x_r + u_r,$$

i.e., for any initial condition $x(0) \in \mathbb{R}^n$ the state trajectory $x(t + 1) = f(x(t), \kappa(x(t), r(t)))$ satisfies $\|x(t) - x_r(t)\| \leq M\rho^t\|x(0) - x_r(0)\|$.

Proof. Consider an auxiliary (pre-stabilized) system defined by $\tilde{f}(x, v) = f(x, K(x)x + v)$. Consider a reference r generated by some input trajectory u_r with the system dynamics (1) (Ass. 1) and some initial condition $x_r(0)$ resulting in the state reference x_r. Now, consider a reference $\tilde{r}$ generated by the input $v_r(t) = u_r(t) - K(x_r(t))x_r(t)$ with the system dynamics according to $\tilde{f}$ and the same initial condition. Due to the definition of the auxiliary system we have $\tilde{x}_r(t) = x_r(t)$, $\forall t \geq 0$. For an arbitrary, but fixed input v, stability of the reference trajectory is equivalent to contractivity of the nonlinear time-varying system $\tilde{f}(x, t)$. This can be established with the contractivity metric $P_f(r(t)) = P_f(x_r(t), t)$, compare [43]. $\square$

In the absence of constraints, it is crucial that P_f has a constant lower and upper bound. If the matrix K_f depends on the full reference r (not just x_r), the controller κ in Proposition 3 is not necessarily well defined.

Remark 10. *The relation between the controller k_f (Prop. 2) and κ (Prop. 3), is that of reference tracking versus pre-stabilization. The first one is more natural in the context of tracking MPC and contains existing results for the design of terminal ingredients as special cases [12], [17], [18]. The second controller κ allows for non-local stability results and is more suited for unconstrained control problems [25]. For constant matrices K the two controllers are equivalent, but the incremental Lyapunov functions (and thus terminal costs) are differently parameterized ($P_f(r)$, $P_f(x, u)$).*

Remark 11. *The problem of computing reference generic terminal ingredients is equivalent to computing an incrementally stabilizing controller and is thus strongly related to the computation of robust positive invariant (RPI) tubes in nonlinear robust MPC schemes, compare [44], [21]. For comparison, in [45], [46] constant matrices P_f, K_f are computed that certify incremental stability for continuous-time*

systems (by considering small Lipschitz nonlinearities or by describing the linearization as a convex combination of different linear systems). This approach can be directly extended to more general nonlinear systems using the proposed terminal ingredients. In particular, by changing the stage cost to

$$\ell(x, u, r) = \|u - u_r + K(x_r)x_r - K(x)x\|_R^2$$

one can design a nonlinear version of [47], compare also [21]. A detailed description of a corresponding nonlinear robust tube based (tracking) MPC scheme based on incremental stabilizability can be found in Appendix B.

Remark 12. *In case a system is not exponentially stabilizable [48], it might be possible to make a nonlinear transformation resulting in a quadratically stabilizable system, (see for example nonlinear systems in normal form [49]).*

B. Robust reference tracking

In the following, we summarize the theoretical results for robust reference tracking based on the reference generic terminal ingredients and [21], where robust setpoint stabilization without terminal constraints was considered. This method is applicable to nonlinear incrementally stabilizable systems (Sec. A) with polytopic constraints and additive disturbances and can be thought of as a nonlinear version of [47].

1) Setup: We consider nonlinear discrete-time systems subject to additive bounded disturbances and polytopic constraints

$$x(t+1) = f(x(t), u(t)) + w(t), \quad \|w\| \leq \hat{w},$$
$$\mathcal{Z} = \{r \in \mathbb{R}^{n+m} | \ L_j r \leq 1, \quad j = 1, \ldots, q\}.$$

2) Incremental stabilizability:

Assumption 3. *[21, Ass. 1][22, Ass. 1] There exist a control law $\kappa : \mathbb{R}^n \times \mathcal{Z} \to \mathbb{R}^m$, an incremental Lyapunov function $V_\delta : \mathbb{R}^n \times \mathcal{Z} \to \mathbb{R}_{\geq 0}$, that is continuous in the first argument and satisfies $V_\delta(x_r, x_r, u_r) = 0$ for all $(x_r, u_r) \in \mathcal{Z}$, and parameters $c_{\delta,l}$, $c_{\delta,u}$, δ_{loc}, $c_j \in \mathbb{R}_{>0}$, $\rho \in (0, 1)$, such that the following properties hold for all $(x, x_r, u_r) \in \mathbb{R}^n \times \mathcal{Z}$, $r^+ = (x_r^+, u_r^+) \in \mathcal{Z}$ with $V_\delta(x, r) \leq \delta_{loc}$:*

$$c_{\delta,l}\|x - x_r\|^2 \leq V_\delta(x, x_r, u_r) \leq c_{\delta,u}\|x - x_r\|^2, \quad (26a)$$
$$L_j(x - x_r, \kappa(x, x_r, u_r) - u_r) \leq c_j\sqrt{V_\delta(x, x_r, u_r)}, \quad (26b)$$
$$V_\delta(x^+, x_r^+, u_r^+) \leq \rho^2 V_\delta(x, x_r, u_r), \quad (26c)$$

with $x^+ = f(x, \kappa(x, x_r, u_r))$, $x_r^+ = f(x_r, u_r)$, $j = 1, \ldots, q$.

This assumption implies incremental stabilizability (Def. 1) for all feasible trajectories r, i.e., $r(t+1) \in \mathcal{R}(r(t))$ (Ass. 1). For $\kappa(x, x_r, u_r) = u_r$ this reduces to incremental stability and correspondingly the robust MPC method in [44] can also be used. This assumption can be verified by using Algorithm 2 to compute a terminal cost that is valid on $\mathcal{Z}$, compare Proposition 2. The contraction rate ρ (26c), is used to design a generic constraint tightening to ensure robust constraint satisfaction. The condition (26b) is satisfied if the control law κ is locally Lipschitz continuous, compare also [21].

3) Constraint tightening: The constraints are tightened using the following scalar operations

$$\epsilon_j = c_j\sqrt{c_u}\hat{w}, \quad \epsilon_{j,k} = \frac{1 - \rho^k}{1 - \rho}\epsilon_j, k = 0, \ldots, N,$$
$$\mathcal{Z}_k = \{r \in \mathbb{R}^n | \ L_j r \leq 1 - \epsilon_{j,k}, \quad j = 1, \ldots, q\}.$$

The following bound on the disturbance is required to ensure that the tightened constraints are non-empty, i.e., $0 \in \text{int}(\mathcal{Z}_N)$:

$$\hat{w} < \frac{1}{\max_j c_j}\frac{1}{\sqrt{c_u}}\frac{1 - \rho^N}{1 - \rho}, \quad (27)$$

4) Terminal ingredients: In [21] the robust constraint tightening is considered for an MPC scheme without terminal constraints, compare Remark 4. Some details regarding the extension/modification of the robust MPC scheme to a setting with terminal constraints are based on [50].

Assumption 4. *There exist matrices $K_f(r) \in \mathbb{R}^{m \times n}$, $P_f(r) \in \mathbb{R}^{n \times n}$ with $c_l I_n \leq P_f(r) \leq c_u I_n$, a terminal set $\mathcal{X}_f(r) = \{x \in \mathbb{R}^n | \ V_f(x, r) \leq \alpha_w\}$ with the terminal cost $V_f(x, r) = \|x - x_r\|_{P_f(r)}^2$, such that the following properties hold for any $r \in \mathcal{Z}_r$, any $x \in \mathcal{X}_f(r)$, any $r^+ \in \mathcal{R}(r)$ and any $w \in \mathcal{W}_N$*

$$V_f(x^+, r^+) \leq V_f(x, r) - \ell(x, k_f(x, r), r), \quad (28a)$$
$$V_f(x^+ + w, r^+) \leq \alpha_w, \quad (28b)$$
$$(x, k_f(x, r)) \in \mathcal{Z}_N, \quad (28c)$$

with $x^+ = f(x, k_f(x, r))$, $k_f(x, r) = u_r + K_f(r) \cdot (x - x_r)$, $\mathcal{W}_N = \{w \in \mathbb{R}^n | \ \|w\| \leq \hat{w}_N = \hat{w}\rho^N\sqrt{c_{\delta,u}/c_{\delta,l}}\}$, and positive constants c_l, c_u, α_w.

Compared to the nominal case (Ass. 2), we have a smaller terminal set size α_w due to the tightened constraints (28c) and an RPI condition that needs to be verified (28b). Due to the quadratic nature of the terminal cost and the stage cost, (28a) implies $V_f(x^+, r^+) \leq \rho_f^2 V_f(x, r,)$, with some $\rho_f \in (0, 1)$, e.g. $\rho_f = 1 - \lambda_{\min}(Q)/c_u$.

Proposition 4. *Let Assumption 2 hold and assume that $\hat{w}$ satisfies (27). Then the terminal ingredients (Ass. 2) satisfy (28c) with a positive constant α_w. Suppose further that*

$$\hat{w} \leq \sqrt{\frac{\alpha_w c_{\delta,l}}{c_{\delta,u}c_u}}\frac{1 - \rho_f}{\rho^N}. \quad (29)$$

Then (28b) and thus Assumption 4 is satisfied.

Proof. Condition (28a) directly follows fom Assumption 2. Inequality (27) ensures that $0 \in \text{int}(\mathcal{Z}_N)$, which in combination with the quadratic bounds on V_f and linear bounds on k_f ensures that (28c) is satisfied for some positive constant α_w, compare the proof of Lemma 1, Algorithm 1 and the optimization problem (24) for the computation of α_w.

Using the quadratic nature of the terminal cost, a sufficient condition for (28b) is given by

$$V_f(x^+ + w, r^+)$$
$$\leq V_f(x^+, r^+) + 2\sqrt{c_u V_f(x^+, r^+)}\hat{w}_N + c_u\hat{w}_N^2 \leq \alpha_w,$$

with $\|w\| \leq \hat{w}_N$. Using the contraction rate ρ_f to bound $V_f(x^+, r^+) \leq \rho_f^2\alpha_w$, this condition reduces to $\hat{w}_N \leq (1 - $

$\rho_f)\sqrt{\alpha_w/c_u}$. The inequality on $\hat{w}$ follows from the definition of $\hat{w}_N$. $\qquad\square$

5) Robust tracking MPC: The robust tracking MPC is based on the following MPC optimization problem

$$V(x(t), r(\cdot|t)) = \min_{u(\cdot|t)} J_N(x(\cdot|t), u(\cdot|t), r(\cdot|t)) \tag{30a}$$

$$\text{s.t. } x(k+1|t) = f(x(k|t), u(k|t)), \tag{30b}$$

$$x(0|t) = x(t), \tag{30c}$$

$$(x(k|t), u(k|t)) \in \mathcal{Z}_k, \tag{30d}$$

$$x(N|t) \in \mathcal{X}_f(r(N|t)). \tag{30e}$$

Compared to (5), in this optimization problem the state and input constraints are tightened.

6) Theoretical guarantees:

Theorem 2. *Let Assumptions 1, 3 and 4 hold. Assume further that $\hat{w} \leq \sqrt{\delta_{loc}/c_{\delta,u}}$ and that (30) is feasible at $t = 0$. The optimization problem (30) is recursively feasible and the tracking error $e_r = 0$ is (uniformly) practically exponentially stable for the resulting closed-loop system (6).*

Proof. The proof is analogous to [21], except for the satisfaction of the terminal constraint, which is guaranteed by Assumption 4, compare also [50, Thm. 7]. $\qquad\square$

Note that both the size of the constraint set (27) and the local incremental stabilizability (29) lead to hard bounds on the size of the disturbance $\hat{w}$, that can be considered in this approach. This approach can also be extended to utilize a general nonlinear state and input dependent characterization of the disturbance in order to reduce the conservatism, compare [51].

C. Continuous-time dynamics

In the following, we summarize the continuous-time analog of the reference generic offline computations in Section III. The nonlinear continuous-time dynamics are given by

$$\frac{d}{dt}[x] = \dot{x} = f(x, u)$$

and f is assumed to be twice continuously differentiable. The following condition characterizes the admissible reference trajectories as the continuous-time analog of Assumption 1.

Assumption 5. *The reference signal $r : \mathbb{R} \to \mathbb{R}^{n+m}$ is continuously differentiable and satisfies*

$$r(t) \in \mathcal{Z}_r \subseteq int(\mathcal{Z}),$$
$$\dot{r}(t) \in \mathcal{R}(r(t)) = \{(\dot{x}_r, \dot{u}_r)| \ \dot{x}_r = f(x_r, u_r), \ \|\dot{u}_r\|_\infty \leq u_{\max}\},$$

for all $t \geq 0$ with some constant $u_{\max}$.

Remark 13. *This assumption can be generalized to consider non-differentiable reference signal r ($\dot{u}_r$ unbounded). In this case, the terminal cost P_f should be parameterized with parameters θ_i independent of u_r, i.e., $P_f(x_r)$.*

The following assumption characterizes the terminal ingredients, as a continuous-time analog of Assumption 2.

Assumption 6. *There exist matrices $K_f(r) \in \mathbb{R}^{m \times n}$, $P_f(r) \in \mathbb{R}^{n \times n}$ with $c_l I_n \leq P_f(r) \leq c_u I_n$, P_f continuously differentiable, a terminal set $\mathcal{X}_f(r) = \{x \in \mathbb{R}^n | \ V_f(x, r) \leq \alpha\}$ with the terminal cost $V_f(x, r) = \|x - x_r\|^2_{P_f(r)}$, such that the following properties hold for any $r \in \mathcal{Z}_r$, any $x \in \mathcal{X}_f(r)$ and any $\dot{r} \in \mathcal{R}(r)$*

$$\frac{d}{dt}[V_f(x, r)] \leq -\ell(x, k_f(x, r), r), \tag{31}$$

$$(x, k_f(x, r)) \in \mathcal{Z}, \tag{32}$$

with positive constants c_l, c_u, α and

$$\dot{x} = f(x, k_f(x, r)), \quad k_f(x, r) = u_r + K_f(r) \cdot (x - x_r),$$
$$\frac{d}{dt}V_f(x, r)$$
$$= 2(x - x_r)^\top P_f(r)(\dot{x} - \dot{x}_r) + \|x - x_r\|^2_{\frac{d}{dt}P_f(r)}.$$

The following Lemma provides sufficient conditions for Assumption 6 to be satisfied based on the linearization, as a continuous-time version of Lemma 1.

Lemma 3. *Assume that there exist matrices $K_f(r) \in \mathbb{R}^{m \times n}$ continuous in r and a positive definite matrix $P_f(r) \in \mathbb{R}^{n \times n}$ continuously differentiable with respect to r, such that for any $r \in \mathcal{Z}_r$, $\dot{r} \in \mathcal{R}(r)$, the following matrix inequality is satisfied*

$$(A(r) + B(r)K_f(r))^\top P_f(r) + P_f(r)(A(r) + B(r)K_f(r))$$
$$+ \sum_{j=1}^{n+m} \frac{\partial P_f}{\partial r_j}\dot{r}_j + (Q + \epsilon I_n + K_f(r)^\top R K_f(r)) \leq 0 \tag{33}$$

with some positive constant ϵ. Then there exists a sufficiently small constant α, such that P_f, K_f satisfy Assumption 6.

Proof. Denote $\Delta x = x - x_r$, $\Delta u = K_f(r)\Delta x$. Using a first order Taylor approximation at $r = (x_r, u_r)$, we get

$$f(x, u) = f(x_r, u_r) + A(r)\Delta x + B(r)\Delta u + \Phi_r(\Delta x),$$

with the remainder term Φ_r. The terminal cost satisfies

$$\frac{d}{dt}V_f(x, r) = 2(x - x_r)^\top P_f(r)(\dot{x} - \dot{x}_r)$$
$$+ (x - x_r)^\top \left[\sum_{j=1}^{n+m} \frac{\partial P_f}{\partial r_j}\dot{r}_j\right](x - x_r)$$
$$\overset{(33)}{\leq} -\ell(x, k_f(x, r)) - \epsilon\|\Delta x\|^2 + 2(x - x_r)^\top P_f(r)\Phi_r(\Delta x).$$

For α sufficiently small, this implies (31) (due to the arbitrarily small local Lipschitz bound on the higher order terms Φ_r). Constraint satisfaction (32) is guaranteed analogous to Lemma 1. $\qquad\square$

The following Lemma provides corresponding LMI conditions, similar to Lemma 2.

Lemma 4. *Suppose that there exists a matrix $Y(r)$ continuous in r and $X(r)$ continuously differentiable with respect to r, that satisfy the constraints in (40) for all $r \in \mathcal{Z}_r$, $\dot{r} \in \mathcal{R}(r)$. Then $P_f = X^{-1}$, $K_f = Y P_f$ satisfy (33).*

Proof. Multiplying (33) from left and right with $X(r)$ yields

$$(A(r)X(r) + B(r)Y(r))^\top + (A(r)X(r) + B(r)Y(r))$$
$$+ X(r)\frac{d}{dt}[X^{-1}(r)]X(r) + X(r)(Q + \epsilon I_n)X(r)$$
$$+ Y(r)^\top RY(r) \leq 0.$$

Note that the chain rule applied to the inverse of X yields

$$X(r)\frac{d}{dt}\left[X^{-1}(r)\right]X(r) = -\frac{d}{dt}[X(r)] = -\sum_{i=1}^{p} X_i \frac{\partial \theta_i}{\partial r}\dot{r}.$$

Applying the Schur complement results in (40). $\square$

If a gridding approach is considered to compute the terminal ingredients, one needs to grid $r \in \mathcal{Z}_r$ and consider the 2^m vertices of $\dot{u}_r$ (since (40) is affine in $\dot{u}_r$).

For the convex approach, polytopic sets need to be constructed such that $(\theta, \dot{\theta}) \in \Theta \times \Omega = \overline{\Theta}, \ \forall r \in \mathcal{Z}_r, \ \dot{r} \in \mathcal{R}(r)$. The following proposition provides the corresponding LMI conditions based on the vertices of $\overline{\Theta}$, similar to Proposition 1.

Proposition 5. *Suppose that there exists matrices X_i, Y_i, Λ_i, $X_{\min}$ that satisfy the constraints in (41). Then the following matrices satisfy (33):*

$$P_f(r) = X^{-1}(r), \quad K_f(r) = Y(r)P_f(r).$$

Proof. The proof is analogous to Proposition 1, based on multi-convexity and Lemma 4. $\square$

Remark 14. *The continuous-time formulation is suitable if the online MPC optimization considers continuous-time input signals, instead of piece-wise constant inputs (as is common in many numerical implementations). Nevertheless, if the sampling time h is sufficiently small, the continuous-time terminal cost (scaled by $1/h$) might satisfy the discrete-time conditions (Ass. 2) with a piece-wise constant input. This can be favorable since the corresponding offline optimization problem (40) or (41) is often easier formulated and faster solved, especially if a non-trivial discretization is considered. Algorithm 1 can be used to ensure the validity of the computed terminal ingredients with the zero-order hold input (instead of the continuous-time feedback).*

D. Output tracking stage cost

In the following, we discuss how the derivation in Section III can be extended to deal with an output tracking stage cost. As an alternative to (3), consider the following output reference tracking stage cost

$$\ell(x, u, r) = \|h(x, u) - h(x_r, u_r)\|^2_{S(r)}, \tag{34}$$

with a nonlinear twice continuously differentiable output function $h : \mathcal{Z} \to \mathbb{R}^p$ and a positive definite weighting matrix $S(r)$, which assumed to be continuous in r. Such a stage cost can be used for output regulation, output trajectory tracking, output path following or manifold stabilization, compare [52].

We denote the Jacobian of the output h around an arbitrary point $r \in \mathcal{Z}_r$ by

$$C(r) = \left[\frac{\partial h}{\partial x}\right]\bigg|_{(x,u)=r}, \quad D(r) = \left[\frac{\partial h}{\partial u}\right]\bigg|_{(x,u)=r}. \tag{35}$$

The following lemma establishes sufficient conditions for Assumption 2 with the stage cost (34) based on the linearization, similar to Lemma 1.

Lemma 5. *Suppose that f, h are twice continuously differentiable. Assume that there exists a matrix $K_f(r) \in \mathbb{R}^{m \times n}$ and a positive definite matrix $P_f(r) \in \mathbb{R}^{n \times n}$ continuous in r, such that for any $r \in \mathcal{Z}_r$, $r^+ \in \mathcal{R}(r)$, the following matrix inequality is satisfied*

$$(A(r) + B(r)K_f(r))^\top P_f(r^+)(A(r) + B(r)K_f(r)) - P_f(r) \tag{36}$$

$$\leq -(C(r) + D(r)K_f(r))^\top S(r)(C(r) + D(r)K_f(r)) - \tilde{\epsilon}I_n$$

with some positive constant $\tilde{\epsilon}$. Then there exists a sufficiently small constant α, such that P_f, K_f satisfy Assumption 2.

Proof. A first order Taylor approximation at $r = (x_r, u_r)$ yields

$$h(x, k_f(x, r)) - h(x_r, u_r)$$
$$= (C(r) + D(r)K_f(r))\Delta x + \tilde{\Phi}_r(\Delta x),$$

with the remainder term $\tilde{\Phi}_r$ and $\Delta x = x - x_r$. The stage cost satisfies

$$\ell(x, k_f(x, r), r) \tag{37}$$
$$\geq \|(C(r) + D(r)K_f(r))\Delta x\|^2_{S(r)} + \|\tilde{\Phi}_r(\Delta x)\|^2_{S(r)}$$
$$- 2\|\tilde{\Phi}_r(\Delta x)\|_{S(r)}\|(C(r) + D(r)K_f(r))\Delta x\|_{S(r)}.$$

Given continuity and compactness, there exists a constant

$$c_y = \max_{r \in \mathcal{Z}_r}\|(C(r) + D(r)K_f(r))\|_{S(r)}. \tag{38}$$

For a sufficiently small α, the remainder term $\tilde{\Phi}_r$ satisfies the following (local) Lipschitz bound

$$\|\tilde{\Phi}_r(\Delta x)\|_{S(r)}/\|\Delta x\| =: \tilde{L}_{r,x} \leq \tilde{L}^* := c_y - \sqrt{c_y^2 - \tilde{\epsilon}/2}, \tag{39}$$

for all $x \in \mathcal{X}_f(r)$ and all $r \in \mathcal{Z}_r$. This implies

$$\ell(x, k_f(x, r))$$
$$\overset{(37),(39)}{\geq} \|(C(r) + D(r)K_f(r))\Delta x\|^2_{S(r)} + \tilde{L}^2_{r,x}\|\Delta x\|^2$$
$$- 2\tilde{L}_{r,x}\|\Delta x\|^2\|(Cr) + D(r)K_f(r)\|_{S(r)}$$
$$\overset{(38)}{\geq} \|(C(r) + D(r)K_f(r))\Delta x\|^2_{S(r)} + \tilde{L}_{r,x}(\tilde{L}_{r,x} - 2c_y)\|\Delta x\|^2$$
$$\overset{(39)}{\geq} \|(C(r) + D(r)K_f(r))\Delta x\|^2_{S(r)} + \tilde{L}^*(\tilde{L}^* - 2c_y)\|\Delta x\|^2$$
$$\overset{(39)}{=} \|(C(r) + D(r)K_f(r))\Delta x\|^2_{S(r)} - \tilde{\epsilon}/2\|\Delta x\|^2.$$

The second to last step follows by using the fact that the function $L(L - 2c_y)$ attains it minimum for $L \in [0, L^*]$ at $L = L^*$. Combining the derived bound on $\ell(x, k_f(x, r))$ with (36) ensures that the terminal cost V_f satisfies inequality (11) in

Lemma 1 with the modified stage cost and with $\epsilon = \tilde{\epsilon}/2$. The remainder of the proof is analogous to Lemma 1. $\qquad\square$

Remark 15. *For the linear output $h(x, u) = [Q^{1/2}x; R^{1/2}u] \in \mathbb{R}^{n+m}$ and $S = I$ we recover the conditions in Lemma 1 with the stage cost* (3).

Remark 16. *Depending on the output h and the reference r, there may exist multiple solutions that achieve exact output tracking. Thus, we can in general not expect asymptotic/exponential stability of the reference r, but instead stability of a corresponding set or manifold, compare [52]. Under suitable (incremental) detectability conditions on the output h, we can recover stability of the specific reference trajectory r.*

Based on these conditions, Lemma 6 provides LMI conditions to compute P_f, K_f, similar to Lemma 2. Furthermore, if the parameters θ_i are chosen, such that

$$S^{-1}(r) = S_0 + \sum_{i=1}^{p} \theta_i(r)S_i,$$

then Proposition 6 yields LMI conditions based on the vertices of $\overline{\Theta}$, similar to Proposition 1.

$$\min_{X(r),Y(r),X_{\min}} \quad -\log\det X_{\min} \tag{40a}$$

$$\text{s.t.} \begin{pmatrix} A(r)X(r)+B(r)Y(r)+(A(r)X(r)+B(r)Y(r))^\top - \dfrac{d}{dt}[X(r)] & ((Q+\epsilon)^{1/2}X(r))^\top & (R^{1/2}Y(r))^\top \\ * & -I & 0 \\ * & 0 & -I \end{pmatrix} \leq 0, \tag{40b}$$

$$X_{\min} \leq X(r), \tag{40c}$$

$$\forall r \in \mathcal{Z}_r,\ \dot{r} \in \mathcal{R}(r). \tag{40d}$$

$$\min_{X_i,Y_i,\Lambda_i,X_{\min}} \quad -\log\det X_{\min} \tag{41a}$$

$$\text{s.t.} \begin{pmatrix} A(\theta)X(\theta)+B(\theta)Y(\theta)+(A(\theta)X(\theta)+B(\theta)Y(\theta))^\top - X(\dot{\theta})+X_0 & ((Q+\epsilon)^{1/2}X(\theta))^\top & (R^{1/2}Y(\theta))^\top \\ * & -I & 0 \\ * & 0 & -I \end{pmatrix}$$

$$\leq - \begin{pmatrix} \sum_{i=1}^{p}\theta_i^2\Lambda_i & 0 \\ 0 & 0 \end{pmatrix}, \tag{41b}$$

$$X_{\min} \leq X(\theta), \quad \forall(\theta,\dot{\theta}) \in \text{Vert}(\overline{\Theta}), \tag{41c}$$

$$\Lambda_i + (A_iX_i+B_iY_i)+(A_iX_i+B_iY_i)^\top \geq 0, \quad \Lambda_i \geq 0, \quad i=1,\ldots,p. \tag{41d}$$

Lemma 6. *Suppose that there exists matrices $X(r)$, $Y(r)$ continuous in r, that satisfy the following constraints*

$$\min_{X(r),Y(r),X_{\min}} \quad -\log\det X_{\min} \tag{42a}$$

$$\text{s.t.} \begin{pmatrix} X(r) & (A(r)X(r)+B(r)Y(r))^\top & (C(r)X(r)+D(r)Y(r))^\top & \sqrt{\tilde{\epsilon}}X(r) \\ * & X(r^+) & 0 & 0 \\ * & * & S^{-1}(r) & 0 \\ * & * & * & I \end{pmatrix} \geq 0, \tag{42b}$$

$$X_{\min} \leq X(r), \quad \forall r \in \mathcal{Z}_r,\ r^+ \in \mathcal{R}(r). \tag{42c}$$

Then $P_f = X^{-1}$, $K_f = YP_f$ satisfy (36).

Proof. The proof is similar to Lemma 1, compare also [23]. Define $X(r) = P_f(r)^{-1}$ and $Y(r) = K_f(r)X(r)$. Multiplying (36) from left and right with $X(r)$ yields

$$(A(r)X(r)+B(r)Y(r))^\top X(r^+)^{-1}(A(r)X(r)+B(r)Y(r)) - X(r) + \tilde{\epsilon}X(r)IX(r)$$
$$+ (C(r)X(r)+D(r)Y(r))^\top S(r)(C(r)X(r)+D(r)Y(r)) \leq 0.$$

This can be equivalently written as

$$X(r) - \begin{pmatrix} A(r)X(r)+B(r)Y(r) \\ C(r)X(r)+D(r)Y(r) \\ \sqrt{\tilde{\epsilon}}X(r) \end{pmatrix}^\top \begin{pmatrix} X(r^+)^{-1} & 0 & 0 \\ 0 & S(r) & 0 \\ 0 & 0 & I \end{pmatrix} \begin{pmatrix} A(r)X(r)+B(r)Y(r) \\ C(r)X(r)+D(r)Y(r) \\ \sqrt{\tilde{\epsilon}}X(r) \end{pmatrix} \geq 0$$

Using the Schur complement this reduces to (42), which is linear in X, Y. $\qquad\square$

Proposition 6. *Suppose that there exists matrices X_i, Y_i, Λ_i, $X_{\min}$ that satisfy the following constraints*

$$\min_{X_i,Y_i,\Lambda_i,X_{\min}} \quad -\log\det X_{\min} \tag{43a}$$

$$\text{s.t.} \begin{pmatrix} X(\theta) & (A(\theta)X(\theta)+B(\theta)Y(\theta))^\top & (C(\theta)X(\theta)+D(\theta)Y(\theta))^\top & \sqrt{\tilde{\epsilon}}X(\theta) \\ * & X(\theta^+) & 0 & 0 \\ * & * & S^{-1}(\theta) & 0 \\ * & * & * & I \end{pmatrix} \geq \begin{pmatrix} \sum_{i=1}^{p}\theta_i^2\Lambda_i & 0 \\ 0 & 0 \end{pmatrix}, \tag{43b}$$

$$X_{\min} \leq X(\theta), \quad \forall(\theta,\theta^+) \in \text{Vert}(\overline{\Theta}), \tag{43c}$$

$$\begin{pmatrix} 0 & (A_iX_i+B_iY_i)^\top & (C_iX_i+D_iY_i)^\top \\ (A_iX_i+B_iY_i) & 0 & 0 \\ (C_iX_i+D_iY_i) & 0 & 0 \end{pmatrix} \leq \Lambda_i, \quad \Lambda_i \geq 0, \quad i=1,\ldots,p. \tag{43d}$$

Then $P_f = X^{-1}$ and $K_f = YP_f$ satisfy (36).

Proof. The proof is analogous to Proposition 1 based on Lemma 6. The constraint (43d) ensures multi-convexity. $\qquad\square$