

A successive convexification approach for robust receding horizon control

Yana Lishkova and Mark Cannon

Abstract—A novel robust nonlinear model predictive control strategy is proposed for systems with nonlinear dynamics and convex state and control constraints. Using a sequential convex approximation approach and a difference of convex functions representation, the scheme constructs tubes that contain predicted model trajectories, accounting for approximation errors and disturbances, and guaranteeing constraint satisfaction. An optimal control problem is solved as a sequence of convex programs. We develop the scheme initially in the absence of external disturbances and show that the proposed nominal approach is non-conservative, with the solutions of successive convex programs converging to a locally optimal solution for the original optimal control problem. We extend the approach to the case of additive disturbances using a novel strategy for selecting linearization points. As a result we formulate a robust receding horizon strategy with guarantees of recursive feasibility and closed-loop stability.

Index Terms—nonlinear systems, receding horizon control, convex programming, robust control

I. INTRODUCTION

MODEL Predictive Control (MPC), also known as receding horizon control, is an optimization-based control strategy applicable to systems with state and input constraints, model uncertainty and external disturbances. The method relies on solving an optimization problem at each discrete time step. The feasibility of this problem and its solution within a single time step are crucial for implementing the resulting control law [1]. For nonlinear systems, the optimal control problem is in general a nonlinear program (NLP), which is not guaranteed to be solvable within a fixed time interval.

To address this problem various approaches have been developed. For example, dynamic programming can determine globally optimal control policies, however they require heavy computation, which restricts their implementation in real-time applications [1]. Another method, known as sequential quadratic programming (SQP) [2], focuses on iterative approximation approaches that approximate NLPs using sequences of quadratic programming (QP) problems. Despite the existence of efficient methods for solving QPs, these schemes can lead to high computational demand and present difficulties

when accounting for approximation errors and determining the appropriate step-sizes for each iteration to ensure convergence.

An alternative approach replaces the system model with a sequence of models obtained from linear approximations around trajectories of the controlled system (e.g. [3–6]). The seminal work [3] relaxes the nonlinear model dynamics into a set of inequalities, which are subsequently linearized to obtain a sequence of convex optimization problems. Under convexity assumptions on the nonlinear dynamics and the state and input constraints, the solutions of this sequence are shown to converge to a local optimum of the original NLP provided a set of parameters (which are introduced into the objective function to enforce satisfaction of the original model dynamics) are chosen appropriately. However, the required values of cost parameters are not usually known, and the method converges only asymptotically to a feasible solution of the NLP.

More recently, [4] proposed using Jacobian model linearization to construct a sequence of convex optimization problems. The authors use trust regions to account for linearization errors and are thus able to demonstrate convergence to a locally optimal solution of the NLP. The disadvantages of the approach are that it relies on heuristics for determining whether model approximations are sufficiently accurate, and that the iterates are not guaranteed to be feasible for the NLP, so the iteration cannot be terminated prior to convergence.

In [5], a tube (or sequence of sets) containing all possible trajectories of the nonlinear system is constructed by linearizing the model around trajectories and bounding linearization errors, and all trajectories within the tube are required to respect the system state and control constraints. Using successive linearization, the NLP is replaced by a sequence of second-order cone programs (SOCs) [5, 6], each of which generates a feasible solution for the original NLP. The implementation of the approach is complicated by the need to provide bounds on the linearization errors, which can be a computationally difficult task that relies on heuristics. Conservative linearization error bounds can cause slow convergence or infeasibility of the sequence of optimization problems even if the NLP itself is feasible. Furthermore, the available schemes can introduce suboptimality and do not account for external disturbances. The tube-based approach in [7] can ensure convergence to a locally optimal solution of the NLP. However, it is computationally costly and does not account for external disturbances.

Despite the disadvantages of SQP and successive convexification approaches based on linearizing nonlinear dynamic models, the efficiency and practicality of convex NLP approx-

Y.L. acknowledges funding from the EPSRC Doctoral Training Partnership EP/R513295/1, project reference 2280382, and the EPSRC Doctoral Prize scheme EP/W524311/1.

Yana Lishkova and Mark Cannon are with the Department of Engineering Science, University of Oxford, OX1 3PJ, UK (e-mail: yana.lishkova@eng.ox.ac.uk, mark.cannon@eng.ox.ac.uk).

imations are well-known in the context of receding horizon control. Many real-time applications of MPC therefore employ convex descriptions of the model dynamics and constraints (in aerospace applications for example, [8–10] among many others). Moreover, successive convexification incorporating robust tube MPC methods provide new opportunities for computationally efficient receding horizon control, particularly when the model is affected by unknown disturbances.

Approaches that consider external disturbances and model inaccuracies are reviewed in [11–13]. The tube-based strategies of [11–15] address both continuous and discrete systems with various cost functions and disturbance realisations. However, these approaches use invariant sets to define tubes, rely on heuristics to determine parameters or approximate the NLP formulation conservatively. Alternatively, [16] proposes a two-tier approach that simultaneously optimizes tubes and trajectories online for feedback linearizable or minimum phase systems. On the other hand, [17] optimizes a linear feedback law, alternating between a Riccati Recursion and a nominal optimal control problem, with approximate robustness due to unaccounted linearization errors in the nonlinear constraints.

The approach presented here exploits the convexity properties of systems with continuous dynamics and convex state, control and disturbance constraints. We use tubes to bound the effects of model approximations and disturbances, and exploit tight bounds on linearization errors to obtain non-conservative solutions for the disturbance-free case. Although the min-max approach presented relies on enumerating polytopic tube vertices, which can be computationally costly for high-order systems, we discuss strategies to ensure linear growth in the number of variables and constraints with the state dimension.

We assume that the system discrete model has a convex difference (DC) form, i.e. it can be represented as the difference of a pair of convex functions. This is not necessarily restrictive, as any continuous function can be approximated to any desired precision as the difference of two convex functions [18]. Differences of convex functions have been used to develop techniques such as DC programming [18], the convex-concave procedure [19, 20] and novel control invariant set computation methods for nonlinear systems [21], which have proven numerical advantages when applied to nonconvex problems ([22, 23]). Using DC decomposition we derive a tube-based receding horizon controller with robustness to disturbances and guarantees of recursive feasibility and stability.

To this end, the first part of this paper considers the nominal (disturbance-free) case in which the system model is known exactly. The proposed approach solves a sequence of convex programs optimizing control variables and tube cross-sections simultaneously, similar to [24]. Our contributions are a novel initialisation strategy and a proof that the resulting optimization procedure is non-conservative in that its iterations converge to an (at least locally) optimal point of the NLP.

In the disturbance-free case, feasibility is guaranteed if at least one trajectory of the system satisfies the constraints of the problem. However, when unknown disturbances are present, model trajectories under all realisations of the disturbance must be accounted for, and this presents non-trivial additional difficulties for choosing linearisation points and guaranteeing

recursive feasibility. The second part of this paper addresses these problems by considering additive disturbances explicitly. As in the disturbance-free case, the proposed approach approximates the NLP with a sequence of convex programs, and a tube containing all possible trajectories is optimized online. Based on the implementation of a novel linearization technique, we prove recursive feasibility and a form of quadratic stability for the resulting closed-loop system.

The contributions of this paper are as follows:

- A novel tube-based receding horizon strategy with guaranteed stability and recursive feasibility for nonlinear systems with additive disturbances.
- Analysis of convergence of the proposed nominal tube-based approach to a local minimum of the NLP.
- A novel initialisation method for both the nominal case and the case in which additive disturbances are present.

II. SUCCESSIVE LINEARIZATION MPC

Consider a discrete time nonlinear system with state $x_i \in \mathbb{R}^{n_x}$ and control input $u_i \in \mathbb{R}^{n_u}$ at the i th time step, where

$$x_{i+1} = f(x_i, u_i), \quad i = 0, 1, \dots \quad (1)$$

and f is differentiable for all (x_i, u_i) in the operating region $\mathcal{X} \times \mathcal{U}$. We pose the problem of optimally controlling the system trajectories $\mathbf{x} : \{x_0, x_1, \dots\}$, $\mathbf{u} : \{u_0, u_1, \dots\}$ to track a given state and input reference trajectory $\mathbf{x}^r : \{x_0^r, x_1^r, \dots\}$, $\mathbf{u}^r : \{u_0^r, u_1^r, \dots\}$ where $x_{i+1}^r = f(x_i^r, u_i^r)$ for all $i \geq 0$ subject to a quadratic objective

$$\sum_{i=0}^{\infty} \left(\|x_i - x_i^r\|_Q^2 + \|u_i - u_i^r\|_R^2 \right) \quad (3)$$

and initial constraints $x_0 = x_{\text{init}}$ and inequality constraints

$$x_i \in \mathcal{X} \subset \mathbb{R}^{n_x}, \quad u_i \in \mathcal{U} \subset \mathbb{R}^{n_u}. \quad (5)$$

Here \mathcal{X} , \mathcal{U} are compact polytopic sets, x_{init} is a known initial condition and $\|x\|_Q^2 := x^T Q x$. We define the following receding horizon optimal control problem (RHOC):

$$\text{nMPC: } (\mathbf{u}_i^*, \mathbf{x}_i^*) := \arg \min_{\mathbf{u}_i, \mathbf{x}_i} J_{\text{nMPC}}(\mathbf{x}_i, \mathbf{u}_i) \quad (6a)$$

subject to, for $k = 0, \dots, N-1$:

$$x_{k|i} \in \mathcal{X} \quad (6b)$$

$$u_{k|i} \in \mathcal{U} \quad (6c)$$

$$x_{k+1|i} = f(x_{k|i}, u_{k|i}) \quad (6d)$$

$$x_{0|i} = x_i \quad (6e)$$

$$x_{N|i} \in \mathcal{X}_N \subseteq \mathcal{X}, \quad (6f)$$

with

$$J_{\text{nMPC}}(\mathbf{x}_i, \mathbf{u}_i) := \|x_{N|i} - x_{N+i}^r\|_{Q_N}^2 + \sum_{k=0}^{N-1} \left(\|x_{k|i} - x_{k+i}^r\|_Q^2 + \|u_{k|i} - u_{k+i}^r\|_R^2 \right), \quad (6g)$$

where $\mathbf{u}_i := \{u_{0|i}, \dots, u_{N-1|i}\}$ and $\mathbf{x}_i := \{x_{0|i}, \dots, x_{N|i}\}$. Here $x_{k|i}$, $u_{k|i}$ denote the state and control input at time $k+i$ predicted at time i . The matrices Q , R , Q_N and the terminal set \mathcal{X}_N are subject to the following assumptions:

Assumption 1. (a) x_i is known at each time i .
 (b) $x_i^r \in \mathcal{X}_N \subseteq \mathcal{X}$ and $u_i^r \in \mathcal{U}$, $\forall i \geq N$.
 (c) $f(x, \kappa_i(x)) \in \mathcal{X}_N$, $\forall x \in \mathcal{X}_N$, $\forall i \geq N$, where $\kappa_i(x) := K(x - x_i^r) + u_i^r$ satisfies $\kappa_i(x) \in \mathcal{U}$, $\forall x \in \mathcal{X}_N$, $\forall i \geq N$.
 (d) Q, R, Q_N are positive definite matrices, and (1) with $u_i = K(x_i - x_i^r) + u_i^r$ satisfies, $\forall x_i \in \mathcal{X}_N$, $\forall i \geq N$,

$$\|x_i - x_i^r\|_{Q_N}^2 - \|x_{i+1} - x_{i+1}^r\|_{Q_N}^2 \geq \|x_i - x_i^r\|_Q^2 + \|u_i - u_i^r\|_R^2.$$

In receding horizon control, at each discrete time step i the nMPC RHOC is solved to obtain an optimal control sequence $\mathbf{u}_i^* : \{u_{0|i}^*, u_{1|i}^*, \dots, u_{N-1|i}^*\}$ and the first element of this sequence is used to define a receding control law $u_i = u_{0|i}^*$. At the next time step a new RHOC is formulated using the measurement (or estimate) of x_{i+1} and the process is repeated. To guarantee stability of the control strategy it is essential that a feasible solution can be obtained within each discrete time step. However, the nMPC RHOC (6a)-(6g) is a nonlinear program and thus in general no guarantees can be given that either the optimal solution or a suboptimal feasible point is determinable within one discrete time step. To address these problems Section III proposes a successive linearization approach to reformulate the problem as a sequence of convex programs. Compared to other such approaches, the proposed method bounds linearization errors as functions of the predicted state tube that is optimized online. This removes the need to compute linearization error bounds prior to solving the RHOC and ensures that a fixed point of the successive approximation scheme corresponds to an optimal point of the nMPC RHOC. As shown in Section IV, it also provides guarantees of convergence to a fixed point, recursive feasibility and stability of the resulting MPC law.

III. PROPOSED CONTROL LAW

To simplify presentation we first strengthen the assumptions on the system model and later discuss how these assumptions can be relaxed to allow the proposed approach to be applied to systems with dynamics that are neither convex nor concave.

Assumption 2. The function $f : \mathcal{X} \times \mathcal{U} \rightarrow \mathbb{R}^{n_x}$ in (1) is differentiable and componentwise convex.

Given state and control sequences $\mathbf{x}_i^0, \mathbf{u}_i^0$, where $\mathbf{x}_i^0 = \{x_{0|i}^0, x_{1|i}^0, \dots, x_{N|i}^0\}$, $\mathbf{u}_i^0 = \{u_{0|i}^0, u_{1|i}^0, \dots, u_{N-1|i}^0\}$, which we refer to as state and control seed trajectories, the model (1) can be equivalently expressed as

$$x_{k+1|i} = x_{k+1|i}^0 + A_{k|i}s_{k|i} + B_{k|i}v_{k|i} + e(x_{k|i}, u_{k|i}) \quad (7)$$

$$s_{k|i} = x_{k|i} - x_{k|i}^0, \quad v_{k|i} = u_{k|i} - u_{k|i}^0 \quad (8)$$

$$A_{k|i} = \frac{\partial f}{\partial x_{k|i}} \Big|_{(x_{k|i}^0, u_{k|i}^0)}, B_{k|i} = \frac{\partial f}{\partial u_{k|i}} \Big|_{(x_{k|i}^0, u_{k|i}^0)} \quad (9)$$

and $e_{k|i}$ is the Jacobian linearization error

$$e_{k|i} = f(x_{k|i}, u_{k|i}) - f(x_{k|i}^0, u_{k|i}^0) - A_{k|i}s_{k|i} - B_{k|i}v_{k|i}. \quad (10)$$

Assumption 3. The initial seed trajectory $(\mathbf{x}^0, \mathbf{u}^0)$ is feasible (but potentially suboptimal) for constraints (6e)-(6f).

Anticipating the explicit treatment of disturbances in Section V, we assume a feedback control law:

$$u_{k|i} = u_{k|i}^0 + v_{k|i} = u_{k|i}^0 + Ks_{k|i} + c_{k|i}, \quad (11)$$

for a given feedback gain K , and consider the sequence $\mathbf{c}_i := \{c_{0|i}, \dots, c_{N-1|i}\}$ as an optimization variable instead of \mathbf{u}_i .

Assumption 2 requires the components of f to be convex so that the components of the linearization error $e_{k|i}$ in (10), denoted $[e_{k|i}(x_{k|i}, u_{k|i})]_j$, $j = 1, \dots, n_x$, are convex functions of $(x_{k|i}, u_{k|i}) \in (\mathcal{X}, \mathcal{U})$. Since $[e_{k|i}(x_{k|i}, u_{k|i})]_j$ is convex and the linearization point $(x_{k|i}^0, u_{k|i}^0)$ lies in $(\mathcal{X}, \mathcal{U})$ we have

$$\min_{(x_{k|i}, u_{k|i}) \in (\mathcal{X}, \mathcal{U})} [e_{k|i}(x_{k|i}, u_{k|i})]_j = 0, \quad (12)$$

and moreover a constraint of the form

$$[e_{k|i}(s_{k|i} + x_{k|i}^0, Ks_{k|i} + c_{k|i} + u_{k|i}^0)]_j \leq r$$

is convex in $s_{k|i}$, $c_{k|i}$ and r .

Let $\mathbf{S}_i = \{S_{0|i}, \dots, S_{N|i}\}$ denote a tube bounding the state perturbations $s_{k|i}$ in (7)-(9), where each set $S_{k|i}$ is defined in terms of optimization variables $\underline{s}_{k|i}, \bar{s}_{k|i} \in \mathbb{R}^{n_x}$ as

$$S_{k|i} = \{s : \underline{s}_{k|i} \leq s \leq \bar{s}_{k|i}\}. \quad (13)$$

Then at time i a convex RHOC can be formulated as

$$\text{cMPC: } (\mathbf{c}_i^*, \mathbf{S}_i^*) := \arg \min_{\mathbf{c}_i, \mathbf{S}_i} J_{\text{cMPC}}(\mathbf{c}_i, \mathbf{S}_i, \mathbf{x}_i^0, \mathbf{u}_i^0) \quad (14a)$$

subject to, for $k = 0, \dots, N-1$:

$$S_{k|i} \oplus \{x_{k|i}^0\} \subseteq \mathcal{X} \quad (14b)$$

$$KS_{k|i} \oplus \{c_{k|i} + u_{k|i}^0\} \subseteq \mathcal{U} \quad (14c)$$

$$[\bar{s}_{k+1|i}]_j \geq [A_{k|i} + B_{k|i}K]_j s + [B_{k|i}]_j c_{k|i} + [e_{k|i}(s + x_{k|i}^0, Ks + c_{k|i} + u_{k|i}^0)]_j$$

$$[\underline{s}_{k+1|i}]_j \leq [A_{k|i} + B_{k|i}K]_j s + [B_{k|i}]_j c_{k|i} \quad \text{for all } s \in S_{k|i}, j = 1, \dots, n_x \quad (14d)$$

$$S_{0|i} = \{0\} \quad (14e)$$

$$S_{N|i} \oplus \{x_{N|i}^0\} \subseteq \mathcal{X}_N \quad (14f)$$

with

$$J_{\text{cMPC}}(\mathbf{c}_i, \mathbf{S}_i, \mathbf{x}_i^0, \mathbf{u}_i^0) := \max_{s_{N|i} \in S_{N|i}} \|x_{N|i}^0 + s_{N|i} - x_{N+i}^r\|_{Q_N}^2 + \sum_{k=0}^{N-1} \max_{s_{k|i} \in S_{k|i}} \left[\|x_{k|i}^0 + s_{k|i} - x_{k+i}^r\|_Q^2 + \|u_{k|i}^0 + Ks_{k|i} + c_{k|i} - u_{k+i}^r\|_R^2 \right], \quad (14g)$$

where $[z]_j$, $[A]_j$ denote the j th element and j th row of vector z and matrix A . The inequalities in (14d) can be imposed for all $s \in S_{k|i}$ by replacing s with the 2^{n_x} vertices of $S_{k|i}$, which depend linearly on $\underline{s}_{k|i}, \bar{s}_{k|i}$. The minimization of the cost (14g) can be performed via an equivalent epigraph form with a linear objective subject to convex constraints on \mathbf{c}_i and $\{\underline{s}_{k|i}, \bar{s}_{k|i}, k = 0, \dots, N\}$ the vertices of $S_{k|i}$. The cMPC RHOC is convex and can be solved using, for example, interior-point methods or large-scale solvers such as the alternating direction method of multipliers (ADMM) [25, 26].

The feedback gain K can be chosen equal to the terminal feedback gain K , which is stabilizing within \mathcal{X}_N due to Assumption 1(d). The computation of K , \mathcal{X}_N , and Q_N is in general non-trivial for nonlinear systems. A scheme to

compute them while maximizing the volume of the terminal set using semidefinite programming can be found in [1, 27].

The cMPC strategy is summarized in Algorithm 1. At each discrete time step i , the algorithm first solves the cMPC RHOCOP to obtain \mathbf{c}_i^* given a feasible seed $(\mathbf{x}_i^0, \mathbf{u}_i^0)$ (lines 3-4), then computes the predicted state and control sequences $(\mathbf{x}_i^*, \mathbf{u}_i^*)$ using the model dynamics (1) with $\mathbf{c}_i = \mathbf{c}_i^*$ and $x_{0|i}^* = x_i$ (lines 5-6), then updates the seed trajectory $(\mathbf{x}_i^0, \mathbf{u}_i^0)$ (line 7) and repeats this iteration until either $\|\mathbf{c}_i^*\| < tol$, where tol is a specified threshold, or the limit $maxiters$ on the number of iterations is reached.

Algorithm 1: cMPC

- Offline:** Determine K , Q_N , \mathcal{X}_N , and an initial seed $(\mathbf{x}_0^0, \mathbf{u}_0^0)$ satisfying (6b)-(6f) with $x_0 = x_{init}$
- Online:** At each discrete time step $i = 0, 1, \dots$
- 1 Obtain x_i and set the iteration counter $n \leftarrow 1$;
 - 2 **while** $n \leq maxiters$ and $\|\mathbf{c}_i^*\| \geq tol$ **do**
 - 3 Compute $A_{k|i}$, $B_{k|i}$ for $k = 0, \dots, N-1$ using (9) and $(\mathbf{x}_i^0, \mathbf{u}_i^0)$;
 - 4 Solve cMPC RHOCOP to obtain $(\mathbf{c}_i^*, \mathbf{S}_i^*)$;
 - 5 **for** $k = 0, \dots, N-1$ **do**
 - 6 Compute $u_{k|i}^* = u_{k|i}^0 + c_{k|i}^* + K(x_{k|i}^* - x_{k|i}^0)$
and $x_{k+1|i}^* = f(x_{k|i}^*, u_{k|i}^*)$ with $x_{0|i}^* = x_i$;
 - 7 Update the seed trajectory: $(\mathbf{x}_i^0, \mathbf{u}_i^0) \leftarrow (\mathbf{x}_i^*, \mathbf{u}_i^*)$;
 - 8 $n \leftarrow n + 1$;
 - 9 Implement $u_i \leftarrow u_{0|i}^*$;
 - 10 Set $u_{N|i}^* = K(x_{N|i}^* - x_{N+i}^r) + u_{N+i}^r$ and
 $\mathbf{u}_{i+1}^0 \leftarrow \{u_{1|i}^*, \dots, u_{N-1|i}^*, u_{N|i}^*\}$
 $\mathbf{x}_{i+1}^0 \leftarrow \{x_{1|i}^*, \dots, x_{N|i}^*, f(x_{N|i}^*, u_{N|i}^*)\}$;
-

Remark 1. A time-varying feedback gain (such as the linear-quadratic optimal controller for the linear time-varying model computed on line 3 of Algorithm 1) may be used instead of a fixed feedback gain K in (11). This is likely to give better numerical conditioning and disturbance rejection since the fixed gain K is only required by Assumption 1 to be locally stabilizing within the terminal set \mathcal{X}_N .

Compared to other successive linearization approaches [4–6], the cMPC strategy does not require prior knowledge or online calculation of sets bounding linearization errors, and it avoids conservative approximate linear constraints by using tight bounds on the approximation error in (14d). In particular, the first inequality in (14d) is equivalent to $\bar{s}_{k+1|i,j} \geq [f(s_{k|i} + x_{k|i}^0, Ks_{k|i} + c_{k|i} + u_{k|i}^0)]_j - [f(x_{k|i}^0, u_{k|i}^0)]_j$, and the second is a tight bound due to (12). As shown in Section IV, this ensures that the tube computed by the cMPC iteration converges to a single trajectory, and hence a fixed point of the cMPC iteration is a local minimum for the nMPC RHOCOP. In addition, the maximum number of iterations can be chosen as any value satisfying $maxiters \geq 1$ without affecting the recursive feasibility and stability guarantees discussed in Section IV.

A procedure for obtaining an initial seed trajectory is suggested in [24]. This assumes knowledge of an initial trajectory satisfying constraints (6b)-(6d) and the initial condition (6e),

but not necessarily the terminal constraint (6f). The method updates this seed via a sequence of convex optimizations to obtain a feasible trajectory for the cMPC RHOCOP.

We propose an alternative method of computing offline an initial seed trajectory $(\mathbf{x}_0^0, \mathbf{u}_0^0)$ for Algorithm 1 that does not require knowledge of a feasible trajectory with initial condition x_{init} . We define an offline optimization of $(\mathbf{c}_0, \mathbf{S}_0)$ subject to the constraints (14b)-(14d), (14f), and $S_{0|0} = \{s\}$, where s is an additional optimization variable, and we minimize the distance of $x_{0|0}^0 + s$ from x_{init} by setting the objective to $\|x_{init} - x_{0|0}^0 - s\|$. Initially we set $\mathbf{x}_0^0 := \{x_N^r, \dots, x_{2N}^r\}$ and $\mathbf{u}_0^0 := \{u_N^r, \dots, u_{2N-1}^r\}$, since by Assumption 1 this choice ensures that constraints (14b)-(14d), (14f) and $S_{0|0} = \{s\}$ are feasible with $s = 0$. To obtain an initial seed for Algorithm 1, we iteratively solve this optimization and update $(\mathbf{x}_0^0, \mathbf{u}_0^0)$ as in lines 2-8 of Algorithm 1, with $x_{0|0}^* = x_0$ in line 6 replaced by $x_{0|0}^* = x_{0|0}^0 + s^*$, until either the optimal objective is equal to zero, or the optimal objective converges to a non-zero value, in which case the constraints should (if possible) be relaxed, for example by increasing N or enlarging \mathcal{X}_N .

Remark 2. Assumption 2 can be relaxed to allow functions f which are DC, i.e. they can be written as the difference of convex functions: $f(x, u) = g(x, u) - h(x, u)$ where $g, h : \mathcal{X} \times \mathcal{U} \rightarrow \mathbb{R}^{n_x}$ are differentiable and componentwise convex and the sets \mathcal{X}, \mathcal{U} are convex. In this case the error function has the following properties

$$\begin{aligned} e_{k|i} &= e_{k|i}^g - e_{k|i}^h \\ e_{k|i}^g &= g(x_{k|i}, u_{k|i}) - g(x_{k|i}^0, u_{k|i}^0) - A_{k|i}^g s_{k|i} - B_{k|i}^g v_{k|i} \\ e_{k|i}^h &= h(x_{k|i}, u_{k|i}) - h(x_{k|i}^0, u_{k|i}^0) - A_{k|i}^h s_{k|i} - B_{k|i}^h v_{k|i} \\ &\min_{(x_{k|i}, u_{k|i}) \in (\mathcal{X}, \mathcal{U})} [e_{k|i}^g(x_{k|i}, u_{k|i})]_j = 0 \\ &\max_{(x_{k|i}, u_{k|i}) \in (\mathcal{X}, \mathcal{U})} -[e_{k|i}^h(x_{k|i}, u_{k|i})]_j = 0 \end{aligned}$$

where $e_{k|i}^g, e_{k|i}^h$ are componentwise convex and

$$\begin{aligned} A_{k|i}^g &= \frac{\partial g}{\partial x_{k|i}} \Big|_{(x_{k|i}^0, u_{k|i}^0)}, \quad B_{k|i}^g = \frac{\partial g}{\partial u_{k|i}} \Big|_{(x_{k|i}^0, u_{k|i}^0)} \\ A_{k|i}^h &= \frac{\partial h}{\partial x_{k|i}} \Big|_{(x_{k|i}^0, u_{k|i}^0)}, \quad B_{k|i}^h = \frac{\partial h}{\partial u_{k|i}} \Big|_{(x_{k|i}^0, u_{k|i}^0)} \end{aligned}$$

Thus Algorithm 1 can be used with constraints (14d) in step 4 replaced, for all $s \in S_{k|i}$, $j = 1, \dots, n_x$, by

$$\begin{aligned} [\bar{s}_{k+1|i}]_j &\geq [A_{k|i} + B_{k|i}K]_j s + [B_{k|i}]_j c_{k|i} \\ &\quad + [e_{k|i}^g(s + x_{k|i}^0, Ks + c_{k|i} + u_{k|i}^0)]_j \\ [\underline{s}_{k+1|i}]_j &\leq [A_{k|i} + B_{k|i}K]_j s + [B_{k|i}]_j c_{k|i} \\ &\quad - [e_{k|i}^h(s + x_{k|i}^0, Ks + c_{k|i} + u_{k|i}^0)]_j. \end{aligned}$$

Remark 3. If the cMPC RHOCOP (14) is implemented by replacing s with the 2^{n_x} vertices of $S_{k|i}$ (which depend linearly on $\underline{s}_{k|i}$, $\bar{s}_{k|i}$), then the number of constraints in (14d) will grow exponentially with the state dimension n_x . However, this exponential growth can be avoided by using homothetic tubes (see e.g. [1]) to bound the state perturbations $s_{k|i}$. In this formulation the tube description in (13) is replaced with

$$S_{k|i} = \{z_{k|i}^0\} \oplus \alpha_{k|i} \text{co}\{z^1, \dots, z^{n_x}\},$$

where co denotes the convex hull. Here $z_{k|i}^0 \in \mathbb{R}^{n_x}$ and $\alpha_{k|i} \in \mathbb{R}$ are optimization variables in the RHOC and the vertices $\{z^1, \dots, z^{n_v}\}$ are fixed by the designer. This reduces the number of optimization variables at each time step from $2n_x$ to $n_x + 1$ and reduces the number of constraints in (14d) to $2mn_x$. Here n_v can be as small as $n_x + 1$ if $S_{k|i}$ is chosen to be a simplex, allowing the number of constraints in the cMPC RHOC to depend quadratically on the state dimension n_x . This is discussed in more detail in Remark 9 in Section V.

IV. FEASIBILITY AND STABILITY GUARANTEES

In this section we demonstrate that Algorithm 1 is recursively feasible and provides a feasible solution for the nMPC RHOC. We further show that the optimal cMPC RHOC objective value is an upper bound on the cost of the nMPC RHOC, and that the optimal cost is non-increasing at successive iterations (lines 2-8 of Algorithm 1) and at successive time steps. Using these properties, we show that the iteration converges (if the iteration limit *maxiters* is sufficiently large) to a solution satisfying the first-order optimality conditions of the nMPC RHOC, and that (for any value of *maxiters*) the control law of Algorithm 1 is asymptotically stable.

Theorem 1 (Recursive feasibility). *If Assumptions 1 and 2 hold and the initial seed trajectory $(\mathbf{x}_0^0, \mathbf{u}_0^0)$ satisfies (6b)-(6f), then the cMPC RHOC is feasible at each iteration of Algorithm 1, at all times $i \geq 0$.*

Proof. Let $\mathbf{c}_i = 0$ and $\mathbf{S}_i = \{\{0\}, \dots, \{0\}\}$, then the constraints of the cMPC RHOC in line 4 of Alg. 1 reduce to $\{x_{k|i}^0\} \subseteq \mathcal{X}$, $u_{k|i}^0 \in \mathcal{U}$ and $\{x_{N|i}^0\} \subseteq \mathcal{X}_N$. These conditions are satisfied if $(\mathbf{x}_i, \mathbf{u}_i) = (\mathbf{x}_i^0, \mathbf{u}_i^0)$ satisfies (6b)-(6f), which is true by assumption for the initial seed trajectory $(\mathbf{x}_0^0, \mathbf{u}_0^0)$, and is true for the seed trajectories $(\mathbf{x}_i^0, \mathbf{u}_i^0)$ and $(\mathbf{x}_{i+1}^0, \mathbf{u}_{i+1}^0)$ in lines 7 and 10 of Alg. 1 due to constraints (14b)-(14f) and the properties of \mathcal{X}_N (Assumption 1(b) and (c)). \square

We next consider how the optimal cMPC RHOC cost relates to the cost associated with the seed trajectory, and hence prove that the iteration converges to a solution that is at least locally optimal for the nMPC RHOC. At the i th time step and n th iteration of Algorithm 1, let $(\mathbf{x}_i^{0,n}, \mathbf{u}_i^{0,n})$ denote the seed trajectory in line 3, let $(\mathbf{c}_i^{*,n}, \mathbf{S}_i^{*,n})$ denote the optimal solution of the cMPC RHOC in line 4 and define $J_{\text{cMPC},i}^{*,n} := J_{\text{cMPC}}(\mathbf{c}_i^{*,n}, \mathbf{S}_i^{*,n}, \mathbf{x}_i^{0,n}, \mathbf{u}_i^{0,n})$.

Lemma 1. *For all $i \geq 0$ and for all $n \geq 1$, we have $J_{\text{nMPC}}(\mathbf{x}_i^{0,n+1}, \mathbf{u}_i^{0,n+1}) \leq J_{\text{cMPC},i}^{*,n} \leq J_{\text{nMPC}}(\mathbf{x}_i^{0,n}, \mathbf{u}_i^{0,n})$.*

Proof. Let $\mathbf{c}_i = 0$ and $\mathbf{S}_i = \{\{0\}, \dots, \{0\}\}$, then

$$J_{\text{cMPC}}(0, \{\{0\}, \dots, \{0\}\}, \mathbf{x}_i^{0,n}, \mathbf{u}_i^{0,n}) = J_{\text{nMPC}}(\mathbf{x}_i^{0,n}, \mathbf{u}_i^{0,n}),$$

and since this is a feasible (possibly suboptimal) choice, the optimal cMPC cost in line 4 of Algorithm 1 satisfies

$$J_{\text{cMPC}}(\mathbf{c}_i^*, \mathbf{S}_i^*, \mathbf{x}_i^{0,n}, \mathbf{u}_i^{0,n}) \leq J_{\text{nMPC}}(\mathbf{x}_i^{0,n}, \mathbf{u}_i^{0,n}).$$

Furthermore, the updated seed in lines 5-7 satisfies

$$J_{\text{nMPC}}(\mathbf{x}_i^{0,n+1}, \mathbf{u}_i^{0,n+1}) \leq J_{\text{cMPC}}(\mathbf{c}_i^*, \mathbf{S}_i^*, \mathbf{x}_i^{0,n}, \mathbf{u}_i^{0,n})$$

since the tube \mathbf{S}_i^* necessarily contains $\mathbf{x}_i^{0,n+1}$. \square

Theorem 2. *For all $i \geq 0$, the iteration on lines 2-8 of Algorithm 1 converges to a seed trajectory $(\mathbf{x}_i^{0,n}, \mathbf{u}_i^{0,n})$ such that $\mathbf{c}_i^{*,n} = 0$ and $\mathbf{S}_i^{*,n} = \{\{0\}, \dots, \{0\}\}$ is an optimal solution of the cMPC RHOC in the limit as $n \rightarrow \infty$.*

Proof. From Lemma 1 we have $J_{\text{cMPC},i}^{*,n+1} \leq J_{\text{cMPC},i}^{*,n}$ for all n , so $J_{\text{cMPC},i}^{*,n} \geq 0$ implies $J_{\text{cMPC},i}^{*,n} - J_{\text{cMPC},i}^{*,n+1} \rightarrow 0$ and hence $J_{\text{cMPC},i}^{*,n} - J_{\text{nMPC}}(\mathbf{x}_i^{0,n}, \mathbf{u}_i^{0,n}) \rightarrow 0$ as $n \rightarrow \infty$. Therefore $(\mathbf{c}_i^{*,n}, \mathbf{S}_i^{*,n}) = (0, \{\{0\}, \dots, \{0\}\})$ is an optimal solution of the cMPC RHOC in the limit as $n \rightarrow \infty$. \square

Remark 4. *The optimal solution of the cMPC RHOC for \mathbf{c}_i^* is unique since R is positive definite. Therefore Theorem 2 implies $\mathbf{c}_i^{*,n} \rightarrow 0$ as $n \rightarrow \infty$, justifying the termination criterion in line 2 of Alg. 1. Moreover, even if the optimal \mathbf{S}_i^* is non-unique, we can choose $\mathbf{S}_i^{*,n} = 0$ if $\mathbf{c}_i^{*,n} = 0$ and thus ensure that the iteration converges to a tube containing a single trajectory.*

Theorem 3. *The iteration of Algorithm 1 converges to a local minimum of the nMPC RHOC (6a)-(6g).*

A proof of Theorem 3 can be found in the Appendix.

Remark 5. *The converse of Theorem 3 is also true: a local minimum of the nMPC problem is a local minimum of the cMPC RHOC (14a)-(14g) due to Theorem 2.*

Theorem 4. *Given Assumptions 1 and 2, the control law of Algorithm 1 ensures that $x - x^r = 0$ is an asymptotically stable equilibrium of the system (1)¹, with region of attraction consisting of the initial conditions x_0 of (1) for which there exists a control sequence such that $u_k \in \mathcal{U}$ and $x_k \in \mathcal{X}$ for $k = 0, \dots, N - 1$, and $x_N \in \mathcal{X}_N$.*

Proof. Assumption 1(d) and the update policy in line 10 of Algorithm 1 imply that the cost associated with the seed trajectory at the first iteration at time $i + 1$ satisfies

$$J_{\text{nMPC}}(\mathbf{x}_{i+1}^{0,1}, \mathbf{u}_{i+1}^{0,1}) \leq J_{\text{nMPC}}(\mathbf{x}_i^{0,n_i}, \mathbf{u}_i^{0,n_i}) - (\|x_i - x_i^r\|_Q^2 + \|u_i - u_i^r\|_R^2),$$

where n_i is the final iteration count at time i . Hence

$$J_{\text{nMPC}}(\mathbf{x}_{i+1}^{0,n_{i+1}}, \mathbf{u}_{i+1}^{0,n_{i+1}}) \leq J_{\text{nMPC}}(\mathbf{x}_i^{0,n_i}, \mathbf{u}_i^{0,n_i}) - (\|x_i - x_i^r\|_Q^2 + \|u_i - u_i^r\|_R^2)$$

due to Lemma 1, and, since Q, R, Q_N are positive definite, $J_{\text{nMPC}}(\mathbf{x}_{i+1}^{0,n_i}, \mathbf{u}_{i+1}^{0,n_i})$ is therefore a Lyapunov function demonstrating stability of $x - x^r = 0$. Summing both sides of this inequality over $i = 0, \dots, t$ we obtain

$$\sum_{i=0}^{\infty} (\|x_i - x_i^r\|_Q^2 + \|u_i - u_i^r\|_R^2) \leq J_{\text{nMPC}}(\mathbf{x}_0^{0,n_1}, \mathbf{u}_0^{0,n_1})$$

and it follows that $(x_i, u_i) \rightarrow (x_i^r, u_i^r)$ as $i \rightarrow \infty$. \square

Theorem 3 and Remark 5 show that the cMPC RHOC formulation allows Algorithm 1 to effectively approximate the optimal control law by solving a sequence of convex programs.

¹More precisely, $x^\delta = x - x^r = 0$ is an asymptotically stable equilibrium of the system $x_{i+1}^\delta = f(x_i^\delta + x_i^r, u_i) - f(x_i^r, u_i^r)$.

V. ROBUST MPC WITH ADDITIVE DISTURBANCE

This section considers uncertain systems of the form

$$x_{i+1} = f(x_i, u_i) + w_i, \quad w_i \in \mathcal{W} \quad (15)$$

where w_i is an unknown disturbance input and $\mathcal{W} \subset \mathbb{R}^{n_x}$ is a known compact set. We consider an optimal tracking control problem similar to (6), with the model (15) replacing (1). In this setting we retain Assumption 2 (we discuss later how this can be relaxed to allow nonconvex dynamics), and we replace Assumption 1 with the following assumption.

Assumption 4. (a) x_i is known at each time i .

(b) $\mathcal{X}_N \subseteq \mathbb{R}^{2n_x}$ is known such that, for all $(\underline{x}_N, \bar{x}_N) \in \mathcal{X}_N$ and all $x \in X(\underline{x}_N, \bar{x}_N) := \{x : \underline{x}_N \leq x \leq \bar{x}_N\}$ we have:

(i) $x_i^r \in X(\underline{x}_N, \bar{x}_N)$ and $u_i^r \in \mathcal{U} \forall i \geq N$, (ii) $X(\underline{x}_N, \bar{x}_N) \subseteq \mathcal{X}$ and $(\underline{x}_N^+, \bar{x}_N^+) \in \mathcal{X}_N$ where, $\forall i \geq N$,

$$\begin{aligned} [\bar{x}_N^+]_j &\geq [f(x, \kappa_i(x))]_j + [\bar{w}]_j \\ [\underline{x}_N^+]_j &\leq [f(x, \kappa_i(x))]_j + [\underline{w}]_j, \quad j = 1, \dots, n_x, \end{aligned}$$

and (iii) $\kappa_i(x) := K(x - x_i^r) + u_i^r \in \mathcal{U}, \forall i \geq N$.

(c) $Q, R, Q_N \succ 0$ and (15) with $u_i = K(x_i - x_i^r) + u_i^r$ satisfies, for all $i \geq N$, for some $\beta \geq 0$ and some $(\underline{x}_N, \bar{x}_N) \in \mathcal{X}_N$,

$$\begin{aligned} \|x_i - x_i^r\|_{Q_N}^2 - \|x_{i+1} - x_{i+1}^r\|_{Q_N}^2 \\ \geq \|x_i - x_i^r\|_Q^2 + \|u_i - u_i^r\|_R^2 - \beta, \quad \forall x_i \in X(\underline{x}_N, \bar{x}_N). \end{aligned}$$

The uncertainty in the future disturbance sequence implies that future state trajectories generated by (15) for a given control sequence or control law are uncertain. Therefore, although a convex successive optimization approach can again be used to define an MPC law, the optimal tube bounding predicted model states cannot converge to a single trajectory as in the nominal case considered in Section IV. In addition, the seed trajectory used to define a linear model approximation in the disturbance-free case must be replaced by a tube.

A. Proposed robust control law

Recursive feasibility requires that at each iteration a feasible seed tube can be constructed, guaranteeing that the state and control constraints are satisfied under all possible realizations of the disturbance input in (15). Given an initial state $x_{0|i}$ and perturbation sequence \mathbf{c}_i^0 , we propose the method summarized in Algorithm 2 for constructing a linear model approximation and a seed tube $\mathbf{X}_i^0 := \{X_{0|i}^0, \dots, X_{N|i}^0\}$, where

$$X_{k|i}^0 = \{x : \underline{x}_{k|i} \leq x \leq \bar{x}_{k|i}\}. \quad (16)$$

The linearization points, denoted $(x_{k|i,j}^0, u_{k|i,j}^0)$, are computed for each component, $[f]_j, j = 1, \dots, n_x$, of f , via

$$x_{k|i,j}^0 = \arg \min_{x \in X_{k|i}^0} [f(x, Kx + c_{k|i}^0)]_j \quad (17)$$

and $u_{k|i,j}^0 = Kx_{k|i,j}^0 + c_{k|i}^0$. We show in Section V-B that this choice ensures recursive feasibility. The seed tube is computed using elementwise bounds, for $j = 1, \dots, n_x$,

$$[\bar{x}_{k+1|i}]_j = \max_{x \in X_{k|i}^0} [f(x, Kx + c_{k|i}^0)]_j + [\bar{w}]_j \quad (18)$$

$$[\underline{x}_{k+1|i}]_j = [f(x_{k|i,j}^0, u_{k|i,j}^0)]_j + [\underline{w}]_j \quad (19)$$

where

$$[\bar{w}]_j = \max_{w \in \mathcal{W}} [w]_j, \quad [\underline{w}]_j = \min_{w \in \mathcal{W}} [w]_j.$$

Note that the minimization in (17) is a convex problem, and the maximization in (18) can be performed as a maximization over the vertices of $X_{k|i}^0$, each of which is a known linear function of $\underline{x}_{k|i}, \bar{x}_{k|i}$.

The linear model approximation computed using Algorithm 2 can be used to formulate the following robust RHOC.

$$\text{RcMPC: } (\mathbf{c}_i^*, \mathbf{X}_i^*) := \arg \min_{\mathbf{c}_i, \mathbf{X}_i} J_{\text{RcMPC}}(\mathbf{c}_i, \mathbf{X}_i, \mathbf{c}_i^0) \quad (20a)$$

subject to, for $k = 0, \dots, N-1$:

$$X_{k|i} \subseteq \mathcal{X} \quad (20b)$$

$$KX_{k|i} \oplus \{c_{k|i} + c_{k|i}^0\} \subseteq \mathcal{U}, \quad (20c)$$

$$\begin{aligned} [\bar{x}_{k+1|i}]_j &\geq [f(x, Kx + c_{k|i} + c_{k|i}^0)]_j + [\bar{w}]_j \\ [\underline{x}_{k+1|i}]_j &\leq [f(x_{k|i,j}^0, u_{k|i,j}^0)]_j + [A_{k|i} + B_{k|i}K]_j(x - x_{k|i,j}^0) \\ &\quad + [B_{k|i}]_j c_{k|i} + [\underline{w}]_j \\ &\text{for all } x \in X_{k|i}, \quad j = 1, \dots, n_x \end{aligned} \quad (20d)$$

$$X_{0|i} = \{x_i\} \quad (20e)$$

$$(\underline{x}_{N|i}, \bar{x}_{N|i}) \in \mathcal{X}_N \quad (20f)$$

with

$$\begin{aligned} J_{\text{RcMPC}}(\mathbf{c}_i, \mathbf{X}_i, \mathbf{c}_i^0) &:= \max_{x_{N|i} \in X_{N|i}} \|x_{N|i} - x_{N+i}^r\|_{Q_N}^2 + \\ &\sum_{k=0}^{N-1} \max_{x_{k|i} \in X_{k|i}} \left[\|x_{k|i} - x_{k+i}^r\|_Q^2 + \|Kx_{k|i} + c_{k|i} + c_{k|i}^0 - u_{k+i}^r\|_R^2 \right]. \end{aligned} \quad (20g)$$

The constraints in (20d) can be imposed for all $x \in X_{k|i}$ by replacing x with the 2^{n_x} vertices of $X_{k|i}$, which are linear functions of the optimization variables $\underline{x}_{k|i}, \bar{x}_{k|i}$. Hence the RcMPC RHOC can be expressed as a convex program similar to the cMPC RHOC for the nominal case.

Algorithm 2: Seed tube and linearization

Require: Initial state x_i and seed \mathbf{c}_i^0

1 Set $\bar{x}_{0|i} = \underline{x}_{0|i} = x_i$ so that $X_{0|i}^0 \leftarrow \{x_i\}$;

2 **for** $k = 0, \dots, N-1$ **do**

3 **for** $j = 1, \dots, n_x$ **do**

4 Solve problem (17) for $x_{k|i,j}^0$ and

$$u_{k|i,j}^0 = Kx_{k|i,j}^0 + c_{k|i}^0;$$

5 Compute $[A_{k|i}]_j = \frac{\partial [f]_j}{\partial x} \Big|_{(x_{k|i,j}^0, u_{k|i,j}^0)}$ and

$$[B_{k|i}]_j = \frac{\partial [f]_j}{\partial u} \Big|_{(x_{k|i,j}^0, u_{k|i,j}^0)};$$

6 Solve (18)-(19) for $[\bar{x}_{k+1|i}]_j, [\underline{x}_{k+1|i}]_j$;

7 Set $X_{k+1|i}^0 \leftarrow \{x : \underline{x}_{k+1|i} \leq x \leq \bar{x}_{k+1|i}\}$;

Remark 6. The feedback law $Kx_{k|i} + c_{k|i} + c_{k|i}^0$ can be interpreted as a controller of the form $Ks_{k|i} + c_{k|i} + v_{k|i}^0$, where $s_{k|i} = x_{k|i} - \hat{x}_{k|i}$, $v_{k|i}^0 = K\hat{x}_{k|i} + c_{k|i}^0$, and $\hat{x}_{k|i}$ is a point in $X_{k|i}^0$ such as the Chebyshev centre of $X_{k|i}^0$, for example. However, this equivalent formulation would require the additional unnecessary computation of $\hat{x}_{k|i}$ for $k = 0, \dots, N$.

The robust cMPC strategy is presented in Algorithm 3. The initial seed \mathbf{c}_0^0 can be computed offline similarly to the initialization of Algorithm 1. Starting with $x_0 := x_N^r$ and $\mathbf{c}_0^0 := \{u_N^r - Kx_N^r, \dots, u_{2N-1}^r - Kx_{2N-1}^r\}$, the tube \mathbf{X}_0^0 generated by Algorithm 2 satisfies $X_{k|i}^0 \subseteq \mathcal{X}_N$ for all k (by Assumption 4), and hence (20b)-(20d), (20f) are feasible with $(\mathbf{c}_0, \mathbf{X}_0) = (0, \mathbf{X}_0^0)$. To generate an initial seed for Algorithm 3, we perform the iteration in lines 2-5 of Algorithm 3, with line 4 replaced by the minimization of $\|x_{\text{init}} - x_0\|$ over variables $(\mathbf{c}_0, \mathbf{X}_0)$ and x_0 subject to (20b)-(20d), $X_{0|0}^0 = \{x_0\}$ and (20f), with $x_0 \leftarrow x_0^*$, $\mathbf{c}_0^0 \leftarrow \mathbf{c}_0^* + \mathbf{c}_0^0$ in line 5. This converges either to a feasible \mathbf{c}_0^0 with $x_0^* = x_{\text{init}}$, or to a non-zero value of $\|x_{\text{init}} - x_0^*\|$ (in which case the constraints should be relaxed, e.g. by increasing N or enlarging \mathcal{X}_N).

Algorithm 3: RcMPC

Offline: Determine K , Q_N , \mathcal{X}_N , and initial seed \mathbf{c}_0^0 with which Algorithm 2 generates a tube \mathbf{X}_0^0 such that $(\mathbf{c}_0, \mathbf{X}_0) = (0, \mathbf{X}_0^0)$ satisfies 20b-(20f) with $x_0 = x_{\text{init}}$

Online: At each discrete time step $i = 0, 1, \dots$:

- 1 Obtain x_i and set the iteration counter $n \leftarrow 1$;
 - 2 **while** $n \leq \text{maxiters}$ and $\|\mathbf{c}_i^*\| \geq \text{tol}$ **do**
 - 3 Compute $(x_{k|i,j}^0, u_{k|i,j}^0)$, $j = 1, \dots, n_x$ and $A_{k|i}$, $B_{k|i}$ for $k = 0, \dots, N-1$ using Alg. 2;
 - 4 Solve RcMPC RHOCPC to obtain $(\mathbf{c}_i^*, \mathbf{X}_i^*)$;
 - 5 Update the seed: $\mathbf{c}_i^0 \leftarrow \mathbf{c}_i^* + \mathbf{c}_i^0$, $n \leftarrow n + 1$;
 - 6 Implement $u_i \leftarrow Kx_i + c_{0|i}^0$ and set $\mathbf{c}_{i+1}^0 \leftarrow \{c_{1|i}^0, \dots, c_{N-1|i}^0, u_{N+i}^r - Kx_{N+i}^r\}$
-

Remark 7. If $\mathcal{W} = \{0\}$, then $(x_{k|i,j}^0, u_{k|i,j}^0) = (x_{k|i,1}^0, u_{k|i,1}^0)$ for all $j = 2, \dots, n_x$, so at each time step k all linearization points are identical. In this case the tube \mathbf{X}_i^* generated by Algorithm 3 contains a single trajectory, which is identical to \mathbf{x}_i^* at each iteration of Algorithm 1. Thus Algorithm 1 is a special case of Algorithm 3 when disturbances are absent.

Various methods are available for computing the terminal set \mathcal{X}_N , feedback gain K and matrix Q_N satisfying Assumption 4. The set \mathcal{X}_N can be chosen for example as the the maximum robust positively invariant (MRPI) set or as a sub-optimal (non-maximal) terminal set which satisfies Assumption 4. Methods for computing K , Q_N , and the MRPI offline, including a constraint-checking approach, can be found in [1, 28–30]

Remark 8. Analogously to the disturbance-free case in Remark 2, Assumption 2 can be relaxed to consider DC functions $f(x, u) = g(x, u) - h(x, u)$, where $g, h : \mathcal{X} \times \mathcal{U} \rightarrow \mathbb{R}^{n_x}$ are differentiable and componentwise convex on the convex sets \mathcal{X}, \mathcal{U} . In this case Algorithms 2 and 3 can still be used, with (17) in step 4 of Algorithm 2 replaced, for $\phi = g, h$ by

$$x_{k|i,j}^{0,\phi} = \arg \min_{x \in X_{k|i}^0} [\phi(x, Kx + c_{k|i}^0)]_j,$$

and with (18), (19) in step 6 replaced by

$$[\bar{x}_{k+1|i}]_j = \max_{x \in X_{k|i}^0} [g(x, Kx + c_{k|i}^0)]_j - [h(x_{k|i,j}^{0,h}, u_{k|i,j}^{0,h})]_j + [\bar{w}]_j$$

$$[\underline{x}_{k+1|i}]_j = [g(x_{k|i,j}^{0,g}, u_{k|i,j}^{0,g})]_j - \min_{x \in X_{k|i}^0} [h(x, Kx + c_{k|i}^0)]_j + [\underline{w}]_j$$

and with (20d) in the RcMPC RHOCPC replaced by

$$\begin{aligned} [\bar{x}_{k+1|i}]_j &\geq [g(x, Kx + c_{k|i} + c_{k|i}^0)]_j - [h(x_{k|i,j}^{0,h}, u_{k|i,j}^{0,h})]_j \\ &\quad - [A_{k|i}^h + B_{k|i}^h K]_j (x - x_{k|i,j}^{0,h}) - [B_{k|i}^h]_j c_{k|i} + [\bar{w}]_j \\ [\underline{x}_{k+1|i}]_j &\leq [g(x_{k|i,j}^{0,g}, u_{k|i,j}^{0,g})]_j + [A_{k|i}^g + B_{k|i}^g K]_j (x - x_{k|i,j}^{0,g}) \\ &\quad + [B_{k|i}^g]_j c_{k|i} - [h(x, Kx + c_{k|i} + c_{k|i}^0)]_j + [\underline{w}]_j. \end{aligned}$$

Remark 9. The RcMPC RHOCPC can be implemented by replacing x in (20d) by each vertex of $X_{k|i}$. If $X_{k|i}$ is defined by elementwise bounds, as in (16), the number of constraints will grow exponentially with the state dimension n_x . However, this exponential growth can be avoided if \mathbf{X}_i is defined instead as a homothetic tube. For example, any closed convex polytopic tube cross-section can be expressed

$$X_{k|i} := \{x : Qx \leq q_{k|i}\} = \text{co}\{Z^v q_{k|i}, v = 1, \dots, n_v\},$$

where $q_{k|i}$ is a variable, while Q, Z^1, \dots, Z^{n_v} are fixed and co denotes the convex hull. Writing $Qf(x, u) = \tilde{g}(x, u) - \tilde{h}(x, u)$, where \tilde{g}, \tilde{h} are differentiable and componentwise convex, we define $x_{k|i,j}^{0,\tilde{g}}$ as in Remark 8 and replace (20d) by

$$\begin{aligned} [q_{k+1|i}]_j &\geq [\tilde{g}(x, Kx + c_{k|i} + c_{k|i}^0)]_j - [\tilde{h}(x_{k|i,j}^{0,\tilde{g}}, u_{k|i,j}^{0,\tilde{h}})]_j \\ &\quad - [A_{k|i}^{\tilde{h}} + B_{k|i}^{\tilde{h}} K]_j (x - x_{k|i,j}^{0,\tilde{g}}) - [B_{k|i}^{\tilde{h}}]_j c_{k|i} + \max_{w \in \mathcal{W}} [Q]_j w \end{aligned} \quad (21)$$

with $x = Z^v q_{k|i}$, $v = 1, \dots, n_v$. This is identical to (20d) if $Q = [I \ -I]^\top$, $q_{k|i} = [\bar{x}_{k|i}^\top \ -\underline{x}_{k|i}^\top]^\top$, but if $Q = [-I \ 1 \dots 1]^\top$, we have simplex cross-sections with only $n_v = n_x + 1$ vertices.

B. Robust feasibility and stability guarantees

This section shows that, given an initial perturbation sequence \mathbf{c}_0^0 such that Algorithm 2 with $x_0 = x_{\text{init}}$ generates a feasible tube, the RcMPC RHOCPC remains feasible at each iteration of Algorithm 3 at all times. We also show that the optimal objective of the RcMPC RHOCPC is non-increasing, and therefore the iteration of Algorithm 3 converges and the MPC law provides a robust quadratic stability guarantee for any limit on the number of iterations satisfying $\text{maxiters} \geq 1$.

To make notation more precise, at the i th time step and n th iteration of Algorithm 3 we denote the seed trajectory as $\mathbf{c}_i^{0,*n}$ and the tube generated by Algorithm 2 with $\mathbf{c}_i^0 = \mathbf{c}_i^{0,*n}$ as $\mathbf{X}_i^{0,*n}$. Similarly, we denote the solution of the RcMPC RHOCPC in line 4 as $(\mathbf{c}_i^{*,n}, \mathbf{X}_i^{*,n})$, and the optimal objective as $J_{\text{RcMPC},i}^{*,n} = J_{\text{RcMPC}}(\mathbf{c}_i^{*,n}, \mathbf{X}_i^{*,n}, \mathbf{c}_i^{0,*n})$.

First note that the tubes $\mathbf{X}_i^{0,*n}$ and $\mathbf{X}_i^{*,n}$ generated by Algorithms 2 and 3 necessarily contain all trajectories of (15) with initial condition $x_{0|i} = x_i$ under the control laws $u_{k|i} = Kx_{k|i} + c_{k|i}^{0,*n}$ and $u_{k|i} = Kx_{k|i} + c_{k|i}^{*,n}$ respectively. This follows from the convexity of f (Assumption 2), and from the definition of $X_{k+1|i}^{0,*n}$ in (18)-(19), and the constraints on $X_{i+1|k}^{*,n}$ in (20d), which imply

$$f(x, Kx + c_{k|i}^{0,*n}) + w \in X_{k+1|i}^{0,*n} \quad \forall x \in X_{k|i}^{0,*n} \quad (22)$$

$$f(x, Kx + c_{k|i}^{*,n}) + w \in X_{k+1|i}^{*,n} \quad \forall x \in X_{k|i}^{*,n} \quad (23)$$

for all $w \in \mathcal{W}$, $k = 0, \dots, N-1$ and $i \geq 0$, where $\mathbf{c}_i^{0,n} = \{c_{0|i}^{0,n}, \dots, c_{N-1|i}^{0,n}\}$ and $\mathbf{c}_i^{*,n} = \{c_{0|i}^{*,n}, \dots, c_{N-1|i}^{*,n}\}$.

The recursive feasibility of Algorithm 3 is a consequence of the following properties of the sets $X_{i|k}^{0,n}$, $X_{i|k}^{*,n}$.

Lemma 2. For all $i \geq 0$ and $n = 1, \dots, n_i$ we have

$$\text{if } X_{k|i}^{0,n+1} \subseteq X_{k|i}^{*,n} \quad \text{then } X_{k+1|i}^{0,n+1} \subseteq X_{k+1|i}^{*,n} \quad (24)$$

$$\text{if } X_{k+1|i}^{0,1} \subseteq X_{k+1|i}^{*,n_i} \quad \text{then } X_{k+1|i+1}^{0,1} \subseteq X_{k+2|i}^{*,n_i} \quad (25)$$

$$\text{if } X_{N-1|i+1}^{0,1} \subseteq X_{N|i}^{*,n_i} \quad \text{then } X_{N|i+1}^{0,1} \subseteq \mathcal{X}_N \quad (26)$$

where (24) holds for $k = 0, \dots, N-1$, (25) holds for $k = 0, \dots, N-2$, and n_i is the final iteration at time i .

Proof. Condition (24) follows from the definition of the linearization points in (17), which implies

$$[f(x_{k|i,j}^{0,n+1}, u_{k|i,j}^{0,n+1})]_j = \min_{x \in X_{k|i}^{0,n+1}} [f(x, Kx + c_{k|i}^{0,n+1})]_j \quad (27)$$

$$\geq \min_{x \in X_{k|i}^{*,n}} [f(x, Kx + c_{k|i}^{0,n+1})]_j \quad (28)$$

$$\geq \min_{x \in X_{k|i}^{*,n}} \left\{ [f(x_{k|i,j}^{0,n}, u_{k|i,j}^{0,n})]_j + [A_{k|i}^n]_j (x - x_{k|i,j}^{0,n}) + [B_{k|i}^n]_j (Kx + c_{k|i}^{0,n+1} - u_{k|i,j}^{0,n}) \right\} \quad (29)$$

$$= \min_{x \in X_{k|i}^{*,n}} \left\{ [f(x_{k|i,j}^{0,n}, u_{k|i,j}^{0,n})]_j + [A_{k|i}^n + B_{k|i}^n K]_j (x - x_{k|i,j}^{0,n}) + [B_{k|i}^n]_j c_{k|i}^{*,n} \right\} \quad (30)$$

where (28) holds if $X_{k|i}^{0,n+1} \subseteq X_{k|i}^{*,n}$, (29) is due to the convexity of f , and (30) follows from $u_{k|i}^{0,n} = Kx_{k|i}^{0,n} + c_{k|i}^{0,n}$ and $c_{k+1|i}^{0,n+1} = c_{k|i}^{*,n} + c_{k|i}^{0,n}$. Therefore, from (20d) we have $\underline{x}_{k+1|i}^{0,n+1} \geq \underline{x}_{k+1|i}^{*,n}$. Similarly, if $X_{k|i}^{0,n+1} \subseteq X_{k|i}^{*,n}$, then

$$\max_{x \in X_{k|i}^{0,n+1}} [f(x, Kx + c_{k|i}^{0,n+1})]_j \leq \max_{x \in X_{k|i}^{*,n}} [f(x, Kx + c_{k|i}^{*,n} + c_{k|i}^{0,n})]_j$$

and hence $\bar{x}_{k+1|i}^{0,n+1} \leq \bar{x}_{k+1|i}^{*,n}$ by (18) and (20d), which demonstrates (24). Conditions (25) and (26) follow from the \mathbf{c}_{i+1}^0 update in line 6 of Algorithm 3, which implies $c_{k+1|i}^{0,1} = c_{k+1|i}^{*,n_i} + c_{k+1|i}^{0,n_i}$, $k = 0, \dots, N-2$, and $c_{N-1|i+1}^{0,1} = u_{N+i}^r - Kx_{N+i}^r$. Hence (25) can be shown analogously to (24), and (26) holds because $X_{N|i}^{*,n_i} \subseteq \mathcal{X}_N$ by (20f) and \mathcal{X}_N satisfies Assumption 4. \square

Theorem 5 (Recursive feasibility). If Assumptions 2 and 4 hold, and if $\mathbf{c}_0, \mathbf{X}_0$ exist satisfying (20b)-(20f) with the initial seed \mathbf{c}_0^0 , then the RcMPC RHOCPC is feasible at each iteration of Algorithm 3, for all times $i \geq 0$.

Proof. If, at time i and iteration n the RcMPC RHOCPC in line 4 of Algorithm 3 has solution $(\mathbf{c}_i^{*,n}, \mathbf{X}_i^{*,n})$, then $(\mathbf{c}_i, \mathbf{X}_i) = (0, \mathbf{X}_i^{0,n+1})$ is a feasible solution at iteration $n+1$ by Lemma 2 since $X_{0|i}^{0,n+1} = \{x_i\} = X_{0|i}^{*,n}$. Similarly, Lemma 2 implies that, if $(\mathbf{c}_i^{*,n_i}, \mathbf{X}_i^{*,n_i})$ is the solution of the RcMPC RHOCPC at the final iteration performed at time i , then $(\mathbf{c}_{i+1}, \mathbf{X}_{i+1}) = (0, \mathbf{X}_{i+1}^{0,1})$ is a feasible solution at the first iteration at time $i+1$ since $X_{0|i+1}^{0,1} = \{x_{i+1}\} \subseteq X_{0|i}^{*,n_i}$. Furthermore, feasibility of (20b)-(20f) with the initial seed \mathbf{c}_0^0 ensures that the first RcMPC RHOCPC is feasible. Hence feasibility is ensured at all iterations and all time steps. \square

Lemma 3. For all $i \geq 0$ and $n \geq 1$ the optimal cost satisfies $J_{\text{RcMPC},i}^{*,n+1} \leq J_{\text{RcMPC},i}^{*,n}$.

Proof. Lemma 2 implies $(\mathbf{c}_i, \mathbf{X}_i) = (0, \mathbf{X}_i^{0,n+1})$ is a feasible solution of the RcMPC RHOCPC at iteration $n+1$. Therefore the optimal solution at iteration $n+1$ satisfies $J_{\text{RcMPC},i}^{*,n+1} \leq J_{\text{RcMPC}}(0, \mathbf{X}_i^{0,n+1}, \mathbf{c}_i^{0,n+1})$. But from Lemma 2 we also have $X_{k|i}^{0,n+1} \subseteq X_{k|i}^{*,n}$ for $k = 0, \dots, N$, and the cost (20g) therefore ensures $J_{\text{RcMPC}}(0, \mathbf{X}_i^{0,n+1}, \mathbf{c}_i^{*,n}) \leq J_{\text{RcMPC},i}^{*,n}$. \square

Remark 10. Since $J_{\text{RcMPC},i}^{*,n} \geq 0$, Lemma 3 implies that, in the ideal case of unlimited iterations, the optimal RcMPC cost in line 4 of Algorithm 3 converges to a finite limit as $n \rightarrow \infty$. As in the nominal case, it follows that $\mathbf{c}_i^{*,n} \rightarrow 0$ as $n \rightarrow \infty$. In this case however, the seed tube $\mathbf{X}_i^{0,n}$ generated by Algorithm 2 will in general contain more than a single trajectory at all iterations $n \geq 1$, and hence $\mathbf{X}_i^{0,n}$ does not converge to a single trajectory.

Theorem 6. If Assumptions 2 and 4 hold and the offline computation of Algorithm 3 is feasible, then the closed-loop system formed by (15) and Algorithm 3 satisfies the state and input constraints (5) and the quadratic stability condition:

$$\lim_{t \rightarrow \infty} \frac{1}{t} \sum_{i=0}^{t-1} (\|x_i - x_i^r\|_Q^2 + \|u_i - u_i^r\|_R^2) \leq \beta \quad (31)$$

where β satisfies Assumption 4(c).

Proof. By Lemma 2 and Theorem 5, $(\mathbf{c}_{i+1}, \mathbf{X}_{i+1}) = (0, \mathbf{X}_{i+1}^{0,1})$ is a feasible solution for the RcMPC RHOCPC at the first iteration of Algorithm 3 at time $i+1$, so

$$\begin{aligned} J_{\text{RcMPC},i+1}^{*,1} &\leq J_{\text{RcMPC}}(0, \mathbf{X}_{i+1}^{0,1}, \mathbf{c}_{i+1}^{0,1}) \\ &\leq J_{\text{RcMPC}}(\mathbf{c}_i^{*,n_i}, \mathbf{X}_i^{*,n_i}, \mathbf{c}_i^{0,n_i}) \\ &\quad - \|x_i - x_i^r\|_Q^2 - \|u_i - u_i^r\|_R^2 + \beta, \end{aligned}$$

where the definition of $\mathbf{c}_{i+1}^{0,1}$ in line 6 and the inequality in Assumption 4(c) have been used. But Lemma 3 implies $J_{\text{RcMPC},i+1}^{*,n_i+1} \leq J_{\text{RcMPC},i+1}^{*,1}$ and hence

$$J_{\text{RcMPC},i+1}^{*,n_i+1} \leq J_{\text{RcMPC},i}^{*,n_i} - \|x_i - x_i^r\|_Q^2 - \|u_i - u_i^r\|_R^2 + \beta. \quad (32)$$

The bound (31) follows by summing both sides of (32) over $i = 0, \dots, t-1$ and noting that $J_{\text{RcMPC},i}^{*,n_i}$ is finite for all i since \mathcal{X}, \mathcal{U} are compact and the RcMPC RHOCPC is feasible. \square

Remark 11. If Assumption 4 reduces to Assumption 1 in the absence of disturbances (i.e. if $\beta \rightarrow 0$ as $\mathcal{W} \rightarrow \{0\}$), then the origin ($x - x^r = 0$) of the closed-loop system of (15) with Alg. 3 is input-to-state stable (ISS) [31], since (32) and Assumption 4(c) imply $J_{\text{RcMPC},i}^{*,n_i}$ is an ISS Lyapunov function.

VI. NUMERICAL EXAMPLES

This section presents numerical results obtained by applying the nominal cMPC (Algorithm 1) and robust RcMPC (Algorithm 3) strategies to two example problems. In each case the algorithms are implemented in Python, using cvxpy [32] and MOSEK [33] for convex optimisation problems, and the SciPy SLSQP algorithm [34] for comparisons with a general nonlinear programming solver. Unless stated otherwise $\text{maxiters} = 3$,

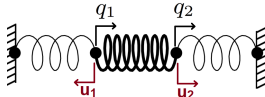


Fig. 1: FPU system with two unit masses with coordinates q_1, q_2 , and control forces u_1, u_2 applied directly to each mass.

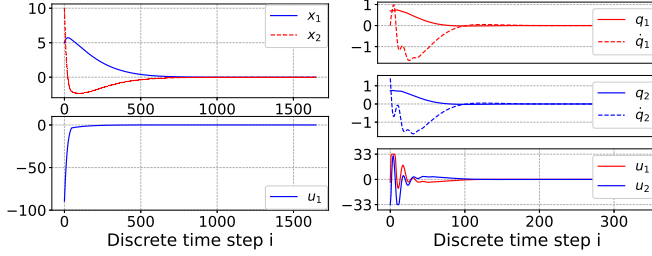


Fig. 2: Closed-loop trajectories with cMPC (Algorithm 1). Left: Example 1. Right: Example 2.

$tol = 10^{-6}$ and MPC simulations are terminated when $\|x_{N|i}\|_2$ reaches 10^{-4} (Example 1) or 10^{-6} (Example 2).

Example 1 First we consider a system with model

$$\begin{aligned}\dot{x}_1(t) &= x_2(t) + w_1(t) \\ \dot{x}_2(t) &= 0.2e^{-x_1(t)} - x_2(t) + u_1(t) - 0.2 + w_2(t).\end{aligned}$$

and cost weights $Q = \text{diag}\{1, 1\}$, $R = 1$, and initial condition $x_1(0) = 5$, $x_2(0) = 10$. Forward Euler discretization is used with time step $\Delta t = 8 \times 10^{-3}$, and the prediction horizon is $N = 25$. The reference trajectory is chosen as $x_i^r = 0$, $u_i^r = 0$ for all i , and the system constraints are $|x_1(t)| \leq 10$, $|x_2(t)| \leq 10$, $|u_1(t)| \leq 150$ at all discrete times $t = i\Delta t$, $i = 0, 1, \dots$

Example 2 Applying a change of coordinates to the Fermi Pasta Ulam (FPU) chain with cubic spring potentials (Figure 1), we obtain a convex model formulation:

$$\begin{aligned}\dot{x}_1(t) &= x_3(t) + w_1(t) \\ \dot{x}_2(t) &= x_4(t) + w_2(t) \\ \dot{x}_3(t) &= -\frac{\eta^2}{2}(x_2(t) + x_1(t)) + 3x_1(t)^2 + u_1(t) + w_3(t) \\ \dot{x}_4(t) &= -\frac{\eta^2}{2}(x_2(t) + x_1(t)) + 3x_2(t)^2 + u_2(t) + w_4(t).\end{aligned}$$

Forward Euler discretization with $\Delta t = 10^{-2}$ and a prediction horizon $N = 25$ are used. The model and algorithm parameters are $\eta = 50$, $x_i^r = 0$, $u_i^r = 0$ for all i , $Q = 10^{-2}I$, $R = 10^{-2}I$, $|x_1(t)| \leq 10$, $|x_2(t)| \leq 10$, $|x_3(t)| \leq 10$, $|x_4(t)| \leq 10$, $|u_1(t)| \leq 33$, $|u_2(t)| \leq 33$, $t = i\Delta t$, $i = 0, 1, \dots$, and

$$\begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} = E \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} x_3(0) \\ x_4(0) \end{bmatrix} = E \begin{bmatrix} 1/\eta \\ 1 \end{bmatrix}, E = \frac{1}{\sqrt{2}} \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix}.$$

A. Nominal cMPC

The performance of the nominal cMPC strategy (Algorithm 1) described in Section III was investigated by setting $w_1(t) = w_2(t) = 0$, $w_3(t) = w_4(t) = 0$ for all t in Examples 1 and 2. The initial seed was obtained using the initialization strategy of Section III. Figure 2 shows the predicted trajectories of Algorithm 1 for both example problems. As expected from Lemma 1 and Theorem 2, the optimal predicted cost is

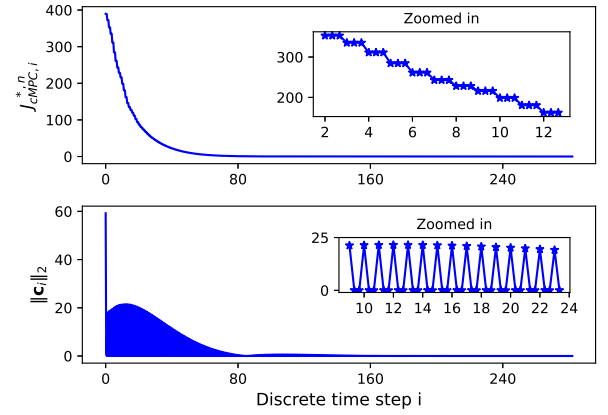


Fig. 3: Convergence of cMPC at successive iterations and time steps for Example 2. Upper plot: Optimal cost. Lower: $\|c_i^*\|$.

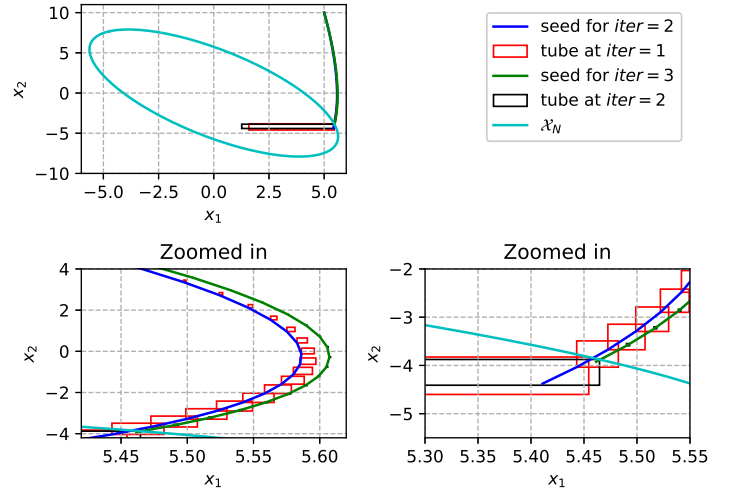


Fig. 4: Seed trajectories and tubes ($x_{i|k}^0$ and $x_{i|k}^0 \oplus S_{i|k}^*$) for cMPC and Example 1 at iterations $n = 1, 2, 3$ at time $i = 0$.

non-increasing at successive iterations and time steps, and c_i^* converges to 0. This is shown in Fig. 3 for Example 2, where the points between integer time steps represent the iterations performed at each time step. The convergence of the tube to a single trajectory (Theorem 2 and Remark 4) is illustrated in Fig. 4 for Example 1. The iteration converges rapidly (S_0^* is already difficult to see at the second iteration) and the tube at iteration n contains the seed trajectory updated for iteration $n+1$ via line 6 of Algorithm 1, as expected from Theorem 1.

Timing tests were conducted by comparing the cvxpy implementation of Algorithm 1 to a receding horizon strategy that solves the nMPC RHOCIP at each time step using the `scipy.optimize` library and its SLSQP solver. The cMPC RHOCIP in line 4 of Algorithm 1 is implemented using the `parameters` functionality of the `cvxpy` library and is thus precompiled offline. The solver timings were tested both with and without pre-compilation using the `jit` functionality of the `jax` library [35]. For the `cvxpy` algorithm `jit` is used to speed up lines 3 and 6, whereas for `scipy.optimize` it is used to pre-compile the constraints, constraint Jacobians, cost function and cost gradient. Results obtained using a i7-10700 CPU at

Example	solver	maxiters	n_v	n_c	n_p	time	total	total	total	J_0^*	J_{cl}
						for iter 1 $i = 0$	time at $i = 0$	time at $i = 1$	time at $i = 2$		
Example 1	cvxpy	1				0.038	0.229	0.053	0.055	132297	80588
	cvxpy	3				0.039	0.331	0.168	0.154	121999	80152
	cvxpy	5				0.040	0.494	0.258	0.262	121932	80138
	cvxpy+jit	1	180	510	277	0.040	0.255	0.039	0.057	132297	80588
	cvxpy+jit	3				0.039	0.292	0.124	0.122	121999	80152
	cvxpy+jit	5				0.041	0.391	0.204	0.191	121932	80137
Example 2	NLP	-	77	355	-	-	13.401	17.881	12.828	121782	77340
	NLP+jit	-					1.547	1.562	1.507	121782	77340
	cvxpy	1				0.970	1.284	0.192	0.192	386.77	198.85
	cvxpy	3				0.902	2.056	0.634	0.629	389.27	198.80
	cvxpy+jit	1	309	2034	854	0.992	1.259	0.167	0.152	389.77	198.85
	cvxpy+jit	3				0.974	1.608	0.447	0.459	389.27	198.80
Example 2	NLP	-	154	709	-	-	31.602	34.365	37.811	389.27	198.80
	NLP+jit	-					2.375	2.099	2.060	389.27	198.80

TABLE I: Comparison of solution times and optimal costs.

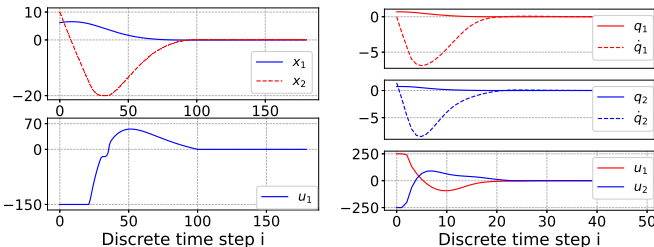
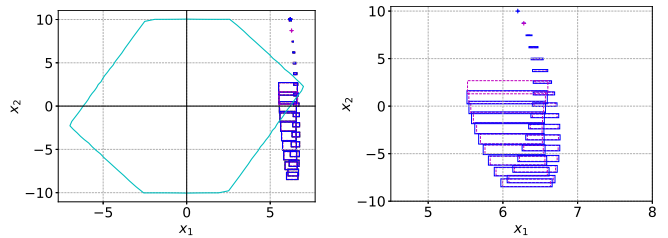
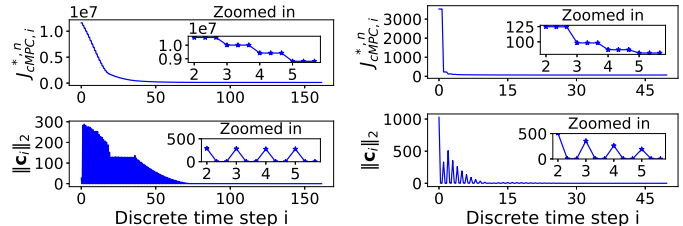


Fig. 5: Closed-loop state trajectories with RcMPC. Left: Example 1. Right: Example 2

2.90GHz are shown in Table I, where n_v , n_c , n_p are the number of variables, constraints and parameters respectively, J_0^* is the optimal cost at $i = 0$ and J_{cl} is the cost evaluated using closed-loop system responses. For Example 1 with standard implementation at $i = 0$ Algorithm 1 requires only 1.71% and 3.69% of the nMPC solution time for $maxiters = 1$ and 5 respectively. Even when special just-in-time compilation with jit is used to speed up both cMPC and nMPC, Algorithm 1 still presents great improvement requiring 16.48% and 25.27% of the nMPC solution time for $maxiters = 1$ and 5 respectively. This percentage reduces to 2.50% and 13.06% at $i = 1$. For Example 2 we have a larger state space and hence more constraints in the cMPC RHOC. Despite this we see that for $i = 0$ Algorithm 1 requires 53.01% and 67.71% of the NLP solution time with jit for both approaches and 4.06% and 6.51% without jit, for $maxiters = 1$ and 3 respectively. Again for $i > 0$ these percentages reduce significantly further. Furthermore, as $maxiters$ increases the optimal cMPC RHOC cost decreases and converges to the optimal nMPC RHOC cost, which is consistent with Theorem 3.

B. Robust cMPC (RcMPC)

The performance of RcMPC (Algorithm 3) presented in Section V was investigated using Examples 1 and 2 with non-zero disturbance bounds of $|w_j(t)| \leq 0.02$, $j = 1, 2$, and $|w_j(t)| \leq 10^{-4}$, $j = 1, 2, 3, 4$, respectively, for all t . All other RHOC parameters remain the same for Examples 1 and 2 except in the former $|x_1(t)| \leq 20$, $j = 1, 2$, $x(0) = [6.2 \ 10]^T$ and in the latter the horizon is shortened to $N = 10$, $Q = 10^{-4}I$, $R = 10^{-4}I$, and the constraints are relaxed to $|x_j(t)| \leq 20$, $j = 1, 2, 3, 4$ and $|u_j(t)| \leq 250$, $j = 1, 2$

Fig. 6: Optimal and seed tubes for RcMPC for Example 1: $X_0^{*,3}$ (blue, solid lines), $X_1^{0,1}$ (pink dashed). Left: Tube cross-sections for prediction times $k = 0, \dots, N - 1$. Right: Tube cross-sections for $k = 0, \dots, N$ and terminal set X_N (cyan).Fig. 7: Convergence of RcMPC optimal cost and control perturbation sequence $\|c_i^*\|$. Left: Example 1. Right: Example 2.

$\forall t$. The initial seed in both examples was obtained using the initialization strategy described in Section V. Figures 5 and 6 show the closed-loop trajectories for RcMPC applied to Example 1 with state x_2 reaching a minimum value of -19.98 , thus robustly satisfying the constraints. Figure 6 shows that, in agreement with Theorem 5, the seed tube $X_1^{0,1}$ computed at the first iteration at time $i = 1$ is contained within the optimal tube $X_0^{*,3}$ computed at the final iteration for $i = 0$. Figure 7 demonstrates that, for any time step i , the optimal cost decreases at successive iterations and the perturbation sequence c_i^* converges to a value close to zero, as expected from Lemma 3 and Remark 10. Although the control perturbation converges to zero during the iteration at time step i , there is no guarantee that it will be smaller at first iteration at $i + 1$. Thus, $\|c_i^*\| \not\rightarrow 0$ as $i \rightarrow \infty$ in Figure 7.

VII. CONCLUSIONS

This paper proposes a robust nonlinear model predictive control strategy for systems with convex state and control constraints, and with dynamics that can be expressed as a difference of convex functions. At each discrete time step, a sequence of convex problems is obtained by linearizing the system model around predicted trajectories. Robust tubes are used to bound model trajectories, thus accounting for linearization errors and external disturbances. In the absence of external disturbances we show that the successive approximation approach is non-conservative, since the predicted tubes and control sequences converge to a single trajectory which is a local optimum for the original nonlinear optimal control problem. We then reformulate the approach to consider additive disturbances, and derive a robust receding horizon strategy using a novel strategy for the selection of linearization points that provides a guarantee of recursive feasibility. We also establish a form of quadratic stability for the closed-loop

system. As in the disturbance-free case, the robust approach solves a sequence of convex programming problems at each discrete time step without needing pre-computed error bounds or heuristics. This iteration can be terminated early without affecting stability or feasibility guarantees.

REFERENCES

- [1] B. Kouvaritakis and M. Cannon, *Model Predictive Control: Classical, Robust and Stochastic*. Springer, 2016.
- [2] H. Bock, M. Diehl, D. Leineweber, and J. Schlöder, “A direct multiple shooting method for real-time optimization of nonlinear DAE processes,” in *Nonlinear Model Predictive Control* (F. Allgöwer and A. Zheng, eds.), pp. 245–267, Birkhäuser Basel, 2000.
- [3] J. Rosen, “Iterative solution of nonlinear optimal control problems,” *SIAM Journal on Control*, vol. 4, no. 1, pp. 223–244, 1966.
- [4] Y. Mao, M. Szmuk, and B. Açikmeşe, “Successive convexification of non-convex optimal control problems and its convergence properties,” in *IEEE 55th Conference on Decision and Control*, pp. 3636–3641, 2016.
- [5] M. Cannon, J. Buerger, B. Kouvaritakis, and S. Rakovic, “Robust tubes in nonlinear model predictive control,” *IEEE Trans Automat Contr*, vol. 56, no. 8, pp. 1942–1947, 2011.
- [6] Y. Lee, B. Kouvaritakis, and M. Cannon, “Constrained receding horizon predictive control for nonlinear systems,” *Automatica*, vol. 38, no. 12, pp. 2093–2102, 2002.
- [7] R. Foust, S.-J. Chung, and F. Y. Hadaegh, “Optimal guidance and control with nonlinear dynamics using sequential convex programming,” *Journal of Guidance, Control, and Dynamics*, vol. 43, no. 4, pp. 633–644, 2020.
- [8] M. Doff-Sotta, M. Cannon, and J. Forbes, “Spacecraft energy management using convex optimisation,” *Energy Conversion and Management: X*, vol. 16, p. 100325, 2022.
- [9] C. Zagaris, H. Park, J. Virgili-Llop, R. Zappulla, M. Romano, and I. Kolmanovsky, “Model predictive control of spacecraft relative motion with convexified keep-out-zone constraints,” *J Guid Control Dyn*, vol. 41, no. 9, pp. 2054–2062, 2018.
- [10] B. Acikmese and S. R. Ploen, “Convex programming approach to powered descent guidance for mars landing,” *J Guid Control Dyn*, vol. 30, no. 5, pp. 1353–1366, 2007.
- [11] M. E. Villanueva, R. Quirynen, M. Diehl, B. Chachuat, and B. Houska, “Robust MPC via min–max differential inequalities,” *Automatica*, vol. 77, pp. 311–321, 2017.
- [12] J. Köhler, R. Soloperto, M. A. Müller, and F. Allgöwer, “A computationally efficient robust model predictive control framework for uncertain nonlinear systems,” *IEEE Transactions on Automatic Control*, vol. 66, no. 2, pp. 794–801, 2020.
- [13] A. Sasfi, M. N. Zeilinger, and J. Köhler, “Robust adaptive MPC using control contraction metrics,” *Automatica*, vol. 155, p. 111169, 2023.
- [14] D. Q. Mayne, E. C. Kerrigan, E. Van Wyk, and P. Falugi, “Tube-based robust nonlinear model predictive control,” *International journal of robust and nonlinear control*, vol. 21, no. 11, pp. 1341–1353, 2011.
- [15] 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, no. 2, pp. 194–200, 2013.
- [16] B. T. Lopez, J.-J. E. Slotine, and J. P. How, “Dynamic tube MPC for nonlinear systems,” in *2019 American Control Conference (ACC)*, pp. 1655–1662, IEEE, 2019.
- [17] F. Messerer and M. Diehl, “An efficient algorithm for tube-based robust nonlinear optimal control with optimal linear feedback,” in *60th IEEE Conference on Decision and Control*, pp. 6714–6721, IEEE, 2021.
- [18] R. Horst and N. V. Thoai, “Dc programming: overview,” *Journal of Optimization Theory and Applications*, vol. 103, pp. 1–43, 1999.
- [19] G. Lanckriet and B. K. Sriperumbudur, “On the convergence of the concave-convex procedure,” in *Advances in Neural Information Processing Systems 22 (NIPS 2009)*, pp. 1759–1767, 2009.
- [20] T. Lipp and S. Boyd, “Variations and extension of the convex–concave procedure,” *Optimization and Engineering*, vol. 17, pp. 263–287, 2016.
- [21] M. Fiacchini, T. Alamo, and E. F. Camacho, “On the computation of convex robust control invariant sets for nonlinear systems,” *Automatica*, vol. 46, no. 8, pp. 1334–1338, 2010.
- [22] A. Ahmadi and G. Hall, “DC decomposition of nonconvex polynomials with algebraic techniques,” *Math. Program.*, vol. 169, pp. 69–94, 2018.
- [23] G. Misra and X. Bai, “Iteratively feasible optimal spacecraft guidance with non-convex path constraints using convex optimization,” in *AIAA Scitech 2020 Forum*, p. 1350, 2020.
- [24] M. Doff-Sotta and M. Cannon, “Difference of convex functions in robust tube MPC,” in *61st IEEE Conference on Decision and Control*, 2022.
- [25] J. Nocedal and S. Wright, *Numerical optimization*. Springer, 1999.
- [26] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Foundations and Trends in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.
- [27] Y. Lishkova, M. Cannon, and S. Ober-Blobbaum, “A multirate variational approach to nonlinear MPC,” in *2022 European Control Conference (ECC)*, pp. 729–735, IEEE, 2022.
- [28] J. Buerger, M. Cannon, and M. Doff-Sotta, “Safe learning in nonlinear model predictive control,” in *Proceedings of the 6th Annual Learning for Dynamics & Control Conference*, vol. 242 of *Proceedings of Machine Learning Research*, pp. 603–614, PMLR, 2024.
- [29] M. Lorenzen, M. Cannon, and F. Allgöwer, “Robust mpc with recursive model update,” *Automatica*, vol. 103, pp. 461–471, 2019.
- [30] F. Blanchini, S. Miani, *et al.*, *Set-theoretic methods in control*, vol. 78. Springer, 2008.
- [31] D. Limon, T. Alamo, D. Raimondo, D. M. De La Peña, J. Bravo, A. Ferramosca, and E. Camacho, “Input-to-state stability: a unifying framework for robust model predictive control,” in *Nonlinear model predictive control*, pp. 1–26, Springer, 2009.
- [32] S. Diamond and S. Boyd, “CVXPY: A Python-embedded modeling language for convex optimization,” *J Mach Learn Res*, vol. 17, no. 83, pp. 1–5, 2016.
- [33] MOSEK ApS, *The MOSEK optimization toolbox for MATLAB manual. Version 10.0.*, 2022.
- [34] P. Virtanen *et al.*, “SciPy 1.0: Fundamental algorithms for scientific computing in Python,” *Nature Methods*, vol. 17, pp. 261–272, 2020.
- [35] J. Bradbury *et al.*, “JAX: composable transformations of Python+NumPy programs,” 2018.

APPENDIX

Proof of Theorem 3: Theorem 2 shows that the cMPC iteration converges to a fixed point. We now prove that this point is a local optimum of the nMPC RHOC. For readability we set $n_x = 1$, but the proof for $n_x > 1$ is analogous. We also assume that the nMPC RHOC (6) satisfies a suitable constraint qualification (e.g. linear independence constraint qualification, [25] Def. 12.4), so that a local minimizer satisfies the first order necessary conditions ([25] Thm. 12.1).

Theorem 2 implies that the cMPC iteration converges to $\{c_{0|i}, \dots, c_{N-1|i}\} = \{0, \dots, 0\}$, $S_{0|i} = \{0\}, \dots, S_{N|i} = \{0\}$, and we then have $(\mathbf{x}_i^*, \mathbf{u}_i^*) = (\mathbf{x}_i^{0,j}, \mathbf{u}_i^{0,j})$ due to (8), (11). To prove that this point coincides with a local optimum for both cMPC and nMPC RHOCs, we consider the equivalence to first-order of the cMPC and nMPC RHOCs at $(\mathbf{x}_i, \mathbf{u}_i) = (\mathbf{x}_i^{0,j}, \mathbf{u}_i^{0,j})$. From (6b)-(6g) and (14b)-(14g) it is clear that if only constraints (6b)-(6c), (6e)-(6f) and (14b)-(14c), (14e)-(14f) were present in the nMPC and cMPC RHOCs, then their first order optimality conditions would be identical at $\mathbf{x}_i = \mathbf{x}_i^0$, $\mathbf{u}_i = \mathbf{u}_i^0$ and $\{c_{0|i}, \dots, c_{N-1|i}\} = \{0, \dots, 0\}$, $S_{0|i} = \{0\}, \dots, S_{N|i} = \{0\}$. Therefore we prove that the same is true when (6d) and (14d) are included, by showing that the following two simplified problems are equivalent to first-order:

$$\text{nMPC simplified: } \mathbf{c}_i^* := \arg \min_{\mathbf{x}_i, \mathbf{c}_i} J_{\text{nMPC}}^c(\mathbf{x}_i, \mathbf{c}_i) \quad (33)$$

subject to, for $k = 0, \dots, N-1$, $x_{k+1|i} = \phi(x_{k|i}, c_{k|i})$, where $\phi(x_{k|i}, c_{k|i}) = f(x_{k|i}, u_{k|i})$ with $u_{k|i} = Kx_{k|i} + c_{k|i}$ and $J_{\text{nMPC}}^c(\mathbf{x}_i, \mathbf{c}_i) = J_{\text{nMPC}}(\mathbf{x}_i, \mathbf{u}_i)$, and

$$\text{cMPC simplified: } (\mathbf{c}_i^*, \mathbf{S}_i^*) = \arg \min_{\mathbf{c}_i, \mathbf{S}_i} J_{\text{cMPC}}^c(\mathbf{c}_i, \mathbf{S}_i, \mathbf{x}_i^0, \mathbf{c}_i^0) \quad (34)$$

subject to, for $k = 0, \dots, N-1$:

$$\bar{s}_{k+1|i} \geq \phi(\bar{s}_{k|i} + x_{k|i}^0, c_{k|i} + u_{k|i}^0 - Kx_{k|i}^0) - \phi(x_{k|i}^0, c_{k|i}^0)$$

$$\begin{aligned}\bar{s}_{k+1|i} &\geq \phi(\underline{s}_{k|i} + x_{k|i}^0, c_{k|i} + u_{k|i}^0 - Kx_{k|i}^0) - \phi(x_{k|i}^0, c_{k|i}^0) \\ \underline{s}_{k+1|i} &\leq \Phi_{k|i}\bar{s}_{k|i} + B_{k|i}c_{k|i} \\ \underline{s}_{k+1|i} &\leq \Phi_{k|i}\underline{s}_{k|i} + B_{k|i}c_{k|i}\end{aligned}$$

with $u_{k|i}^0 = Kx_{k|i}^0 + c_{k|i}^0$, $J_{\text{cMPC}}(\cdot, \cdot, \cdot, \cdot) = J_{\text{cMPC}}(\cdot, \cdot, \cdot, \cdot)$, and

$$\begin{aligned}\phi(s + x^0, c + u^0 - Kx^0) &= f(s + x^0, Ks + c + u^0), \\ \frac{\partial \phi(x, c)}{\partial x} &= \frac{\partial f(x, Kx + c)}{\partial x} = \frac{\partial f(x, u)}{\partial x} + \frac{\partial f(x, u)}{\partial u} \frac{\partial u}{\partial x} = A + BK = \Phi, \\ \frac{\partial \phi(x, c)}{\partial c} &= \frac{\partial f(x, Kx + c)}{\partial c} = \frac{\partial f(x, u)}{\partial u} \frac{\partial u}{\partial c} = B.\end{aligned}$$

The Lagrangian function for the simplified nMPC RHOCp: $L_{\text{nMPC}}(\mathbf{x}_i, \mathbf{c}_i, \lambda_i) = J_{\text{nMPC}}^c(\mathbf{x}_i, \mathbf{c}_i) - \sum_{k=0}^N \lambda_{k|i} [x_{k+1|i} - \phi(x_{k|i}, c_{k|i})]$, implies the following Karush-Kuhn-Tucker (KKT) first-order conditions [25] for (33)

$$0 = \frac{\partial J_{\text{nMPC}}^c(\mathbf{x}_i, \mathbf{c}_i)}{\partial x_{0|i}} |_{(\mathbf{x}_i^{*,n}, \mathbf{c}_i^{*,n})} + \lambda_{0|i}^* \frac{\partial \phi(x, c)}{\partial x} |_{(x_{0|i}^{*,n}, c_{0|i}^{*,n})} \quad (35a)$$

and for $k = 0, \dots, N-1$

$$0 = \frac{\partial J_{\text{nMPC}}^c(\mathbf{x}_i, \mathbf{c}_i)}{\partial x_{k|i}} |_{(\mathbf{x}_i^{*,n}, \mathbf{c}_i^{*,n})} + \lambda_{k|i}^{*,n} \frac{\partial \phi(x, c)}{\partial x} |_{(x_{k|i}^{*,n}, c_{k|i}^{*,n})} - \lambda_{k-1|i}^{*,n} \quad (35b)$$

$$0 = \frac{\partial J_{\text{nMPC}}^c(\mathbf{x}_i, \mathbf{u}_i)}{\partial u_{k|i}} |_{(\mathbf{x}_i^{*,n}, \mathbf{c}_i^{*,n})} + \lambda_{k|i}^{*,n} \frac{\partial \phi(x, c)}{\partial c} |_{(x_{k|i}^{*,n}, u_{k|i}^{*,n})} \quad (35c)$$

$$0 = x_{k+1|i}^{*,n} - \phi(x_{k|i}^{*,n}, c_{k|i}^{*,n}) \quad (35d)$$

where $(\cdot)^{*,n}$ denotes a local optimum point of (33). For the simplified cMPC RHOCp we define the Lagrangian $L_{\text{cMPC}}(\mathbf{c}_i, \mathbf{S}_i, \mathbf{x}_i^0, \mathbf{u}_i^0, \alpha_i, \beta_i, \gamma_i, \mu_i) = J_{\text{cMPC}}^c(\mathbf{c}_i, \mathbf{S}_i, \mathbf{x}_i^0, \mathbf{u}_i^0) - \sum_{k=0}^{N-1} \alpha_{k|i} [\bar{s}_{k+1|i} - \phi(\bar{z}_{k|i}, c_{k|i}^0)] - \sum_{k=0}^{N-1} \beta_{k|i} [\underline{s}_{k+1|i} - \phi(\underline{z}_{k|i}, c_{k|i}^0)] - \sum_{k=0}^{N-1} \gamma_{k|i} (\Phi_{k|i}\bar{s}_{k|i} + B_{k|i}c_{k|i} - \underline{s}_{k+1|i}) - \sum_{k=0}^{N-1} \mu_{k|i} (\Phi_{k|i}\underline{s}_{k|i} + B_{k|i}c_{k|i} - \underline{s}_{k+1|i})$ with KKT conditions:

$$0 = \frac{\partial J_{\text{cMPC}}}{\partial \bar{s}_{0|i}} |_{\mathbf{z}_i^{*,c}} + \alpha_{0|i}^{*,c} \frac{\partial \phi(x, c)}{\partial x} |_{\bar{z}_{0|i}^{*,c}} - \gamma_{0|i}^* \frac{\partial \phi(x_{0|i}^0, c_{0|i}^0)}{\partial x_{0|i}^0} \quad (36a)$$

$$0 = \frac{\partial J_{\text{cMPC}}}{\partial \underline{s}_{0|i}} |_{\mathbf{z}_i^{*,c}} + \beta_{0|i}^{*,c} \frac{\partial \phi(x, c)}{\partial x} |_{\underline{z}_{0|i}^{*,c}} - \mu_{k|i}^* \frac{\partial \phi(x_{k|i}^0, c_{k|i}^0)}{\partial x_{k|i}^0} \quad (36b)$$

and for $k = 1, \dots, N-1$

$$\begin{aligned}0 &= \frac{\partial J_{\text{cMPC}}}{\partial \bar{s}_{k|i}} |_{\mathbf{z}_i^{*,c}} - (\alpha_{k-1|i}^{*,c} + \beta_{k-1|i}^{*,c}) + \alpha_{k|i}^{*,c} \frac{\partial \phi(x, u)}{\partial x} |_{\bar{z}_{k|i}^{*,c}} \\ &\quad - \gamma_{k|i}^* \frac{\partial \phi(x_{k|i}^0, c_{k|i}^0)}{\partial x_{k|i}^0} \quad (36c)\end{aligned}$$

$$\begin{aligned}0 &= \frac{\partial J_{\text{cMPC}}}{\partial \underline{s}_{k|i}} |_{\mathbf{z}_i^{*,c}} + (\mu_{k-1|i}^{*,c} + \gamma_{k-1|i}^{*,c}) + \beta_{k|i}^{*,c} \frac{\partial \phi(x, c)}{\partial x} |_{\underline{z}_{k|i}^{*,c}} \\ &\quad - \mu_{k|i}^* \frac{\partial \phi(x_{k|i}^0, c_{k|i}^0)}{\partial x_{k|i}^0} \quad (36d)\end{aligned}$$

$$\begin{aligned}0 &= \frac{\partial J_{\text{cMPC}}}{\partial c_{k|i}} |_{\mathbf{z}_i^{*,c}} + \alpha_{k|i}^{*,c} \frac{\partial \phi(x, c)}{\partial c} |_{\bar{z}_{k|i}^{*,c}} + \beta_{k|i}^{*,c} \frac{\partial \phi(x, c)}{\partial c} |_{\underline{z}_{k|i}^{*,c}} \\ &\quad - (\gamma_{k|i}^* + \mu_{k|i}^*) \frac{\partial \phi(x_{k|i}^0, c_{k|i}^0)}{\partial c_{k|i}^0} \quad (36e)\end{aligned}$$

$$0 = \alpha_{k|i} (\bar{s}_{k+1|i}^{*,c} - \phi(\bar{z}_{k|i}^{*,c}) + \phi(x_{k|i}^0, u_{k|i}^0)) \quad (36f)$$

$$0 = \beta_{k|i} (\underline{s}_{k+1|i}^{*,c} - \phi(\underline{z}_{k|i}^{*,c}) + \phi(x_{k|i}^0, u_{k|i}^0)) \quad (36g)$$

$$0 = \gamma_{k|i} (\Phi_{k|i}\bar{s}_{k|i}^{*,c} + B_{k|i}c_{k|i}^{*,c} - \underline{s}_{k+1|i}^{*,c}) \quad (36h)$$

$$0 = \mu_{k|i} (\Phi_{k|i}\underline{s}_{k|i}^{*,c} + B_{k|i}c_{k|i}^{*,c} - \underline{s}_{k+1|i}^{*,c}) \quad (36i)$$

$$\alpha_{k|i}^{*,c} \geq 0, \beta_{k|i}^{*,c} \geq 0, \gamma_{k|i}^{*,c} \geq 0, \mu_{k|i}^{*,c} \geq 0 \quad (36j)$$

where $\mathbf{z}_i^{*,c} = (\mathbf{c}_i^{*,c}, \mathbf{S}_i^{*,c}, \mathbf{x}_i^0, \mathbf{u}_i^0)$ and

$$\nabla_{c_{k|i}} \phi(\underline{s}_{k|i} + x_{k|i}^0, c_{k|i} + u_{k|i}^0 - Kx_{k|i}^0) = \nabla_c \phi(x, c) |_{\underline{z}_{k|i}^{*,c}}$$

$$\nabla_{\underline{s}_{k|i}} \phi(\underline{s}_{k|i} + x_{k|i}^0, c_{k|i} + u_{k|i}^0 - Kx_{k|i}^0) = \nabla_x \phi(x, u) |_{\underline{z}_{k|i}^{*,c}}$$

$$\begin{aligned}(\bar{s}_{k|i} + x_{k|i}^0, c_{k|i} + u_{k|i}^0 - Kx_{k|i}^0) &= \bar{z}_{k|i} \\ (\underline{s}_{k|i} + x_{k|i}^0, c_{k|i} + u_{k|i}^0 - Kx_{k|i}^0) &= \underline{z}_{k|i} \\ (\bar{s}_{k|i}^{*,c} + x_{k|i}^0, c_{k|i}^{*,c} + u_{k|i}^0 - Kx_{k|i}^0) &= \bar{z}_{k|i}^{*,c} \\ (\underline{s}_{k|i}^{*,c} + x_{k|i}^0, c_{k|i}^{*,c} + u_{k|i}^0 - Kx_{k|i}^0) &= \underline{z}_{k|i}^{*,c}\end{aligned}$$

where $(\cdot)^{*,c}$ denotes the solution of (34). For each k , at least one of the constraints in cMPC simplified must be active, but all four constraints cannot be active (unless $S_{k|i} = \{0\}$, in which case $\bar{s}_{k|i} = \underline{s}_{k|i}$), so at least one multiplier is non-zero and at least one is zero. Therefore $\frac{\partial J_{\text{cMPC}}}{\partial \bar{s}_{k|i}} |_{\mathbf{z}_i^{*,c}}$ or $\frac{\partial J_{\text{cMPC}}}{\partial \underline{s}_{k|i}} |_{\mathbf{z}_i^{*,c}}$ is zero in (36c), (36d) for each k ; of these, the nonzero gradient is equal to $\frac{\partial J_{\text{cMPC}}}{\partial x_{k|i}} |_{(\mathbf{x}_i^r, \mathbf{c}_i^r)}$ due to the definition of J_{cMPC} , where $\mathbf{c}_{k|i}^r = c_{k|i}^{*,c} + c_{k|i}^0$ and $\mathbf{x}_{k|i}^r = \bar{s}_{k|i}^{*,c} + x_{k|i}^0$ or $\underline{s}_{k|i}^{*,c} + x_{k|i}^0$ depending on which constraint in (34) is active. From (36a)-(36d) we have

$$\begin{aligned}0 &= \frac{\partial J_{\text{cMPC}}}{\partial x_{0|i}} |_{(\mathbf{x}_i^r, \mathbf{c}_i^r)} + \alpha_{0|i}^{*,c} \frac{\partial \phi(x, c)}{\partial x} |_{\bar{z}_{0|i}^{*,c}} + \beta_{0|i}^{*,c} \frac{\partial \phi(x, c)}{\partial x} |_{\underline{z}_{0|i}^{*,c}} \\ &\quad - (\gamma_{0|i}^{*,c} + \mu_{0|i}^{*,c}) \frac{\partial \phi(x_{0|i}^0, c_{0|i}^0)}{\partial x_{0|i}^0} \quad (37) \\ 0 &= \frac{\partial J_{\text{cMPC}}}{\partial x_{k|i}} |_{(\mathbf{x}_i^r, \mathbf{c}_i^r)} + \alpha_{k|i}^{*,c} \frac{\partial \phi(x, c)}{\partial x} |_{\bar{z}_{k|i}^{*,c}} + \beta_{k|i}^{*,c} \frac{\partial \phi(x, c)}{\partial x} |_{\underline{z}_{k|i}^{*,c}} \\ &\quad - (\gamma_{k|i}^{*,c} + \mu_{k|i}^{*,c}) \frac{\partial \phi(x_{k|i}^0, c_{k|i}^0)}{\partial x_{k|i}^0} - (\alpha_{k-1|i}^{*,c} + \beta_{k-1|i}^{*,c} - \mu_{k-1|i}^{*,c} - \gamma_{k-1|i}^{*,c}). \quad (38)\end{aligned}$$

By Theorem 2, the cMPC iteration converges to a fixed point with $\mathbf{c}_i^* = 0$, $\mathbf{S}_i^* = \{\{0\}, \dots, \{0\}\}$, which we denote $(0, \{0\}, \mathbf{x}_i^0, \mathbf{u}_i^0)$. As this is optimal for the cMPC RHOCp, the simplified cMPC KKT conditions must be satisfied. Note that $(\mathbf{x}_i^r, \mathbf{c}_i^r) = (\mathbf{x}_i^0, \mathbf{c}_i^0)$ if $\mathbf{S}_i^* = \{\{0\}, \dots, \{0\}\}$ and $\mathbf{c}_i^* = 0$; hence comparing (35a) with (37), (35b) with (38), and (35c) with (36e) for $(\mathbf{x}_i^{*,n}, \mathbf{c}_i^{*,n}) = (\mathbf{x}_i^0, \mathbf{c}_i^0)$, it can be seen that they are equivalent if $\lambda_{k|i}^{*,n} = \alpha_{k|i}^{*,c} + \beta_{k|i}^{*,c} - \gamma_{k|i}^{*,c} - \mu_{k|i}^{*,c}$ for all k . Furthermore Theorem 1 ensures (35d) is satisfied by $(\mathbf{x}_i^{*,n}, \mathbf{c}_i^{*,n}) = (\mathbf{x}_i^0, \mathbf{c}_i^0)$. Hence the KKT conditions for cMPC and nMPC RHOCps are equivalent at the cMPC optimum. \square



Yana Lishkova received an integrated M.Eng. degree in Aerospace Engineering and Instrumentation & Control from the University of Cambridge. She is currently a D.Phil. student at the Oxford University Control Group, researching novel methods for simulation, optimization, and control of multirate systems. Applications include safety-critical problems in the aerospace domain. Her research is supported by an EPSRC Research Studentship, the Amelia Earhart Fellowship and the EPSRC Doctoral Prize Award.



Mark Cannon received the M.Eng. (1993) and D.Phil (1998) degrees from the University of Oxford, and M.S. (1995) from the Massachusetts Institute of Technology. He is currently Associate Professor of Engineering Science at Oxford University and a Fellow of St. John's College. His research interests include robust and stochastic predictive control, data-driven control and optimization, adaptive control, and aerospace, automotive and biomedical engineering applications.