Input Constraint Sets for Robust Regulation of Linear Systems

Sampath Kumar Mulagaleti, Alberto Bemporad, *Fellow, IEEE*, Mario Zanon, *Member, IEEE*

***Abstract*— In robust control under state constraints the set of admissible inputs is usually considered as given, under the assumption that the actuators have been already designed. However, if the input set is too small any controller will fail in stabilizing the closed-loop system while satisfying all prescribed constraints for some initial states of interest, or vice versa the chosen actuators may be over-sized. To handle this issue, in this paper we address the problem of computing the smallest input constraint set such that the closed-loop system is stabilizable from a prescribed set of initial states while respecting all constraints. We focus our attention on linear systems with additive disturbances, and develop the algorithm based on recursive feasibility of robust model predictive control. We demonstrate the results using numerical examples, in which we consider different metrics for the input constraint set selection.**

I. Introduction

Constrained systems with unknown but bounded disturbances can be robustly stabilized using several control strategies, e.g., Robust Model Predictive Control (RMPC) schemes [1]–[3]. The main components that are required to synthesize controllers using these schemes are: a) a model of the system to control, including the descriptions of the state constraints and model uncertainty set; b) tuning parameters defining the cost function; c) a set of feasible inputs (the input constraint set). Then, the RMPC controller solves an online optimization problem to compute inputs that belong to the input constraint set, such that the system is stabilized from a given set of initial conditions. Component (a) can be obtained by using a system identification procedure, e.g., [4]–[6]; component (b) can be obtained by some tuning procedure, e.g., by preference-based calibration [7], or, if a desired linear feedback is available, through a controller matching procedure, e.g., [8], [9]. In this paper, we tackle the computation of component (c), i.e., the input constraint set.

Typically, the input constraint set is directly characterized by the parameters that describe the technical specifications of the actuators. For example, pump parameters such as impeller size and motor capacity dictate the set of flow-rate inputs [10]. These parameters are usually selected during the system design phase by optimizing a criterion that captures various specifications such as costs, reliability, performance, etc. Hence, the procedures employed in the system design phase dictate the input constraints enforced in the control design phase. Given a set of input constraints, the set of initial-conditions from which the system can be robustly regulated is called the Maximal Robust Control Invariant (MRCI) set [11]–[13]. Then, given a desired set of initial-conditions of the system, the input constraint set could be *undersized*, i.e., the initial-condition set is not included in the MRCI set, or *oversized*, i.e., a potentially smaller input constraint set could be used to stabilize the system from those initial-conditions. In this paper, we present a methodology to bridge the system design and control design phases by computing an *optimally-sized* input constraint set that explicitly accounts for the stabilizability requirements, i.e., it computes the actuator parameters that optimize the selection criterion used in the system design phase, while ensuring that the MRCI set corresponding to the resulting input constraint set contains a desired set of initial states. We also present a simple extension to the proposed methods to account for the modification in the system dynamics that can accompany actuator selection, thus enhancing its practicality as an engineering tool. In the rest of this paper, we refer to the selection criterion as the input constraint set *size*, which is meant in an extended sense as a user-defined optimality metric. We consider linear time-invariant systems of the form

$$x(t+1) = Ax(t) + Bu(t) + B_w w(t), \tag{1}$$

with state $x \in \mathbb{R}^{n_x}$, control input $u \in \mathbb{R}^{n_u}$, bounded disturbance $w \in \mathcal{W} \subset \mathbb{R}^{n_w}$, and subject to state constraints $x \in \mathcal{X}$. In order to design the actuators for a given set $\Omega \subset \mathbb{R}^n$ of initial states, one can formulate the following problem:

***Problem** 1: Find the **smallest set** $\mathcal{U}$ of input constraints required to **robustly** regulate $x(t)$, i.e., to guarantee constraint satisfaction $x \in \mathcal{X}$ with inputs $u \in \mathcal{U}$ for all possible disturbances $w \in \mathcal{W}$, **from all** initial states $x(0) \in \Omega$.*

Note that the existence of a solution to Problem 2 entails the existence of a control law κ with a corresponding nonempty Robust Positive Invariant (RPI) set $\mathcal{X}_\kappa \supseteq \Omega$, as we will clarify in Section II. Similar problems have been tackled previously in the context of actuator selection: in [14], the smallest number of actuators required to drive all $x(0) \in \Omega$ to some subset of the state-space is computed for a diagonal matrix B; in [15], the minimal actuator set problem is solved with an additional upper bound on the control effort required to reach the desired subset; in [16], an algorithm is presented to perform the actuator selection online, in a model predictive control fashion. However, none of these works consider systems with uncertainties and state constraints. The closest approach to the one we discuss was presented in [17], in which set-invariance properties were used to formulate an actuator-saturation design problem. Similar to this paper, it is assumed that a set of desired initial conditions is given a priori, and safe actuator saturation limits are computed. However, differently from our approach, it is assumed that the system is equipped with a static feedback law (requiring to work with positive invariant sets, rather than control invariant sets), and both uncertainty and state constraints can not be included in the formulation of the problem. This work is essentially different from the one we present in [18], in which we use the minimum positive invariant set to compute an input constraint set by solving a reachability problem, as opposed to the stabilizability problem we tackle in this paper.

Contribution. Unfortunately, as we will discuss in Section II, solving Problem 1 might be difficult, so we propose to rely on RMPC to define κ and reformulate the problem in the following more tractable, though slightly conservative, way:

***Problem** 2: Find the **smallest** input constraint set $\mathcal{U}$ required to **guarantee recursive feasibility** of the RMPC scheme presented in [1] **for all** $x(0) \in \Omega$.*

This second formulation is justified by the observation that, in practice, the technique of choice for enforcing robust invariance is often RMPC. To address Problem 2, we formulate an optimization problem by using tools of set invariance and provide an algorithm to solve it. We prove that the algorithm always terminates, and analyze its properties that are of practical significance.

This paper is organized as follows. In Section II we formulate the problem introduced above of determining the input constraint

IMT School for Advanced Studies Lucca, Italy. (Email: s.mulagaleti, alberto.bemporad, mario.zanon@imtlucca.it.)

set. Then, in Section III we develop an algorithm to compute the set $\mathcal{U}$, and present its relevant properties. In Section IV, we discuss the implementation of the developed algorithm. Finally in Section V we present three numerical examples, with the first to illustrate some basic properties of the methods, the second showing an application of the methods to perform actuator selection with practical considerations, and the third to show the scalability of the proposed methodology.

Notation: The set $\mathcal{B}_p^n := \{x \in \mathbb{R}^n : \|x\|_p \leq 1\}$ denotes the p-norm ball in $\mathbb{R}^n$. The Minkowski set addition is defined as $\mathcal{X} \oplus \mathcal{Y} := \{x + y : x \in \mathcal{X}, y \in \mathcal{Y}\}$. If $\mathcal{Y} \subset \mathcal{X}$, then set subtraction is defined as $\mathcal{X} \ominus \mathcal{Y} := \{x : \{x\} \oplus \mathcal{Y} \subset \mathcal{X}\}$. Given a set $\mathcal{X}$ we denote its interior as $\text{int}(\mathcal{X})$. Given two matrices $T, S \in \mathbb{R}^{n\times m}$, T_i denotes row i of matrix T, and $T \leq S$ denotes element-wise inequality. If T is a square matrix, $\rho(T)$ denotes its spectral radius. The symbols $\mathbf{1}$, $\mathbf{0}$, and $\mathbf{I}$ denote all-ones, all-zeros and identity matrix respectively. The set $\mathbb{I}_m^n := \{m, \dots, n\}$ is the set of natural numbers between m and n. Given $v \in \mathbb{R}^n$, $S \in \mathbb{R}^{n\times n}$, we define $\|v\|_S^2 := v^\top S v$.

II. Problem formulation

In this section, we formulate Problems 1 and 2 by recalling the concepts of control invariance from [11], and the tube-based RMPC scheme from [1].

Formulation of Problem 1: Problem 1 can be formulated using the *Maximal Robust Control Invariant (MRCI)* set $\mathcal{X}_\infty$, which is such that [11, Definition 2.5] $\mathcal{X}_{\text{RCI}} \subseteq \mathcal{X}_\infty \in \mathbb{X}$, for all $\mathcal{X}_{\text{RCI}}$ satisfying

$$x \in \mathcal{X}_{\text{RCI}} \Rightarrow \begin{cases} x \in \mathcal{X}, \\ \exists\, u \in \mathcal{U} \,|\, Ax + Bu + B_w w \in \mathcal{X}_{\text{RCI}}, \ \forall\, w \in \mathcal{W} \end{cases} \tag{2}$$

$$\mathbb{X} := \{\mathcal{X}_{\text{RCI}} \,|\, (2) \text{ holds}\}. \tag{3}$$

This implies that for every initial state $x(0) \in \mathcal{X}_\infty$ of system (1) and every time instant $N \geq 1$, there exists an admissible control sequence $u(k) \in \mathcal{U}$, $k = 0, \dots, N-1$ resulting in an admissible state sequence $x(k) \in \mathcal{X}_\infty$, $k = 0, \dots, N$ for all possible disturbances $w(k) \in \mathcal{W}$. Then, Problem 1 can be formulated as

$$\min_{\mathcal{U}} \ f(\mathcal{U}) \quad \text{s.t.} \quad \Omega \subseteq \mathcal{X}_\infty(\mathcal{U}), \tag{4}$$

where $f(\mathcal{U})$ is, e.g., a measure of the size of the input constraint set $\mathcal{U}$, and we made the dependence of $\mathcal{X}_\infty$ on $\mathcal{U}$ explicit. Note that, if $\mathcal{X}_\infty(\mathcal{U})$ is known, one can define a control law κ as a function which, for each $x \in \mathcal{X}_\infty$, selects any input u which satisfies (2). Then, the associated maximum RPI (MRPI) set satisfies $\mathcal{X}_\kappa = \mathcal{X}_\infty$.

Problem 1 guarantees that state constraints can be robustly satisfied at all times and the system is regulated to $\mathcal{X}_\infty$. However, solving (4) is difficult, since one needs to solve an optimization problem with variable $\mathcal{U}$ and the MRCI set as a function of $\mathcal{U}$. Additionally, one is often interested in regulating the state of system (1) to a smaller target neighborhood of the origin. A popular technique that allows one to achieve this objective is RMPC. In RMPC, robust invariance is enforced by requiring that the RMPC control law is able to steer all initial states $x(0)$ to a target RPI set [11, Definition 2.1] in N-steps. This implicitly defines a second larger RPI set (the feasible domain of RMPC) which approximates the MRCI set $\mathcal{X}_\infty$, but is by definition no larger, and hence a certain degree of conservativeness is introduced. Note that, rather than constructing the MRPI set first and defining a control law κ next, this second approach amounts to the opposite, which defines the mechanism used to formulate Problem 2.

Formulation of Problem 2: In this paper, we present the formulation for the tube-based RMPC scheme from [1], which is constructed using the disturbance-free nominal system

$$\hat{x}(t+1) = A\hat{x}(t) + B\hat{u}(t), \tag{5}$$

and a parametrized system input

$$u(t) = \hat{u}(t) - K(x(t) - \hat{x}(t)), \tag{6}$$

where K is a static feedback gain. Defining $A_K := A - BK$, the following standing assumptions are made.

Assumption 1: a) the static gain K is such that $\rho(A_K) < 1$; b) the sets $\mathcal{X}$ and $\mathcal{W}$ are compact, convex, and have the origin in their nonempty interiors. □

Defining $\Delta x := x - \hat{x}$ and $\Delta u := u - \hat{u}$, an RPI set $\mathcal{E}$ is then computed for the uncertain system $\Delta x(t+1) = A_K \Delta x(t) + B_w w(t)$, which satisfies the property $\Delta x(t) \in \mathcal{E}$ implies $\Delta x(t+1) \in \mathcal{E}$ for all $w(t) \in \mathcal{W}$. This property implies that if the current system state $x(t) \in \{\hat{x}(t)\} \oplus \mathcal{E}$, and an input is computed as in (6), then the successive system state satisfies $x(t+1) \in \{\hat{x}(t+1)\} \oplus \mathcal{E}$, i.e., the system state always belongs to the *uncertainty tube* $\{\hat{x}\} \oplus \mathcal{E}$. Then, from (6) the system input always belongs to the set $\{\hat{u}\} \oplus -K\mathcal{E}$.

Since the uncertainty tubes define all possible future evolutions of system (1), the RMPC scheme provides robust constraint satisfaction using the tightened constraint sets $\mathcal{X} \ominus \mathcal{E}$ and $\mathcal{U} \ominus -K\mathcal{E}$. These sets guarantee that, if $\hat{x}(t) \in \mathcal{X} \ominus \mathcal{E}$ and $\hat{u}(t) \in \mathcal{U} \ominus -K\mathcal{E}$, then the state and input satisfy $x(t) \in \mathcal{X}$ and $u(t) \in \mathcal{U}$, and $x(t+1) \in \mathcal{X}$.

Assumption 2: The RPI set $\mathcal{E}$ is small enough such that the origin belongs to the nonempty interior of the tightened constraint sets, i.e.,

$$\mathbf{0} \in \text{int}(\mathcal{X} \ominus \mathcal{E}), \qquad \mathbf{0} \in \text{int}(\mathcal{U} \ominus -K\mathcal{E}). \tag{7}$$

The nominal input $\hat{u}(t)$ is computed at each time instant by measuring $x(t)$ and solving the following optimization problem [1]:

$$\min_{\mathbf{z}} \sum_{s=t}^{t+N-1} \|\hat{x}(s)\|_Q^2 + \|\hat{u}(s)\|_R^2 + \|\hat{x}(t+N)\|_P^2 \tag{8a}$$

$$\text{s.t.} \quad \{\hat{x}(t)\} \oplus \mathcal{E} \ni x(t), \tag{8b}$$

$$\hat{x}(s+1) = A\hat{x}(s) + B\hat{u}(s), \qquad s \in \mathbb{I}_t^{t+N-1}, \tag{8c}$$

$$\hat{x}(s) \in \mathcal{X} \ominus \mathcal{E}, \qquad s \in \mathbb{I}_{t+1}^{t+N-1}, \tag{8d}$$

$$\hat{u}(s) \in \mathcal{U} \ominus -K\mathcal{E}, \qquad s \in \mathbb{I}_t^{t+N-1}, \tag{8e}$$

$$\hat{x}(t+N) \in \mathcal{T}_{\text{f}}, \tag{8f}$$

where $\mathbf{z} := \{\hat{x}(t), \dots, \hat{x}(t+N), \hat{u}(t), \dots, \hat{u}(t+N-1)\}$. The parameters K and P are chosen to satisfy the Discrete Algebraic Riccati Equation that solves the LQR problem for the nominal system (5) with positive definite matrices Q and R. The terminal set $\mathcal{T}_{\text{f}}$ in (8f) is chosen to be a positive invariant (PI) set satisfying

$$A_K \mathcal{T}_{\text{f}} \subseteq \mathcal{T}_{\text{f}} \subseteq \mathcal{X} \ominus \mathcal{E}, \qquad -K\mathcal{T}_{\text{f}} \subseteq \mathcal{U} \ominus -K\mathcal{E}, \tag{9}$$

which ensures that with $\hat{u}(s) = -K\hat{x}(s)$ for all $s \geq t+N$, we have $\hat{x}(s) \in \mathcal{T}_{\text{f}}$. Denoting the optimal solution of problem (8) by $\mathbf{z}_* := \{\hat{x}_*(t), \dots, \hat{x}_*(t+N), \hat{u}_*(t), \dots, \hat{u}_*(t+N-1)\}$, the input $u(t) := \hat{u}_*(t) - K(x(t) - \hat{x}_*(t))$ is applied to the plant.

The set of all initial states $x(0)$ of system (1) from which the RMPC controller is recursively feasible and stabilizing [1, Proposition 2] is the *N-step controllable set* [11, Definition 2.3] defined as

$$\mathcal{K}_N(\mathcal{U}, \mathcal{T}_{\text{f}}) := \hat{\mathcal{K}}_N(\mathcal{U}, \mathcal{T}_{\text{f}}) \oplus \mathcal{E}, \tag{10}$$

where $\hat{\mathcal{K}}_N(\mathcal{U}, \mathcal{T}_{\text{f}})$ is the *N-step nominal controllable set*, i.e., the set of all initial states $\hat{x}(0)$ of the nominal system (5) for which there exists an admissible nominal control sequence that drives it to the PI terminal set $\mathcal{T}_{\text{f}}$ in N-steps with an admissible nominal state evolution. Mathematically, it is defined as

$$\begin{aligned} \hat{\mathcal{K}}_N(\mathcal{U}, \mathcal{T}_{\text{f}}) := \{\hat{x}(0) : \exists\, \hat{u}(t) \in \mathcal{U} \ominus -K\mathcal{E}, \ t \in \mathbb{I}_0^{N-1}, \ \hat{x}(t) \in \mathcal{X} \ominus \mathcal{E}, \\ \hat{x}(t+1) = A\hat{x}(t) + B\hat{u}(t), \ \forall t \in \mathbb{I}_0^{N-1}, \ \hat{x}(N) \in \mathcal{T}_{\text{f}} \ \}. \end{aligned} \tag{11}$$

By (10) and (11), the N-step controllable set $\mathcal{K}_N(\mathcal{U}, \mathcal{T}_\mathrm{f})$ is an RPI set for the RMPC scheme. Hence, by fixing the control law κ to be the RMPC scheme, we approximate the MRCI set $\mathcal{X}_\infty$ by the RPI set $\mathcal{K}_N(\mathcal{U}, \mathcal{T}_\mathrm{f})$ (feasible domain of RMPC). Since all initial states $x(0)$ of system (1) belonging to this set can be driven to the smaller target RPI set $x(N) \in \mathcal{T}_\mathrm{f} \oplus \mathcal{E}$ with an admissible state and input evolution, a desired set of initial conditions Ω is stabilizable if the inclusion

$$\Omega \subseteq \mathcal{K}_N(\mathcal{U}, \mathcal{T}_\mathrm{f}) \tag{12}$$

holds. Based on this observation, Problem 2 that approximates Problem 1 can be formulated as

$$\min_{\mathcal{U}, N, K, \mathcal{T}_\mathrm{f}, \mathcal{E}} \; f(\mathcal{U}) \quad \text{s.t.} \quad (7), (9), (12). \tag{13}$$

In this paper, we assume that the feedback gain K and RPI set $\mathcal{E}$ are given a priori and optimize over $\mathcal{U}$, $\mathcal{T}_\mathrm{f}$ and N.

Conservativeness of the proposed approach: The requirement to drive $x(N)$ to a target RPI subset of the MRCI set, and the input parametrization in (6) with a static linear feedback law introduce conservativeness into (13) as compared to (4). Moreover, additional conservativeness is introduced by fixing K and $\mathcal{E}$, since they affect both the uncertainty tube and the PI terminal set $\mathcal{T}_\mathrm{f}$. Jointly solving (13) also over these variables is a subject of future research. We note that (13) can also be formulated for the RMPC scheme proposed in [2]. Since the scheme uses exact uncertainty tubes, $\mathcal{E}$ is not present in the resulting formulation. This reduces conservativeness in the proposed approach, as we demonstrate in Example V-A.

Remark 1: The formulation in (8) assumes full knowledge of the state $x(t)$. In case only an estimate is available, one can enlarge the RPI set $\mathcal{E}$ to account for the estimation error, provided it is bounded. Further details of this formulation can be found in [19]. □

III. Computation of Sets $\mathcal{U}$ and $\mathcal{T}_\mathrm{f}$

We now discuss the computation of the smallest set $\mathcal{U}$ and a corresponding terminal set $\mathcal{T}_\mathrm{f}$ that solves Problem 2. To this end, we parameterize $\mathcal{U}$ with a finite-dimensional vector $\epsilon \in \mathbb{R}^{n_\epsilon}$ such that $\mathcal{U} = \mathbb{U}(\epsilon)$, and define the size (or any other measure to be minimized) of $\mathcal{U}$ as the scalar function $\boldsymbol{f}(\epsilon) : \mathbb{R}^{n_\epsilon} \to \mathbb{R}$. We note that $\boldsymbol{f}(\epsilon) = f(\mathbb{U}(\epsilon))$, where $f(\cdot)$ is used to formulate (13).

Assumption 3: Set $\mathbb{U}(\epsilon)$ and function $\boldsymbol{f}(\epsilon)$ satisfy:

(a) $\mathbb{U}(\epsilon)$ is compact and convex for all ϵ; moreover, for all $\delta \geq 0$, there exists an ϵ such that $\delta\mathcal{B}_\infty^{n_u} \subseteq \mathbb{U}(\epsilon)$;

(b) The value of $\boldsymbol{f}(\epsilon)$ is a measure of the set $\mathbb{U}(\epsilon)$, i.e., $\mathbb{U}(\epsilon^1) \subset \mathbb{U}(\epsilon^2) \implies 0 \leq \boldsymbol{f}(\epsilon^1) < \boldsymbol{f}(\epsilon^2) < \infty$. □

Assumption 3(a) ensures that there always exists a parameter ϵ such that $\mathbb{U}(\epsilon)$ includes any compact subset of $\mathbb{R}^{n_u}$ containing the origin. Then, in Assumption 3(b), we ensure that $\boldsymbol{f}(\epsilon)$ is well defined for every $\mathbb{U}(\epsilon)$, and the inequalities enforce strict monotonicity properties on $\boldsymbol{f}(\epsilon)$ with respect to $\mathbb{U}(\epsilon)$. We provide a clarifying example next.

Example: One possible parametrization of the input constraint set is the polytope $\mathbb{U}(\epsilon) = \{u : F^u u \leq \epsilon\}$. Then, examples of the size function that satisfy Assumption 3(b) are: (a) If a vector $\mathrm{c} > \mathbf{0}$ is such that each c_i denotes the unit cost of actuation in direction i, then $\boldsymbol{f}(\epsilon) = \mathrm{c}^\top \epsilon$ denotes the total cost of selecting the input constraint set $\mathbb{U}(\epsilon)$; (b) Defining the ellipsoid $\mathcal{M} := \{u : u^\top R u \leq 1\}$, the size function $\boldsymbol{f}(\epsilon) = \min\{\alpha : \mathbb{U}(\epsilon) \subseteq \alpha\mathcal{M}\}$ denotes the upper bound to the largest energy input $u^\top R u$ into system (1).

In the sequel, we propose an algorithm to compute the parameter ϵ such that $\mathbb{U}(\epsilon)$ satisfies the requirements (7),(9),(12), and minimizes $\boldsymbol{f}(\epsilon)$. To this end, we formulate an optimization problem in Subsection A that is equivalent to (13). In Subsection B, we develop an algorithm to solve the optimization problem, and discuss its properties in Subsection C. In Subsection D, we analyze the variation of the size of the optimal input constraint set with the horizon length N.

A. Input Constraint Set Computation Problem

For the finite dimensional parametrization $\mathcal{U} = \mathbb{U}(\epsilon)$ of the input constraint set, we write the N-step controllable set defined in (10) as $\mathbb{K}_N(\epsilon, \mathcal{T}_\mathrm{f}) := \mathcal{K}_N(\mathbb{U}(\epsilon), \mathcal{T}_\mathrm{f})$. Then, we define the tightened constraint admissible set $\mathbb{C}(\epsilon) := \{\hat{x} : \hat{x} \in \mathcal{X} \ominus \mathcal{E},\ -K\hat{x} \in \mathbb{U}(\epsilon) \ominus -K\mathcal{E}\}$, such that system (5) with nominal input $\hat{u} = -K\hat{x}$ satisfies

$$\hat{x}(t+1) = A_K \hat{x}(t), \qquad \hat{x}(t) \in \mathbb{C}(\epsilon) \implies \hat{u}(t) \in \mathbb{U}(\epsilon). \tag{14}$$

Based on these sets, consider the following optimization problem that is equivalent to (13) for a fixed K, $\mathcal{E}$ and N:

$$(\hat{\epsilon}^N, \hat{\mathcal{T}}_\mathrm{f}) := \arg\min_{\epsilon, \mathcal{T}} \; \boldsymbol{f}(\epsilon) \tag{15a}$$

$$\text{s.t.} \quad \Omega \subseteq \mathbb{K}_N(\epsilon, \mathcal{T}), \tag{15b}$$

$$A_K \mathcal{T} \subseteq \mathcal{T} \subseteq \mathbb{C}(\epsilon), \tag{15c}$$

$$\delta\mathcal{B}_\infty^{n_u} \subseteq \mathbb{U}(\epsilon) \ominus -K\mathcal{E}, \tag{15d}$$

where constraint (15c) ensures that $\mathcal{T}$ is a PI subset of $\mathbb{C}(\epsilon)$, thus satisfying (9); constraint (15b) is equivalent to (12); constraint (15d) formulated with some scalar $\delta > 0$ ensures that (7) is satisfied.

Since the problem defined in (15) involves optimizing over PI sets $\mathcal{T}$, solving it directly can be computationally challenging. To tackle this issue, we introduce the i-step feedback admissible set $\mathbb{O}_i(\epsilon) := \left\{\hat{x} : A_K^t \hat{x} \in \mathbb{C}(\epsilon),\ \forall t \in \mathbb{I}_0^i\right\}$, which is the set of initial states of system (14) that remain inside $\mathbb{C}(\epsilon)$ for i steps. Then, $\mathbb{O}_\infty(\epsilon)$ is the *Maximal Positive Invariant* (MPI) subset of $\mathbb{C}(\epsilon)$ [11, Definition 2.3]. Using this set, we propose to relax problem (15) by enforcing $\mathcal{T} = \mathbb{O}_\infty(\epsilon)$, thus formulating the problem:

$$\mathbf{P}^N : \begin{cases} \epsilon^N := \arg\min_{\epsilon} \; \boldsymbol{f}(\epsilon) & \text{(16a)} \\ \qquad \text{s.t.} \quad \Omega \subseteq \mathbb{K}_N(\epsilon, \mathbb{O}_\infty(\epsilon)), & \text{(16b)} \\ \qquad \delta\mathcal{B}_\infty^{n_u} \subseteq \mathbb{U}(\epsilon) \ominus -K\mathcal{E}. & \text{(16c)} \end{cases}$$

The constraint equivalent to (15c) is eliminated from the formulation of $\mathbf{P}^N$ since the inclusions $A_K\mathbb{O}_\infty(\epsilon) \subseteq \mathbb{O}_\infty(\epsilon) \subseteq \mathbb{C}(\epsilon)$ hold by construction [20] under Assumptions 1 and 2. In the following result, we show that $\mathbf{P}^N$ is not more conservative than problem (15), i.e., if $\hat{\epsilon}^N$ solves $\mathbf{P}^N$ then it must also solve (15).

Proposition 1: Suppose Assumptions 1, 2 and 3 hold. If problem (15) is feasible, then $\mathbf{P}^N$ is feasible and $\boldsymbol{f}(\epsilon^N) = \boldsymbol{f}(\hat{\epsilon}^N)$. □

Proof: Feasibility of problem (15) implies bounded solution under Assumption 3. This solution satisfies $\hat{\mathcal{T}}_\mathrm{f} \subseteq \mathbb{O}_\infty(\hat{\epsilon}^N)$, since $\mathbb{O}_\infty(\hat{\epsilon}^N)$ is the MPI subset of $\mathbb{C}(\epsilon^N)$ [11, Definition 2.2] under Assumptions 1 and 2. Hence, $\hat{\epsilon}^N$ is feasible for $\mathbf{P}^N$, which implies $\boldsymbol{f}(\epsilon^N) \leq \boldsymbol{f}(\hat{\epsilon}^N)$. The proof is concluded by noting that $\boldsymbol{f}(\hat{\epsilon}^N) \leq \boldsymbol{f}(\epsilon^N)$ since $(\epsilon^N, \mathbb{O}_\infty(\epsilon^N))$ is feasible for problem (15). ■

Remark 2: If Assumptions 1, 2 and 3 hold, and $\mathcal{U}$ is parametrized as $\mathbb{U}(\epsilon) = \{u : F^u u \leq \epsilon\}$, then the constraint set of problem (15) is convex. Then, if $\boldsymbol{f}(\epsilon)$ is chosen to be a convex function, (15) is a convex optimization problem. Moreover, if $\boldsymbol{f}(\epsilon)$ is a strictly convex function (for example, $\boldsymbol{f}(\epsilon) = \|\epsilon\|_2^2$), then the optimizer $\hat{\epsilon}^N$ is guaranteed to be unique. Since problem $\mathbf{P}^N$ is also convex, then $\epsilon^N = \hat{\epsilon}^N$ along with $\boldsymbol{f}(\epsilon^N) = \boldsymbol{f}(\hat{\epsilon}^N)$ if $\boldsymbol{f}(\epsilon)$ is strictly convex. □

Remark 3: In the formulation of $\mathbf{P}^N$, we assume that the dynamics of system (1) are unaffected by a change in input constraint set parameter ϵ. This assumption, however, might not be valid in certain scenarios. For example, a modification in the engine mass and inertia affect the dynamic properties of a car. In such cases, one can formulate constraint (16b) with the modified dynamical system $x(t+1) = Ax(t) + Bu(t) + B_w w(t) + g(x(t), u(t), w(t), \epsilon)$. The

Algorithm 1 Algorithm to solve $\mathbf{P}^N$ given $A, B, K, \mathcal{X}, \mathcal{E}, \Omega, N$

1: Initialize $i \geq 0$;
2: Solve $\mathbf{P}^{i,N}$ for $\epsilon^{i,N}$;
3: If $\mathbb{O}_i(\epsilon^{i,N})$ is PI, stop. Else, increment i, go to Step 2;
4: **return** $\mathcal{U} = \mathbb{U}(\epsilon^{i,N})$, $\mathcal{T}_{\mathrm{f}} = \mathbb{O}_\infty(\epsilon^{i,N})$.

development of structure exploiting approaches to tackle this problem is a subject of future research. A simple approach, that we present in Example V-C, models this modification as an increase in uncertainty by parametrizing the disturbance set $\mathcal{W}$ as $\mathbb{W}(\epsilon)$. This follows from the observation that $g(x, u, w, \epsilon)$ lies in a compact set for all $x \in \mathcal{X}$, $u \in \mathbb{U}(\epsilon)$ and $w \in \mathcal{W}$ under Assumptions 1 and 3. □

B. Solution Algorithm

We now present an iterative algorithm to solve problem $\mathbf{P}^N$. We require a tailored algorithm since the set $\mathbb{O}_\infty(\epsilon)$ formulating constraint (16b) is not known apriori. To this end, consider the variant $\mathbf{P}^{i,N}$ of $\mathbf{P}^N$ obtained by replacing the MPI set $\mathbb{O}_\infty(\epsilon)$ with the i-step feedback admissible set $\mathbb{O}_i(\epsilon)$ in constraint (16b). Problem $\mathbf{P}^{i,N}$ is related to problem $\mathbf{P}^N$ as follows: for every parameter ϵ satisfying constraint (16c), there exists a finite MPI set termination index [20, Theorem 4.1] given by

$$i^*(\epsilon) := \min\{i : A_K \mathbb{O}_i(\epsilon) \subseteq \mathbb{O}_i(\epsilon)\} < \infty, \tag{17}$$

such that $\mathbb{O}_i(\epsilon) = \mathbb{O}_\infty(\epsilon)$ for all $i \geq i^*(\epsilon)$. Labeling $\epsilon^{i,N}$ as the solution of $\mathbf{P}^{i,N}$, this implies that if $i \geq i^*(\epsilon^{i,N})$, then $\mathbb{O}_i(\epsilon^{i,N})$ is a PI set, and $\epsilon^{i,N}$ is a feasible solution to $\mathbf{P}^N$. Hence, we propose to solve a sequence of problems $\mathbf{P}^{i,N}$ for increasing values of i, and terminating the sequence at index $i = i_N$ at which the PI condition is satisfied. We summarize this procedure in Algorithm 1.

Computational considerations: We will discuss how to formulate $\mathbf{P}^{i,N}$ in practice for polyhedral sets in Section IV. In this case, the problem has linear constraints and a monotonic (possibly convex) cost, such that efficient algorithms can be deployed. The case of ellipsoidal sets is both more involved to analyze and more conservative, and is not discussed further in this paper for lack of space.

Remark 4: Algorithm 1 follows a reasoning similar to the recursive computation of the MPI set proposed in [20], [21]. Index i is incremented until the invariance condition is satisfied. The difference is that we also recursively compute the input constraint set along with the MPI set in order to solve $\mathbf{P}^N$. □

C. Feasibility, Convergence and Optimality of Algorithm 1

In this section, we show that Algorithm 1 solves $\mathbf{P}^N$. To this end, we will first formulate requirements on the initial-condition set Ω and horizon length N for $\mathbf{P}^{i,N}$ to be feasible. Then, we will show that Algorithm 1 terminates at some finite index i_N. Finally, we will show that $\mathbf{P}^N$ is solved at termination, i.e., $\boldsymbol{f}(\epsilon^N) = \boldsymbol{f}(\epsilon^{i_N,N})$.

1) Feasibility of $\mathbf{P}^{i,N}$: Problem $\mathbf{P}^{i,N}$ is feasible only if all initial-states $x(0) \in \Omega$ are controllable in N steps. In order to formalize this statement, we introduce the sets $\mathbb{O}_\infty(\infty)$ and $\mathbb{K}_N(\infty, \mathbb{O}_\infty(\infty))$, which we define using unconstrained inputs, i.e., $u \in \mathbb{R}^{n_u}$. The set $\mathbb{O}_\infty(\infty)$ is the MPI subset of $\mathbb{C}(\infty) = \mathcal{X} \ominus \mathcal{E}$ for system (14), and $\mathbb{K}_N(\infty, \mathbb{O}_\infty(\infty))$ is an N-step controllable set [11] with unconstrained inputs u. Using these sets, we formulate the following N-step controllability assumption:

Assumption 4: All $x(0) \in \Omega$ are included in the N-step unconstrained controllable set, i.e., $\Omega \subseteq \mathbb{K}_N(\infty, \mathbb{O}_\infty(\infty))$. □

Proposition 2: Suppose Assumptions 1, 2, 3, and 4 hold. Then problem $\mathbf{P}^{i,N}$ is feasible and bounded. □

Proof: Under Assumption 4, there exists a sequence of inputs $\{u_z(t), t \in \mathbb{I}_0^{N-1}\}$ such that $x_z(N) \in \mathbb{O}_\infty(\infty) \oplus \mathcal{E}$ from each $x_z(0) = z \in \Omega$. Under Assumption 3, there exists an ϵ satisfying $\{u_z(t) \in \mathbb{U}(\epsilon), \forall t \in \mathbb{I}_0^{N-1},\ z \in \Omega\}$ and $\mathbb{C}(\epsilon) = \mathcal{X} \ominus \mathcal{E}$. These conditions guarantee the existence of an $\epsilon < \infty$ such that, for all i, we have $\Omega \subseteq \mathbb{K}_N(\epsilon, \mathbb{O}_\infty(\infty)) \subseteq \mathbb{K}_N(\epsilon, \mathbb{O}_i(\epsilon))$. Hence, ϵ is feasible for $\mathbf{P}^{i,N}$, with boundedness imposed by constraint (16c). ■

2) Termination of Algorithm 1: If we characterize an index ι_δ that is the maximum value of the MPI set termination index $i^*(\epsilon)$ for all parameters ϵ satisfying constraint (16c), then for all indices $i \geq \iota_\delta$, the solution $\epsilon^{i,N}$ of problem $\mathbf{P}^{i,N}$ satisfies the PI condition $A_K \mathbb{O}_i(\epsilon^{i,N}) \subseteq \mathbb{O}_i(\epsilon^{i,N})$. Then, there exists a termination index $i_N \leq \iota_\delta$ for Algorithm 1. However, characterizing ι_δ is not computationally possible, since the set of all ϵ satisfying constraint (16c) is open. In the following result, we establish an upper bound to i_N that can, in fact, be computed. To this end, consider the set

$$\mathcal{C}_\delta := \left\{\hat{x} : \hat{x} \in \mathcal{X} \ominus \mathcal{E}, -K\hat{x} \in \delta\mathcal{B}_\infty^{n_u}\right\}, \tag{18}$$

which satisfies $\mathcal{C}_\delta \subseteq \mathbb{C}(\epsilon^{i,N})$ for all i, and the index

$$k_\delta := \min\{i : A_K^{i+1}(\mathcal{X} \ominus \mathcal{E}) \subseteq \mathcal{C}_\delta\}. \tag{19}$$

The existence of k_δ follows from Assumptions 1 and 2.

Proposition 3: Suppose Assumptions 1, 2, 3, and 4 hold, then Algorithm 1 terminates at an index $i_N \leq k_\delta$. □

Proof: The following inclusions hold for all i:

$$\mathbb{O}_i(\epsilon^{i,N}) \subseteq \mathbb{C}(\epsilon^{i,N}) \subseteq \mathcal{X} \ominus \mathcal{E}. \tag{20}$$

Then at the index k_δ, we have

$$A_K^{k_\delta+1}\mathbb{O}_{k_\delta}\left(\epsilon^{k_\delta,N}\right) \subseteq A_K^{k_\delta+1}\mathbb{C}\left(\epsilon^{k_\delta,N}\right) \subseteq A_K^{k_\delta+1}(X \ominus \mathcal{E}) \subseteq \mathcal{C}_\delta \subseteq \mathbb{C}\left(\epsilon^{k_\delta,N}\right)$$

from (18), (19), and (20). By definition of the feedback admissible set, the first and the last terms imply $\mathbb{O}_{k_\delta+1}(\epsilon^{k_\delta,N}) = \mathbb{O}_{k_\delta}(\epsilon^{k_\delta,N})$. Then, $A_K \mathbb{O}_{k_\delta}(\epsilon^{k_\delta,N}) \subseteq \mathbb{O}_{k_\delta}(\epsilon^{k_\delta,N})$ from [20, Theorem 2.2]. ■

Smaller values of the tuning factor δ result in a smaller set $\mathcal{C}_\delta$. This increases the upper bound k_δ to the termination index i_N of Algorithm 1, resulting in a larger number of iterations. However, from the formulation of $\mathbf{P}^{i,N}$, we see that a smaller value of δ results in a smaller lower bound on the optimal value of $\boldsymbol{f}(\epsilon^{i,N})$ (through constraint (16c)). Hence, δ dictates the trade-off between optimality and computational difficulty.

3) Solution to $\mathbf{P}^N$: We finally show that the termination of Algorithm 1 corresponds to the solution of $\mathbf{P}^N$, i.e., the optimal values coincide as $\boldsymbol{f}(\epsilon^{i_N,N}) = \boldsymbol{f}(\epsilon^N)$. We reason as follows : for all indices $i < i_N$, the PI condition is not satisfied, which implies $\epsilon^{i,N}$ is not feasible for $\mathbf{P}^N$. Hence, we must show that if $\mathbf{P}^{i,N}$ is solved for some $i > i_N$, then the optimal value $\boldsymbol{f}(\epsilon^{i,N})$ cannot be smaller than $\boldsymbol{f}(\epsilon^{i_N,N})$.

Proposition 4: Suppose Assumptions 1, 2, 3 and 4 hold, then $\boldsymbol{f}(\epsilon^N) = \boldsymbol{f}(\epsilon^{i_N,N})$. □

Proof: Since the inclusion $\mathbb{O}_{i+1}(\epsilon^{i,N}) \subseteq \mathbb{O}_i(\epsilon^{i,N})$ holds for all i, the solution $\epsilon^{i+1,N}$ of $\mathbf{P}^{i+1,N}$ is feasible for $\mathbf{P}^{i,N}$. Then, the optimal values are non-decreasing as $\boldsymbol{f}(\epsilon^{i,N}) \leq \boldsymbol{f}(\epsilon^{i+1,N})$. Hence, $\boldsymbol{f}(\epsilon^{i_N,N}) \leq \boldsymbol{f}(\epsilon^{i,N})$ for all $i > i_N$, thus concluding the proof. ■

Remark 5: In some cases, Ω might violate the N-step controllability condition. Then, we propose to solve the optimization problem

$$\tilde{\Omega} := \arg\min_{\tilde{\Omega}}\ \boldsymbol{d}(\Omega, \tilde{\Omega}) \ \text{ s.t. } \ \tilde{\Omega} \subseteq \mathbb{K}_N(\infty, \mathbb{O}_\infty(\infty)), \tag{21}$$

where $\boldsymbol{d}(\Omega, \tilde{\Omega})$ is some distance metric. Since (21) guarantees that $\tilde{\Omega}$ satisfies Assumption 4, the aforementioned properties of Algorithm 1 continue to hold for the projected initial-condition set $\tilde{\Omega}$. □

Remark 6: Since $\mathbb{O}_i(\epsilon^{i_N,N}) = \mathbb{O}_{i_N}(\epsilon^{i_N,N})$ for all $i \geq i_N$, the solution $\epsilon^{i_N,N}$ of $\mathbf{P}^{i_N,N}$ is feasible for all $\mathbf{P}^{i,N}$ with $i \geq i_N$. This implies $\boldsymbol{f}(\epsilon^{i,N}) = \boldsymbol{f}(\epsilon^{i_N,N})$ for all $i \geq i_N$. Hence, Algorithm 1 can be initialized at any index $i = i_{\text{init}} \geq 0$, and incremented in Step 3 with any $i_{\text{incr}} \geq 1$, i.e., $i \leftarrow i + i_{\text{incr}}$. Moreover, if $i_{\text{init}} = k_\delta$ from (19), then Algorithm 1 terminates in one iteration. □

D. Effect of the Horizon Length on the Input Constraint Set Size

In this section, we discuss the effect of the horizon length N on the optimal input constraint set size. In particular, we show that $\boldsymbol{f}(\epsilon^N)$ is monotonically non-increasing and convergent in N.

To this end, we use an auxiliary optimization problem $\tilde{\mathbf{P}}^N$ that computes the smallest input constraint set required to maintain the state of system (1) inside the constraint set $\mathcal{X}$ for N steps. It is formulated by replacing the target set $\mathbb{O}_\infty(\epsilon)$ in constraint (16b) by $\mathcal{X} \ominus \mathcal{E}$, such that $\mathbb{K}_N(\epsilon, \mathcal{X} \ominus \mathcal{E})$ is an *N-step admissible set* [11, Definition 2.11]. We label the solution of this problem as $\tilde{\epsilon}^N$.

Proposition 5: Suppose Assumptions 1, 2, 3, and 4 hold. Then, (i) $\boldsymbol{f}(\epsilon^N) \geq \boldsymbol{f}(\epsilon^{N+1})$; (ii) $\lim_{N\to\infty} \boldsymbol{f}(\epsilon^N)$ exists. □

Proof: For all ϵ satisfying constraint (16c), the N-step stabilizable and admissible sets satisfy the inclusions

$$\begin{aligned}\mathbb{K}_N(\epsilon, \mathbb{O}_\infty(\epsilon)) &\subseteq \mathbb{K}_{N+1}(\epsilon, \mathbb{O}_\infty(\epsilon)) \\ &\subseteq \mathbb{K}_{N+1}(\epsilon, \mathcal{X} \ominus \mathcal{E}) \subseteq \mathbb{K}_N(\epsilon, \mathcal{X} \ominus \mathcal{E}),\end{aligned}$$

from [11, Propositions 2.3,2.4]. The first inclusion implies ϵ^N is feasible for $\mathbf{P}^{N+1}$, hence $\boldsymbol{f}(\epsilon^N) \geq \boldsymbol{f}(\epsilon^{N+1})$. The remaining two inclusions respectively imply $\boldsymbol{f}(\tilde{\epsilon}^N) \leq \boldsymbol{f}(\epsilon^N)$ and $\boldsymbol{f}(\tilde{\epsilon}^N) \leq \boldsymbol{f}(\tilde{\epsilon}^{N+1})$ by the same reasoning. Hence, $\{\boldsymbol{f}(\epsilon^N)\}_N$ is a non-increasing sequence, that is lower bounded by the non-decreasing sequence $\{\boldsymbol{f}(\tilde{\epsilon}^N)\}_N$. Thus, finite limits $\lim_{N\to\infty} \boldsymbol{f}(\epsilon^N)$ and $\lim_{N\to\infty} \boldsymbol{f}(\tilde{\epsilon}^N)$ exist. ■

This result implies that the problem $\min_N \mathbf{P}^N$ which is equivalent to (13) can be solved by choosing a large enough value of N.

IV. Polytopic Implementation of Algorithm 1

In this section we discuss the implementation of Algorithm 1 using polytopic sets $\mathcal{X} := \{x : H^x x \leq h^x\}$ and $\mathcal{W} := \{w : F^w w \leq f^w\}$ satisfying Assumption 1(b) with $h^x > \mathbf{0}$ and $f^w \geq \mathbf{0}$. A feedback gain K is assumed to be computed apriori. Then, we compute a polytopic RPI set $\mathcal{E} := \{\Delta x : H^\Delta \Delta x \leq h^\Delta\}$ for the system $x(t+1) = A_K x(t) + B_w w(t)$ with established methods, e.g., those given in [22], [23]. Using $\mathcal{E}$, we tighten the state constraint set as $\mathcal{X} \ominus \mathcal{E} = \{\hat{x} : H^x \hat{x} \leq h^x - \bar{h}^x\}$, where each component $\bar{h}^x_j = \max_{\Delta x \in \mathcal{E}} H^x_j \Delta x$. We choose an input constraint set $\mathcal{U}$ parameterized as the polytope $\mathbb{U}(\epsilon) := \{u : F^u u \leq \epsilon\}$ and satisfying Assumption 3(a). Then, the tightened input constraint set is $\mathbb{U}(\epsilon) \ominus -K\mathcal{E} = \{\hat{u} : F^u \hat{u} \leq \epsilon^{\text{m}}(\epsilon) - \bar{\epsilon}\}$, where each component $\epsilon^{\text{m}}_j(\epsilon) := \max_{u \in \mathbb{U}(\epsilon)} F^u_j u$ and $\bar{\epsilon}_j := \max_{\Delta x \in \mathcal{E}} -F^u_j K \Delta x$. The function $\epsilon^{\text{m}}(\epsilon)$ is such that $\mathbb{U}(\epsilon^{\text{m}}(\epsilon))$ is in minimal hyperplane representation. We now show that $\epsilon^{\text{m}}(\epsilon^{i,N}) = \epsilon^{i,N}$.

Proposition 6: Suppose Assumptions 1, 2, 3 and 4 hold, then $\mathbb{U}(\epsilon^{i,N})$ is in minimal representation. □

Proof: We prove this result by contradiction. Suppose $\mathbb{U}(\epsilon^{i,N})$ is not in minimal representation. Then, there exists some $\epsilon < \epsilon^{i,N}$ such that $\mathbb{U}(\epsilon) = \mathbb{U}(\epsilon^{i,N})$, and $\mathbb{U}(\epsilon)$ is in minimal representation. By Assumption 3(b), $\boldsymbol{f}(\epsilon) < \boldsymbol{f}(\epsilon^{i,N})$ follows, which is a contradiction since $\boldsymbol{f}(\epsilon^{i,N})$ is the optimal value of $\mathbf{P}^{i,N}$. ■

This result permits us to tighten the input constraint set as $\mathbb{U}(\epsilon) \ominus -K\mathcal{E} = \{\hat{u} : F^u \hat{u} \leq \epsilon - \bar{\epsilon}\}$. Using these definitions, we write the i-step feedback admissible set as $\mathbb{O}_i(\epsilon) := \{\hat{x} : S^{[i]} \hat{x} \leq q^{[i]}(\epsilon)\}$, which we use to formulate problem $\mathbf{P}^{i,N}$ with a vertex notation of the initial-condition set $\Omega = \{x_0^{[k]}, k \in \mathbb{I}_1^{N_\Omega}\}$ as

$$\begin{aligned}\epsilon^{i,N} = \arg \min_{\epsilon, \mathbf{z}^{[k]} \forall k \in \mathbb{I}_1^{N_\Omega}} \; & \boldsymbol{f}(\epsilon) \qquad (22)\\ \text{s.t.} \;\; & H^\Delta x_0^{[k]} \leq h^\Delta + H^\Delta \hat{x}^{[k]}(0) \\ & \hat{x}^{[k]}(s+1) = A\hat{x}^{[k]}(s) + B\hat{u}^{[k]}(s), \qquad s \in \mathbb{I}_0^{N-1}, \\ & H^x \hat{x}^{[k]}(s) \leq h^x - \bar{h}^x, \qquad s \in \mathbb{I}_1^{N-1}, \\ & F^u \hat{u}^{[k]}(s) \leq \epsilon - \bar{\epsilon}, \qquad s \in \mathbb{I}_0^{N-1}, \\ & S^{[i]} \hat{x}^{[k]}(N) \leq q^{[i]}(\epsilon), \\ & \epsilon - \bar{\epsilon} \geq \delta\lambda,\end{aligned}$$

where $\mathbf{z}^{[k]} := \{\hat{x}^{[k]}(0), \ldots, \hat{x}^{[k]}(N), \hat{u}^{[k]}(0), \ldots, \hat{u}^{[k]}(N-1)\}$. The first constraint implies $x_0^{[k]} \in \hat{x}^{[k]}(0) \oplus \mathcal{E}$, and the last constraint indicates $\delta\mathcal{B}_\infty^{n_u} \subseteq \mathbb{U}(\epsilon) \ominus -K\mathcal{E}$, where $\lambda_j := \max_{u \in \mathcal{B}_\infty^{n_u}} F_j u$. The details of the formulation of $\mathbf{P}^{i,N}$ with a hyperplane notation $\Omega = \{x : R^\Omega x \leq r^\Omega\}$ is given in Appendix VII-A.

For the computed $\epsilon^{i,N}$, we verify if the set $\mathbb{O}_i(\epsilon^{i,N})$ is PI by checking for the existence of a matrix Λ^{p} satisfying

$$\Lambda^{\text{p}} \geq \mathbf{0}, \quad \Lambda^{\text{p}} q^{[i]}(\epsilon^{i,N}) \leq q^{[i]}(\epsilon^{i,N}), \quad \Lambda^{\text{p}} S^{[i]} = S^{[i]} A_K,$$

which are necessary and sufficient conditions for the inclusion $A_K \mathbb{O}_i(\epsilon^{i,N}) \subseteq \mathbb{O}_i(\epsilon^{i,N})$ to hold [24].

Remark 7: Since $\mathbf{P}^N$ explicitly minimizes the input constraint set size, the resulting closed-loop performance of the RMPC controller with sets $\mathcal{U} = \mathbb{U}(\epsilon^N)$ and $\mathcal{T}_{\text{f}} = \mathbb{O}_\infty(\epsilon^N)$ might be unsatisfactory. This can be ameliorated by formulating the objective function of $\mathbf{P}^N$ with a trade-off parameter $\kappa_{\text{p}} \geq 0$, e.g., as

$$\boldsymbol{d}(\epsilon, \mathbf{z}) := \boldsymbol{f}(\epsilon) + \kappa_{\text{p}} \sum_{k=1}^{N_\Omega} \left(\sum_{s=0}^{N-1} \left(\left\| \hat{x}^{[k]}(s) \right\|_Q^2 \right) + \left\| \hat{x}^{[k]}(N) \right\|_P^2 \right). \qquad (23)$$

Then, the optimal value of $\boldsymbol{d}$ is non-increasing in N. The formulation of an objective function similar to (23) if Ω is given in a hyperplane notation is a subject of future research. □

V. Numerical Examples

In this section we present three numerical examples, with the first aimed at illustrating the properties of the approach, the second being a practical example in which we use the proposed methods to size pneumatic actuators for a force control application, and the third demonstrating the scalability of problem $\mathbf{P}^{i,N}$. We choose affine and piecewise affine size functions for these examples. This choice results in a Linear Program (LP) or Mixed-Integer LP (MILP), respectively, which we solve using the CPLEX solver [25]. We remark that other choices of size functions can be used in their formulation.

A. Illustrative system

The goal of the following example is to illustrate some fundamental properties of the approach. Consider the unstable system with dynamics $x(t+1) = \begin{bmatrix} 1.4 & 1 \\ -1 & 0.1 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 3 \end{bmatrix} u(t) + w(t)$ and state constraints $x \in 5\mathcal{B}_\infty^2$. We equip the system with a feedback controller K which is the solution of the Riccati equation for matrices $Q = 0.01\mathbf{I}$ and $R = 1$. The disturbance set is $\mathcal{W} = 0.1\mathcal{B}_\infty^2$, for which we compute the RPI set $\mathcal{E}$ as a polytope with 200 facets using the method presented in [23]. The initial-condition set is $\Omega = \{x_{[k]}^0, k \in \mathbb{I}_1^4\}$, where $x_{[1]}^0 = [-4, 4]^\top$, $x_{[2]}^0 = [-4, 6]^\top$, $x_{[3]}^0 = [-6, 6]^\top$ and $x_{[4]}^0 = [-6, 4]^\top$. Since this set does not satisfy Assumption 4, we follow the discussion in Remark 3 and compute the set $\tilde{\Omega} = \left\{\tilde{x}_{[k]}^0, k \in \mathbb{I}_1^4\right\}$ by solving the problem in (21)

with the distance $\boldsymbol{d}\left(\Omega, \tilde{\Omega}\right) = \sum_{k=1}^{4} \left\| x_{[k]}^0 - \tilde{x}_{[k]}^0 \right\|_1$. This choice leads to problem (21) being an LP, and results in $\tilde{x}_{[1]}^0 = [-4, 4]^\top$, $\tilde{x}_{[2]}^0 = [-4, 5]^\top$, $\tilde{x}_{[3]}^0 = [-4.9876, 4.9983]^\top$ and $\tilde{x}_{[4]}^0 = [-5, 4]^\top$, satisfying Assumption 4.

We parametrize the input constraint set $\mathcal{U}$ using $\epsilon \in \mathbb{R}^2$ as the saturation $\mathbb{U}(\epsilon) = \{u : -\epsilon_2 \leq u \leq \epsilon_1\}$ with $\epsilon_1, \epsilon_2 \geq 0$, and use the size function $\boldsymbol{f}(\epsilon) = \epsilon_1 + \epsilon_2$. Then, $\mathbf{P}^{i,N}$ is also an LP. We select a tuning factor of $\delta = 10^{-4}$, for which the upper bound to the termination index (19) of Algorithm 1 is $k_\delta = 28$. The corresponding results are shown in Figure 1. We report that Algorithm 1 converges at index $i = 2$ for $N = 3, 5$, $i = 3$ for $N = 2, 4, 11$, $i = 5$ for $N = 8, 10, 13, 16, 18$, and $i = 6$ for all other values of N. In Figure 1-upper-left plot, we see that $\boldsymbol{f}(\epsilon^N)$ is non-increasing and convergent, and is lower bounded by $\boldsymbol{f}(\tilde{\epsilon}^N)$. This follows from Proposition 5. We also plot $\boldsymbol{f}(\underline{\epsilon}^N)$, which is obtained by formulating Problem $\mathbf{P}^N$ with recursive feasibility conditions in [2]. Since this formulation uses exact uncertainty tubes, the resulting $\boldsymbol{f}(\underline{\epsilon}^N)$ is less conservative than $\boldsymbol{f}(\epsilon^N)$ [26]. In the upper-right plot, we see that, despite $\boldsymbol{f}(\epsilon^N)$ being non-increasing, the sets $\mathbb{U}(\epsilon^N)$ are not nested, i.e., $\mathbb{U}(\epsilon^{N+1}) \not\subseteq \mathbb{U}(\epsilon^N)$. This is because $\boldsymbol{f}(\epsilon^{N+1}) < \boldsymbol{f}(\epsilon^N)$ does not necessarily imply $\mathbb{U}(\epsilon^{N+1}) \subset \mathbb{U}(\epsilon^N)$ as per Assumption 3(b). However, this inclusion holds at $N = 3, 5, 11, 13, 16, 18, 26$, at which the termination index of Algorithm 1 increases ([20, Theorem 4.1]). In order to demonstrate the effect of δ on the performance of the algorithm, we plot the optimal values $\boldsymbol{f}(\epsilon^N)$ in the upper-left plot when $\delta = 10^{-1}$ is chosen in Algorithm 1. As discussed in Section III-C.2, this choice results in conservative values of $\boldsymbol{f}(\epsilon^N)$. However, the termination index upper bound (19) is $k_\delta = 8$, thus resulting in reduced computational difficulty. We report that Algorithm 1 then converges at $i \leq 5$ for $N = 2, \ldots, 10$. The lower-left plot shows the terminal sets $\mathbb{O}_\infty(\epsilon^N) \oplus \mathcal{E}$, and the lower-right plot shows the feasible sets $\mathbb{K}_N(\epsilon^N, \mathbb{O}_\infty(\epsilon^N))$. We observe that these sets are not nested since they correspond to different input constraint sets. The terminal sets computed by Algorithm 1 are such that $\mathbb{K}_N(\epsilon^N, \mathbb{O}_\infty(\epsilon^N))$ are aligned in the direction of $\tilde{\Omega}$, thus minimizing the actuation effort required for stabilizing. The closed-loop trajectories with the RMPC controller for $N = 30$ are shown. We also formulate $\mathbf{P}^N$ with the objective in (23) with $\kappa_{\mathrm{p}} = 0.01$ to improve closed-loop performance, which we solve using Algorithm 1 for $N = 30$. The resulting nominal state cost $\sum_{k=1}^{4} \sum_{t=0}^{N_{\mathrm{sim}}} \hat{x}^{[k]}(t)^\top Q \hat{x}^{[k]}(t)$ for $N_{\mathrm{sim}} = 70$ reduces from 2.9447 to 2.8777, while the nominal input cost $\sum_{k=1}^{4} \sum_{t=0}^{N_{\mathrm{sim}}-1} \hat{u}^{[k]}(t)^\top R \hat{u}^{[k]}(t)$ increases from 0.7132 to 0.7524.

B. Actuator selection

The goal of the following example is to demonstrate a simple application of the methods presented in this paper to select a set of pneumatic actuators and corresponding compressors for the mass-spring-damper system shown in Figure 2: Actuators 1 and 2 are double-acting pneumatic cylinders, that provide force inputs to masses m_1 and m_3 respectively. The force acting on m_1 is $u_1 = A_1^{\mathrm{c}} P_{11}^{\mathrm{c}} - (A_1^{\mathrm{c}} - a_1^{\mathrm{c}}) P_{12}^{\mathrm{c}}$, where A_1^{c} and a_1^{c} are the cylinder bore and piston-rod cross-sectional areas respectively, and $(P_{11}^{\mathrm{c}}, P_{12}^{\mathrm{c}})$ are pressures acting on the left and right sides of the piston respectively. These pressures are given by the Pneumatic Valves module, which responds to a force command by adjusting the position of a 3/5-servo-valve [27]. This adjustment is performed to the pressure $\hat{P}_1$ of the airflow incoming from Compressor 1, which has the volumetric flowrate $q_1 = A_1^{\mathrm{c}} x_1^{\mathrm{v}}$ if $x_1^{\mathrm{v}} > 0$, and $q_2 = (A_1^{\mathrm{c}} - a_1^{\mathrm{c}}) x_1^{\mathrm{v}}$ otherwise (where x_1^{v} is the velocity of m_1). Hence, sizing the actuators for this system involves selecting the areas $(A_i^{\mathrm{c}}, a_i^{\mathrm{c}})$ of the actuators and

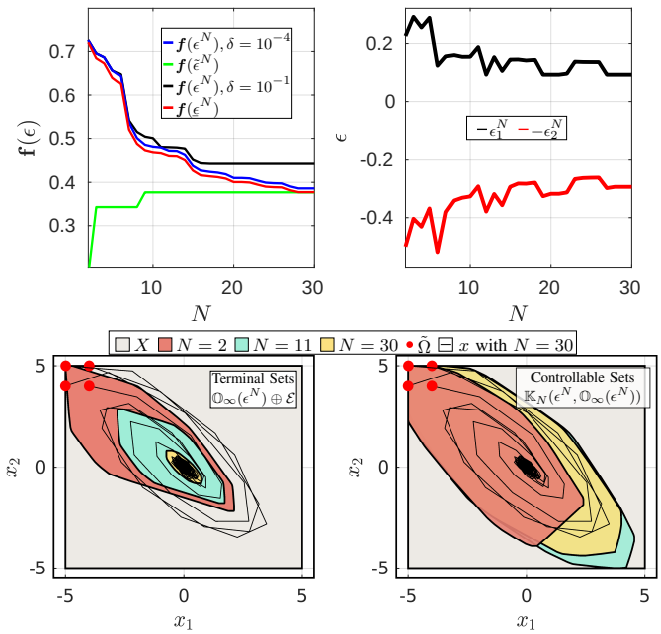

Fig. 1: Numerical results for Example 1. (Upper-left plot) Optimal input constraint set size $\boldsymbol{f}(\epsilon^N)$ computed by Algorithm 1, along with the lower bound $\boldsymbol{f}(\tilde{\epsilon}^N)$. These values satisfy the monotonically convergent properties discussed in Proposition 5. (Upper-right plot) Optimal input constraint set parameters ϵ^N computed by Algorithm 1. The sets are not necessarily nested. (Lower-left plot) Terminal sets computed by Algorithm 1 for $N = 2, 11, 30$ (Lower-right plot) Corresponding controllable sets for $N = 2, 11, 30$. The parameters ϵ^N are computed such that the terminal set and controllable set align towards $\tilde{\Omega}$.

compressor pressures $\hat{P}_i$, since they dictate the force limits on the system, and the compressor flowrate capacity required for actuation.

We consider a discretized model of the mass-spring-damper system, obtained using the forward Euler scheme with a time step of 0.1s. We assume that this model has additive disturbance inputs w acting on the velocity states, with $w \in W = 10^{-5} \mathcal{B}_\infty^3$. We equip the system with a static feedback gain K, that is the solution of the Riccati equation with matrices $Q = \mathbf{I}$ and $R = \mathbf{I}$. For this system, we compute an RPI set $\mathcal{E}$ as a polytope in $\mathbb{R}^6$ with 1812 faces using the method in [23]. The states of this system are constrained as $\mathcal{X} = \{x : |x_i^{\mathrm{p}}| \leq \hat{x}^{\mathrm{p}}, |x_i^{\mathrm{v}}| \leq \hat{x}_i^{\mathrm{v}},\ x_i^{\mathrm{p}} - x_{i+1}^{\mathrm{p}} \leq \hat{x}^{\mathrm{p}} - g^{\mathrm{p}}\}$, where g^{p} is the minimum gap between the masses. In order to parametrize the input constraint set, we assume that the piston-rod cross-sectional area is constrained as $a_i^{\mathrm{c}} = 0.1 A_i^{\mathrm{c}}$, and the compressors supply a constant pressure of $\hat{P}_i = 10^4$ N/m^2, following which we introduce the design vector $\epsilon := [A_1^{\mathrm{c}}\ A_2^{\mathrm{c}}]^\top$. Hence, $\mathcal{U}$ is parametrized as

$$\mathbb{U}(\epsilon) = \left\{ u : \begin{bmatrix} -(A_1^{\mathrm{c}} - a_1^{\mathrm{c}}) \hat{P}_1 \\ -A_2^{\mathrm{c}} \hat{P}_2 \end{bmatrix} \leq u \leq \begin{bmatrix} A_1^{\mathrm{c}} \hat{P}_1 \\ (A_2^{\mathrm{c}} - a_2^{\mathrm{c}}) \hat{P}_2 \end{bmatrix} \right\}. \tag{24}$$

Then, to compute the vector ϵ, we use the criterion $\boldsymbol{f}(\epsilon) = \boldsymbol{f}_{\mathrm{M}}(\epsilon) + \boldsymbol{f}_{\mathrm{P}}(\epsilon)$, with $\boldsymbol{f}_{\mathrm{M}}$ and $\boldsymbol{f}_{\mathrm{P}}$ defined as follows:

- Since each A_i^{c} represents material costs, it is natural to consider the criterion $\boldsymbol{f}_{\mathrm{M}}(\epsilon) := A_1^{\mathrm{c}} + A_2^{\mathrm{c}}$ to minimize.
- Since velocities $|x_i^{\mathrm{v}}| \leq \hat{x}_i^{\mathrm{v}}$, corresponding maximum compressor flowrates are $\hat{q}_i = A_i^{\mathrm{c}} \hat{x}_i^{\mathrm{v}}$. We assume that the compressors

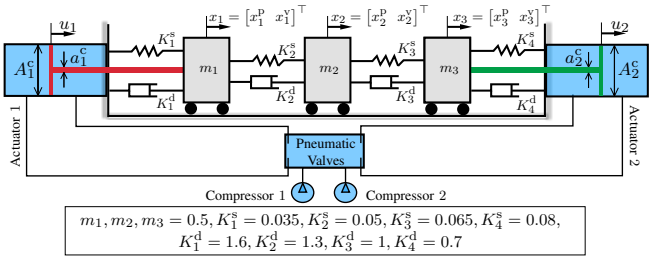

Fig. 2: Schematic of the mass-spring-damper system. The Pneumatic Valves module regulates the compressor flow pressure. Units: m_i in Kg, K_i^{s} in N/m, K_i^{d} in Ns/m, $A_i^{\mathrm{c}}, a_i^{\mathrm{c}}$ in m^2.

are priced according to their maximum flowrate capacities as

Type	1	2	3	4	5
Capacity ($\times 10^{-4}$m^3/s)	0.1	0.15	0.2	0.25	0.35
Price ($\times 10^{-2}$)	1.0	1.1	1.2	1.3	1.4

We encode this criterion as the piecewise constant function $\boldsymbol{f}_{\mathrm{P}}(\epsilon) := \beta_1 + \beta_2$, where $\beta_i = 1e^{-2}$ if $A_i^{\mathrm{c}}\hat{x}_i^{\mathrm{v}} \in [0, 0.1]e^{-4}$, $\beta_i = 1.1e^{-2}$ if $A_i^{\mathrm{c}}\hat{x}_i^{\mathrm{v}} \in (0.1, 0.15]e^{-4}$, $\beta_i = 1.2e^{-2}$ if $A_i^{\mathrm{c}}\hat{x}_i^{\mathrm{v}} \in (0.15, 0.2]e^{-4}$, $\beta_i = 1.3e^{-2}$ if $A_i^{\mathrm{c}}\hat{x}_i^{\mathrm{v}} \in (0.2, 0.25]e^{-4}$ and $\beta_i = 1.4e^{-2}$ if $A_i^{\mathrm{c}}\hat{x}_i^{\mathrm{v}} \in (0.25, 0.35]e^{-4}$. Then, the largest feasible area to ensure that the maximum feasible velocity can be reached is $\hat{A}_i^{\mathrm{c}} := 0.35e^{-4}/\hat{x}_i^{\mathrm{v}}$.

We also consider the loading effects of the actuators: compressibility of air modifies the stiffness and damping within the system [28]. Then, the modified dynamics of the system can be written as $x(t+1) = Ax(t)+Bu(t)+B_w w(t)+g(x(t), u(t), \epsilon)$. To account for this modification, we follow Remark 3: we introduce the appended disturbance $\tilde{w} := B_w w + g(x, u, \epsilon)$, such that $x(t+1) = Ax(t) + Bu(t) + \tilde{w}(t)$ is the modified system. Since $\mathcal{X}$, $\mathbb{U}(\epsilon)$ and $\mathcal{W}$ are compact, $\tilde{w}$ lies in a parametrized disturbance set, i.e., $\tilde{w} \in B_w \mathbb{W}(\epsilon)$. We use the scaling parametrization $\mathbb{W}(\epsilon) = (1 + \kappa_{\mathrm{d}}(\epsilon))\mathcal{W}$, where $\kappa_{\mathrm{d}}(\epsilon) := 250(A_1^{\mathrm{c}} + A_2^{\mathrm{c}})$. In order to formulate $\mathbf{P}^{i,N}$ with the parametrized disturbance set $\mathbb{W}(\epsilon)$, we first note that the RPI set scales with $\mathbb{W}(\epsilon)$ as $(1 + \kappa_{\mathrm{d}}(\epsilon))\mathcal{E}$. Using positive homogeneity of support functions, we derive the formulation of $\mathbf{P}^{i,N}$ by replacing $h^{\Delta}, \bar{h}^x, \bar{\epsilon}$ and $q^{[i]}(\epsilon)$ in (22) by the parametrized versions $(1 + \kappa_{\mathrm{d}}(\epsilon))h^{\Delta}, (1 + \kappa_{\mathrm{d}}(\epsilon))\bar{h}^x, (1 + \kappa_{\mathrm{d}}(\epsilon))\bar{\epsilon}$ and $\bar{q}^{[i]}(\epsilon, \kappa_{\mathrm{d}}(\epsilon))$ respectively. Hence, $\mathbf{P}^{i,N}$ is an optimization problem with linear constraints and a piecewise affine objective function. We formulate the resulting problem as a Mixed-Integer Linear Program (MILP) using the method presented in [29]. We consider the initial-condition set Ω to be the vertices of $\mathcal{X}$, with the modification $x_1^{\mathrm{v}}, x_3^{\mathrm{v}} = 0$. Then, we formulate and solve the projection problem in (21) to obtain an initial-condition set $\tilde{\Omega}$ for which $\mathbf{P}^{i,N}$ is feasible for all $N \geq 2$ as follows. Defining $\hat{\kappa}_{\mathrm{d}} := 250(\hat{A}_1^{\mathrm{c}} + \hat{A}_2^{\mathrm{c}})$, the largest possible parametrized RPI set is $(1 + \hat{\kappa}_{\mathrm{d}})\mathcal{E}$. Using this set, we compute the MPI set $\hat{\mathbb{O}}_{\infty}(\hat{\epsilon})$ and the 2-step stabilizable set $\hat{\mathbb{K}}_2(\hat{\epsilon}, \hat{\mathbb{O}}_{\infty}(\hat{\epsilon}))$ where $\hat{\epsilon} := [\hat{A}_1^{\mathrm{c}}\ \hat{A}_2^{\mathrm{c}}]^{\top}$, based on which we formulate problem in (21) with the constraint $\tilde{\Omega} \subseteq \hat{\mathbb{K}}_2(\hat{\epsilon}, \hat{\mathbb{O}}_{\infty}(\hat{\epsilon}))$. We choose $\tilde{\Omega}$ to be a set of vertices with the same cardinality as Ω, and the distance function to be the sum of 1-norm of the difference between the vertices, such that resulting problem is an LP. Using $\tilde{\Omega}$ and $\mathbf{P}^{i,N}$, we then compute optimal ϵ for different horizon lengths by applying Algorithm 1. The results are shown in Figure 3. We note that the upper bound to the termination index of Algorithm 1 is now computed using the largest possible parametrized disturbance set $(1 + \hat{\kappa}_{\mathrm{d}})\mathcal{W}$. For $\delta = 10^{-4}$, this bound equals $k_{\delta} = 1200$. In Figure 3 (left plot), we see that the results in Proposition 5 continue to hold, i.e., $\boldsymbol{f}(\epsilon^N)$ is non-increasing

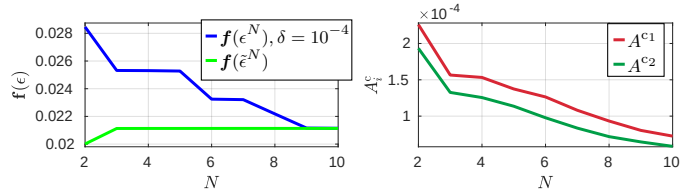

Fig. 3: Numerical results for Example 2 with constraints $\hat{x}^{\mathrm{P}} = 0.8$m, $\hat{x}_1^{\mathrm{v}}, \hat{x}_3^{\mathrm{v}} = 0.15$m/s, $\hat{x}_2^{\mathrm{v}} = 1$m/s, $g^{\mathrm{P}} = 10^{-3}$m. (Left) Optimal input constraint set size $\boldsymbol{f}(\epsilon^N)$ computed by Algorithm 1, along with the lower bound $\boldsymbol{f}(\tilde{\epsilon}^N)$. (Right) Corresponding actuator areas in m^2. These values compose the parameter vector ϵ^N.

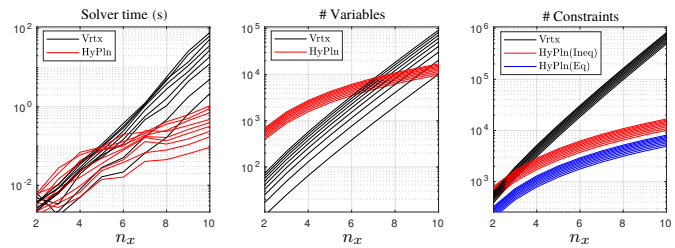

Fig. 4: Numerical results for Example 3, demonstrating the solution time, number of variables, and number of constraints in the formulation of $\mathbf{P}^{i,N}$ when Ω is given in vertex and hyperplane notations. The different lines correspond to horizon lengths $N = 2, \cdots, 10$.

and convergent in N. This is because the stabilizable and admissible sets for a given input and disturbance set are nested sequences in N. This plot can be used as a tool to decide the minimum horizon length required given an upper bound on the budget allotted for pump selection. For example, if the maximum budget spent on pumps must not exceed 0.025 units, then the RMPC controller must have $N \geq 6$. In the right plot, the corresponding optimal actuator areas that can provide the required maximum flowrates $\hat{q}_i = A_i^{\mathrm{c}}\hat{x}_i^{\mathrm{v}}$ are shown, which correspond to the following optimal compressor types:

Horizon length N	2	3	4	5	6	7	8	9	10
Compressor 1 Type	5	4	4	4	3	3	2	2	2
Compressor 2 Type	5	3	3	3	2	2	2	1	1

C. Scalability of $\mathbf{P}^{i,N}$

The goal of the following example is to demonstrate scalability of problem $\mathbf{P}^{i,N}$. We consider the disturbance free system $x(t+1) = Ax(t) + u(t)$, where matrix A has diagonal components $A_{mm} = 1$, and off-diagonal components $A_{mn} = 0.01, m \neq n$. The system is subject to constraints $\mathcal{X} = \mathcal{B}_{\infty}^{n_x}$, and the initial-condition set is $\Omega = 0.2\mathcal{B}_{\infty}^{n_x}$. We parametrize the input constraint set as $\mathbb{U}(\epsilon) = \{u : -\epsilon\mathbf{1} \leq u \leq \epsilon\mathbf{1}\}$, where ϵ is a scalar, and choose $\boldsymbol{f}(\epsilon) = \epsilon$. We equip the system with a feedback controller K which is the solution of the Riccati equation for matrices $Q = \mathbf{I}$ and $R = 0.1\mathbf{I}$. Then, we use both vertex and hyperplane notations of Ω to formulate $\mathbf{P}^{i,N}$ with $i = 10$, $N \in [2, 10]$ and state-space dimension $n_x \in [2, 10]$. The resulting problems are LPs in both cases. The computational time spent by the solver and the number of variables and constraints, are shown in Figure 4. We observe that the dimension n_x significantly affects solver performance in the vertex notation, since the number of vertices in Ω increases exponentially with n_x. This issue is avoided if $\mathbf{P}^{i,N}$ is formulated using Ω in a hyperplane notation (Appendix VII-A). However, since the conditions used to encode the inclusion constraint $\Omega \subseteq \mathbb{K}_N(\epsilon, \mathbb{O}_i(\epsilon))$ are only sufficient, this might lead to conservative solutions $\boldsymbol{f}(\epsilon^{i,N})$. We report that in this example, there was no increase in conservativeness.

VI. Conclusions

We have tackled the problem of computing the smallest input constraint set required to robustly regulate a given set of initial states of an uncertain constrained linear system to the origin. To that end, we used set-invariance properties to develop and analyze an algorithm that computes an input constraint set satisfying recursive feasibility properties of the RMPC scheme [1]. We have demonstrated through numerical examples that the algorithm is capable of handling a variety of size functions, and can accommodate practical considerations while performing actuator selection. Future work will focus on a) including the computation of the feedback gain K and invariant set $\mathcal{E}$ along with the input constraint set $\mathcal{U}$; b) developing the methods for a reference tracking RMPC formulation; c) enhancing the formulation to explicitly account for modification in the dynamics while performing actuator selection.

References

[1] D. Mayne, M. Seron, and S. Raković, "Robust model predictive control of constrained linear systems with bounded disturbances," *Automatica*, vol. 41, no. 2, pp. 219 – 224, 2005.

[2] L. Chisci, J. Rossiter, and G. Zappa, "Systems with persistent disturbances: predictive control with restricted constraints," *Automatica*, vol. 37, no. 7, pp. 1019 – 1028, 2001.

[3] R. Ghaemi, J. Sun, and I. V. Kolmanovsky, "Robust control of constrained linear systems with bounded disturbances," *IEEE Transactions on Automatic Control*, vol. 57, no. 10, pp. 2683–2688, 2012.

[4] L. Ljung, *System Identification (2nd Ed.): Theory for the User*. USA: Prentice Hall PTR, 1999.

[5] F. D. Palma and L. Magni, "A multi-model structure for model predictive control," *Annual Reviews in Control*, vol. 28, no. 1, pp. 47 – 52, 2004.

[6] S. Mulagaleti, M. Zanon, and A. Bemporad, "Dynamic output disturbance models for robust model predictive control," *Proceedings of the 21st IFAC World Congress*, 2020. to be published.

[7] M. Zhu, A. Bemporad, and D. Piga, "Preference-based mpc calibration," 2020.

[8] S. Di Cairano and A. Bemporad, "Model predictive controller matching: Can mpc enjoy small signal properties of my favorite linear controller?," in *2009 European Control Conference (ECC)*, pp. 2217–2222, 2009.

[9] M. Zanon and A. Bemporad, "Constrained control and observer design by inverse optimality," 2020.

[10] H. Janocha, *Actuators: Basics and Applications*. 01 2004.

[11] E. C. Kerrigan and J. M. Maciejowski, "Invariant sets for constrained nonlinear discrete-time systems with application to feasibility in model predictive control," in *Proceedings of the 39th IEEE Conference on Decision and Control*, vol. 5, pp. 4951–4956 vol.5, 2000.

[12] M. Rungger and P. Tabuada, "Computing robust controlled invariant sets of linear systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 7, pp. 3665–3670, 2017.

[13] F. Scibilia, S. Olaru, and M. Hovd, "On feasible sets for mpc and their approximations," *Automatica*, vol. 47, no. 1, pp. 133 – 139, 2011.

[14] V. Tzoumas, A. Jadbabaie, and G. J. Pappas, "Minimal Reachability Problems," *arXiv e-prints*, p. arXiv:1503.07021, Mar. 2015.

[15] V. Tzoumas, M. A. Rahimian, G. J. Pappas, and A. Jadbabaie, "Minimal actuator placement with optimal control constraints," in *2015 American Control Conference (ACC)*, pp. 2081–2086, 2015.

[16] V. L. Silva, L. F. O. Chamon, and A. Ribeiro, "Model predictive selection: A receding horizon scheme for actuator selection," in *2019 American Control Conference (ACC)*, pp. 347–353, 2019.

[17] D. Rotondo and G. Rizzello, "On the optimization of actuator saturation limits for lti systems: an lmi-based invariant ellipsoid approach," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 5567–5572, 2020. 21th IFAC World Congress.

[18] S. K. Mulagaleti, A. Bemporad, and M. Zanon, "Computation of input disturbance sets for constrained output reachability." submitted for publication.

[19] D. Mayne, S. Raković, R. Findeisen, and F. Allgöwer, "Robust output feedback model predictive control of constrained linear systems," *Automatica*, vol. 42, no. 7, pp. 1217 – 1222, 2006.

[20] E. G. Gilbert and K. T. Tan, "Linear systems with state and control constraints: the theory and application of maximal output admissible sets," *IEEE Transactions on Automatic Control*, vol. 36, pp. 1008–1020, Sep. 1991.

[21] I. Kolmanovsky and E. G. Gilbert, "Theory and computation of disturbance invariant sets for discrete-time linear systems," *Mathematical Problems in Engineering*, vol. 4, no. 4, pp. 317–367, 1998.

[22] S. V. Raković, E. C. Kerrigan, K. I. Kouramas, and D. Q. Mayne, "Invariant approximations of the minimal robust positively invariant set," *IEEE Transactions on Automatic Control*, vol. 50, pp. 406–410, March 2005.

[23] P. Trodden, "A one-step approach to computing a polytopic robust positively invariant set," *IEEE Transactions on Automatic Control*, vol. 61, pp. 4100–4105, Dec 2016.

[24] S. Sadraddini and R. Tedrake, "Linear Encodings for Polytope Containment Problems," *arXiv e-prints*, p. arXiv:1903.05214, Mar 2019.

[25] I. I. Cplex, "V12. 1: User's manual for cplex," *International Business Machines Corporation*, vol. 46, no. 53, p. 157, 2009.

[26] M. Zanon and S. Gros, "On the similarity between two popular tube mpc formulations," in *2021 European Control Conference (Accepted)*, 2021.

[27] A. Ilchmann, O. Sawodny, and S. Trenn, "Pneumatic cylinders: Modelling and feedback force-control," *International Journal of Control - INT J CONTR*, vol. 79, 04 2005.

[28] Xiangrong Shen and M. Goldfarb, "Independent stiffness and force control of pneumatic actuators for contact stability during robot manipulation," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pp. 2697–2702, 2005.

[29] G. Ferrari-Trecate, P. Letizia, and M. Spedicato, "Optimization with piecewise affine cost functions," 2001.

VII. Appendix

A. Formulation of $\mathbf{P}^{\mathrm{i,N}}$ with Polytopic Ω

If the hyperplane notation $\Omega = \{x : R^{\Omega}x \leq r^{\Omega}\}$ is given, then $\mathbf{P}^{i,N}$ is formulated using sufficiency conditions from [24] as

$$\begin{aligned}
\epsilon^{i,N} = &\arg \min_{\epsilon,\Gamma^{\mathrm{s}},\beta^{\mathrm{s}},\Lambda^{\mathrm{s}}} \quad \boldsymbol{f}(\epsilon) \\
&\text{s.t.} \quad \Lambda^{\mathrm{s}} \geq \mathbf{0},\ \mathbf{I} = [\mathbf{I}\ \ \mathbf{0}\ \ \mathbf{I}]\Gamma^{\mathrm{s}}, \\
&\qquad \mathbf{0} = [\mathbf{I}\ \ \mathbf{0}\ \ \mathbf{I}]\beta^{\mathrm{s}}, \\
&\qquad \Lambda^{\mathrm{s}} R^{\Omega} = \Sigma^{[N,i]}\Gamma^{\mathrm{s}}, \\
&\qquad \Lambda^{\mathrm{s}} r^{\Omega} \leq \sigma^{[N,i]}(\epsilon) + \Sigma^{[N,i]}\beta^{\mathrm{s}}, \\
&\qquad \epsilon - \bar{\epsilon} \geq \delta\lambda,
\end{aligned} \tag{25}$$

We define the matrices that were used in the formulation of $\mathbf{P}^{i,N}$.

$$\eta^{[N]} := \begin{bmatrix} A^{N-1}B & A^{N-2}B & \dots & B \end{bmatrix},$$

$$g^{[N]} := \begin{bmatrix} h^x - \bar{h}^x \\ \vdots \\ h^x - \bar{h}^x \end{bmatrix}, \quad \sigma^{[N,i]}(\epsilon) := \begin{bmatrix} g^{[N]} \\ q^{[i]}(\epsilon) \\ \epsilon^{[N]} \\ h^{\Delta} \end{bmatrix}, \quad \epsilon^{[N]} := \begin{bmatrix} \epsilon - \bar{\epsilon} \\ \vdots \\ \epsilon - \bar{\epsilon} \end{bmatrix},$$

$$G^{\mathrm{x},[N]} := \begin{bmatrix} H^x \\ H^x A \\ \vdots \\ H^x A^{N-1} \end{bmatrix}, \quad G^{\mathrm{u},[N]} := \begin{bmatrix} \mathbf{0} & & & \\ H^x B & & & \\ \vdots & & \ddots & \\ H^x A^{N-2} B & \dots & H^x B & \mathbf{0} \end{bmatrix},$$

$$S^{[i]} := \begin{bmatrix} \begin{bmatrix} H^x \\ -F^u K \end{bmatrix} A_K^0 \\ \vdots \\ \begin{bmatrix} H^x \\ -F^u K \end{bmatrix} A_K^i \end{bmatrix}, \quad \Sigma^{[N,i]} := \begin{bmatrix} G^{\mathrm{x},[N]} & G^{\mathrm{u},[N]} & \mathbf{0} \\ S^{[i]} A^N & S^{[i]}\eta^{[N]} & \mathbf{0} \\ \mathbf{0} & F^{\mathrm{u},[N]} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & H^{\Delta} \end{bmatrix},$$

$$F^{\mathrm{u},[N]} := \begin{bmatrix} F^u & & \\ & \ddots & \\ & & F^u \end{bmatrix}, \quad q^{[i]}(\epsilon) := \begin{bmatrix} \begin{bmatrix} h^x - \bar{h}^x \\ \epsilon - \bar{\epsilon} \end{bmatrix}^\top \dots \begin{bmatrix} h^x - \bar{h}^x \\ \epsilon - \bar{\epsilon} \end{bmatrix}^\top \end{bmatrix},$$

$$\bar{q}^{[i]}(\epsilon, \kappa_{\mathrm{d}}) := \begin{bmatrix} \begin{bmatrix} h^x - (1+\kappa_{\mathrm{d}})\bar{h}^x \\ \epsilon - (1+\kappa_{\mathrm{d}})\bar{\epsilon} \end{bmatrix}^\top \dots \begin{bmatrix} h^x - (1+\kappa_{\mathrm{d}})\bar{h}^x \\ \epsilon - (1+\kappa_{\mathrm{d}})\bar{\epsilon} \end{bmatrix}^\top \end{bmatrix}.$$