

# Feasibility Governor for Linear Model Predictive Control

Terrence Skibik, Dominic Liao-McPherson, Torbjørn Cunis, Ilya Kolmanovsky, and Marco M. Nicotra

**Abstract**—This paper introduces the Feasibility Governor (FG): an add-on unit that enlarges the region of attraction of Model Predictive Control by manipulating the reference to ensure that the underlying optimal control problem remains feasible. The FG is developed for linear systems subject to polyhedral state and input constraints. Offline computations using polyhedral projection algorithms are used to construct the feasibility set. Online implementation relies on the solution of a convex quadratic program that guarantees recursive feasibility. The closed-loop system is shown to satisfy constraints, achieve asymptotic stability, and exhibit zero-offset tracking.

## I. INTRODUCTION

Model Predictive Control [1]–[3] (MPC) is a feedback policy that computes the solution of a receding horizon Optimal Control Problem (OCP) at every sampling instant. A common approach for guaranteeing the stability of MPC is to impose suitable conditions on the final step of the OCP [4], [5]. The Region of Attraction (ROA) of the resulting closed-loop system is then given by all the states that can reach the terminal constraint set within the prediction horizon.

Since the terminal set is centered on the desired reference, sudden reference changes can cause the OCP to become infeasible if the system is unable to reach the new terminal set within the prediction horizon. Although this issue could be avoided by increasing the prediction horizon, doing so can significantly increase the computational complexity of the controller.

A different option for increasing the ROA is to treat aspects of the terminal set as optimization variables and use the additional degrees of freedom to enlarge the feasible set. This approach has been applied to regulation [6], and reference tracking [7], [8] of linear systems, and also economic operation of nonlinear systems [9]. Alternatively, [10] enlarges the ROA by computing a contractive sequence of terminal sets offline and incorporating them into the OCP. The drawback of all these methods is that they rely on a non-standard OCP, making them difficult to combine with other MPC schemes.

In this paper, we introduce the Feasibility Governor (FG), an add-on unit that filters the reference signal to ensure that the terminal set remains reachable within the prediction

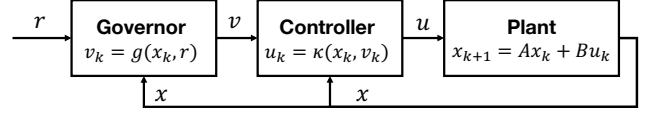


Fig. 1. A block diagram of the control architecture. Given a reference  $r$ , the Feasibility Governor manipulates the auxiliary reference  $v$  to ensure that the primary MPC controller is able to produce a valid control input  $u$ .

horizon. The FG takes advantage of offline polyhedral set manipulation tools [11], [12] to reduce online complexity while minimizing conservatism. Doing so enables the FG to expand the ROA to the set of initial conditions that can reach the terminal set of *any* steady state admissible reference, as opposed to just the target reference. This is achieved without any modifications to the existing MPC controller. Moreover, we prove that the FG ensures the constraints are never violated, and exhibits finite time convergence to the desired reference. The proposed control architecture is illustrated in Figure 1.

The idea of manipulating the reference to avoid infeasibility in MPC can be found in prior literature. The recovery mode featured in [13] simultaneously computes a modified reference and control input to enforce feasibility at the expense of performance. An FG-like algorithm that combines a governor and explicit MPC controller into a single unit is proposed in [14], but suffers from the complexity limitations of explicit MPC [15]. A suboptimal continuous-time analog of the FG is proposed in [16]. Finally, a spatial governor specifically designed for precision machining applications is proposed in [17]. This paper provides both a detailed look into the theoretical properties of the proposed FG, and a method for computing the feasible sets.

*Notation:* For vectors  $a$  and  $b$ ,  $(a, b) = [a^T \ b^T]^T$ . The identity matrix is  $I_N \in \mathbb{R}^{N \times N}$ . Given  $M \in \mathbb{R}^{m \times n}$ ,  $\text{Ker } M = \{x \mid Mx = 0\}$ . Given  $x \in \mathbb{R}^n$  and a positive definite matrix  $P \in \mathbb{R}^{n \times n}$ , the weighted norm is  $\|x\|_P = \sqrt{x^T P x}$ . Given  $x \in \mathbb{R}^n$ ,  $y \in \mathbb{R}^m$  and a set  $\Gamma \subseteq \mathbb{R}^{n+m}$ , the projection of  $\Gamma$  onto the domain of  $x$  is  $\Pi_x \Gamma$ , where  $\Pi_x = [I_n \ 0_{n \times m}]$ , and the slice (or cross-section) operation is  $S_y(\Gamma, x) = \{y \mid (x, y) \in \Gamma\}$ .

## II. PROBLEM SETTING

Consider the linear time invariant (LTI) system

$$x_{k+1} = Ax_k + Bu_k \quad (1a)$$

$$y_k = Cx_k + Du_k \quad (1b)$$

$$z_k = Ex_k + Fu_k, \quad (1c)$$

M. Nicotra and T. Skibik are with the University of Colorado, Boulder, Email: {marco.nicotra, terrence.skibik}@colorado.edu.

D. Liao-McPherson, T. Cunis and I. Kolmanovsky are with the University of Michigan, Ann Arbor. Email: {dliaomcp, tcunis, ilya}@umich.edu.

This research is supported by the National Science Foundation Award Numbers CMMI 1904441, CMMI 1904394, and the Toyota Research Institute (TRI). TRI provided funds to assist the authors with their research but this article solely reflects the opinions and conclusions of its authors and not TRI or any other Toyota entity.

where  $k \in \mathbb{N}$  is the discrete-time index and  $x_k \in \mathbb{R}^{n_x}$ ,  $u_k \in \mathbb{R}^{n_u}$ ,  $y_k \in \mathbb{R}^{n_y}$ , and  $z_k \in \mathbb{R}^{n_z}$  are the states, control inputs, constrained outputs, and tracking outputs, respectively. System (1) is subject to pointwise-in-time constraints

$$y_k \in \mathcal{Y}, \quad \forall k \in \mathbb{N}, \quad (2)$$

where  $\mathcal{Y} \subseteq \mathbb{R}^{n_y}$  is the constraint set.

**Assumption 1.** *The pair  $(A, B)$  is stabilizable.*

As detailed in [7], Assumption 1 implies that

$$Z = \begin{bmatrix} A - I_{n_x} & B & 0 \\ E & F & -I_{n_z} \end{bmatrix} \quad (3)$$

satisfies  $\text{Ker}(Z) \neq \{0\}$  and, as a result, system (1) admits a family of equilibrium points satisfying  $Z\zeta = 0$  with  $\zeta = (x, u, z) \neq 0$ . Moreover, it is possible to introduce an auxiliary reference vector  $v \in \mathbb{R}^{n_v}$  to parameterize the equilibrium manifold as  $\tilde{\zeta}_v = (\bar{x}_v, \bar{u}_v, \bar{z}_v) = Gv$  where

$$G^T := [G_x^T \ G_u^T \ G_z^T] \quad (4)$$

is a basis for  $\text{Ker}(Z)$ . The following assumption ensures that there is a one-to-one correspondence between the reference  $v$  and the tracking output  $z$ .

**Assumption 2.** *The matrix  $G_z$  is invertible.*

Under Assumption 2, it is possible to impose  $G_z = I_{n_z}$  using the change of basis  $G \leftarrow GG_z^{-1}$ .

**Assumption 3.** *The set  $\mathcal{Y}$  is a compact polyhedron with representation  $\mathcal{Y} = \{y \mid Yy \leq h\}$  and satisfies  $0 \in \text{Int } \mathcal{Y}$ .*

Given the design parameter  $\epsilon \in (0, 1)$  and the corresponding set of strictly steady-state admissible references

$$\mathcal{R}_\epsilon = \{v \mid (CG_x + DG_u)v \in (1 - \epsilon)\mathcal{Y}\}, \quad (5)$$

we now state the control problem addressed by this paper.

**Control Objectives:** Given the LTI system (1) subject to constraints (2), let  $r \in \mathbb{R}^{n_z}$  be a target reference. The goal of this paper is to design a full state feedback law that achieves

- *Safety:*  $y_k \in \mathcal{Y}, \quad \forall k \geq 0$ ;
- *Convergence:*  $\lim_{k \rightarrow \infty} z_k = r^*$ , where

$$r^* = \arg \min_{v \in \mathcal{R}_\epsilon} \|v - r\|.$$

**Remark 1.** *When the tracking problem is well posed, i.e.,  $r \in \mathcal{R}_\epsilon$ , we recover  $\lim_{k \rightarrow \infty} z_k = r$ .*

### III. CONTROL STRATEGY

Due the constraints, we approach the control objectives using a typical MPC formulation where the feedback policy is defined using the solution to the following OCP

$$\min_{\mu} \|\xi_N - \bar{x}_v\|_P^2 + \sum_{i=0}^{N-1} \|\xi_i - \bar{x}_v\|_Q^2 + \|\mu_i - \bar{u}_v\|_R^2 \quad (6a)$$

$$\text{s.t. } \xi_0 = x, \quad (6b)$$

$$\xi_{i+1} = A\xi_i + B\mu_i, \quad i \in \mathbb{N}_{[0, N-1]}, \quad (6c)$$

$$C\xi_i + D\mu_i \in \mathcal{Y}, \quad i \in \mathbb{N}_{[0, N-1]}, \quad (6d)$$

$$(\xi_N, v) \in \mathcal{T}, \quad (6e)$$

where  $N \in \mathbb{N}_{>0}$  is the prediction horizon,  $\mu = (\mu_0, \dots, \mu_{N-1})$ ,  $P$ ,  $Q$ , and  $R$  are weighting matrices, and  $\mathcal{T} \subseteq \mathbb{R}^{n_x} \times \mathbb{R}^{n_v}$  is the terminal set, which is assumed to be polyhedral, i.e.,

$$\mathcal{T} = \{(x, v) \mid T_x x + T_v v \leq c\}. \quad (7)$$

The following assumptions ensure that (6) is well-posed and can be used to construct a stabilizing feedback law.

**Assumption 4.** *The stage cost matrices satisfy  $Q \succeq 0$ , with  $(A, Q)$  detectable, and  $R \succ 0$ .*

Given Assumption 4, let  $P$  be the solution to the discrete algebraic Riccati equation

$$P = Q + A^T P A - (A^T P B)(R + B^T P B)^{-1}(B^T P A), \quad (8)$$

let  $K$  be the associated LQR gain

$$K = (R + B^T P B)^{-1}(B^T P A), \quad (9)$$

and let  $\mathcal{T} = \tilde{O}_\infty^\epsilon$  be the terminal set, with  $\tilde{O}_\infty^\epsilon$  defined in [18]. By construction, the terminal set  $\mathcal{T}$  is invariant and constraint admissible, i.e.,  $(x, v) \in \mathcal{T}$  implies

$$(A - BK)x + B(\bar{u}_v + K\bar{x}_v) \in \mathcal{X}(v), \quad (10a)$$

$$(C - DK)x + D(\bar{u}_v + K\bar{x}_v) \in \mathcal{Y}, \quad (10b)$$

where  $\mathcal{X}(v) = S_x(\mathcal{T}, v)$ . We now have all the elements typically used to define an asymptotically stable MPC feedback policy [4]. However, the control action can be computed only if (6) admits a solution. The set of all parameters for which the OCP admits a solution, i.e., the feasible set, is

$$\Gamma_N = \{(x, v) \mid \exists \mu : (6b) - (6e)\} \subseteq \mathbb{R}^{n_x} \times \mathbb{R}^{n_v}, \quad (11)$$

which is the  $N$ -step backwards reachable set of  $\mathcal{T}$ . Since  $\mathcal{T}$  is polyhedral, then  $\Gamma_N$  is also polyhedral and can be computed offline, as detailed in Section IV-C. Assuming  $(x, v) \in \Gamma_N$ , it is possible to compute the MPC feedback policy

$$\kappa(x, v) = \mu_0^*(x, v) \quad (12)$$

where  $\mu^*(x, v) = [\mu_0^{*T}, \mu_1^{*T}, \dots, \mu_{N-1}^{*T}]^T$  is the minimizer of (6). The following theorem summarizes the properties of the closed-loop system for a constant auxiliary reference.

**Theorem 1.** *Let Assumptions 1–4 hold and let  $\phi(\ell, x, v)$  denote the solution of the closed-loop dynamics*

$$x_{k+1} = f(x_k, v) := Ax_k + B\kappa(x_k, v). \quad (13)$$

*starting from the initial condition  $x_0 = x$  at timestep  $\ell \geq 0$ . Then for all  $(x, v) \in \Gamma_N$ .*

- $(\phi(\ell, x, v), v) \in \Gamma_N, \quad \forall \ell \geq 0$ ;
- $y_\ell \in \mathcal{Y}, \quad \forall \ell \geq 0$ ;
- $\lim_{\ell \rightarrow \infty} \phi(\ell, x, v) = \bar{x}_v$ .

*If, in addition,  $v \in \mathcal{R}_\epsilon$  then  $\bar{x}_v$  is asymptotically stable.*

*Proof.* Since the auxiliary reference  $v$  is constant for  $\ell \geq 0$ , the statement follows from [2, Theorem 4.4.2].  $\square$

Theorem 1 achieves the control objectives given  $v = r$  and  $x_0$  satisfying  $(x_0, r) \in \Gamma_N$  with an ROA of  $S_x(\Gamma_N, r)$ .

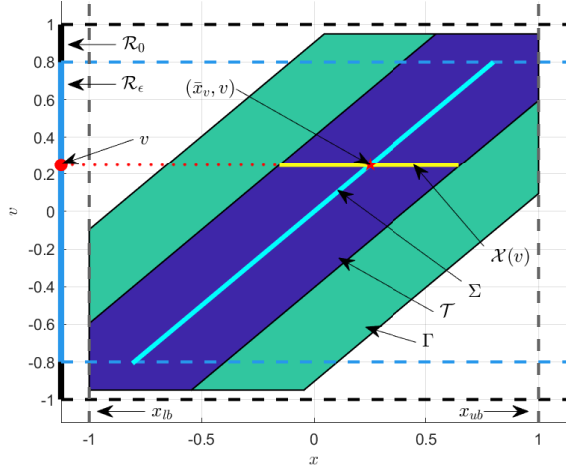


Fig. 2. Illustration of the sets used in the paper for the discrete-time integrator  $x_{k+1} = x_k + u_k$  subject to  $|x_k| \leq 1$  and  $|u_k| \leq 0.25$ , given  $\epsilon = 0.2$ ,  $\mathcal{T} = \tilde{\mathcal{O}}_\infty^{0.05}$  and  $N = 2$ .

Its main limitation, however, lies in the fact that the OCP (6) is infeasible if  $x_0$  cannot be steered to  $\mathcal{X}(r)$  within  $N$  steps. Although increasing the prediction horizon may seem like a suitable workaround, this solution may be inapplicable under real-time restrictions since the computational time required to solve (6) scales unfavorably with  $N$ . In view of extending the ROA, we define the set of strictly steady-state admissible equilibria

$$\Sigma = \{(x, v) \mid x = G_x v, v \in \mathcal{R}_\epsilon\}, \quad (14)$$

and note the following.

**Lemma 1.**  $\Sigma \subset \text{Int } \Gamma_N$ .

*Proof.* Since  $(\bar{x}_v, v) \in \text{Int } \tilde{\mathcal{O}}_\infty^\epsilon$  [18, Theorem 2.1], the result follows from  $\Sigma \subset \text{Int } \mathcal{T} \subseteq \text{Int } \Gamma_N$   $\square$

Figure 2 depicts all the sets defined in this section. In the next section, we describe an *add-on* unit that expands the closed-loop domain of attraction without extending the prediction horizon or modifying the MPC formulation.

#### IV. THE FEASIBILITY GOVERNOR

The MPC feedback policy (12) is stabilizing only if the terminal set associated with the target equilibrium is  $N$ -step reachable from the current state. Intuitively, this limitation can be overcome by selecting a sequence of intermediate targets that are pair-wise reachable. This paper formalizes this idea by redefining the auxiliary reference  $v$  as a time-varying system  $v_k$  to ensure  $(x_k, v_k) \in \Gamma_N$ ,  $\forall k \in \mathbb{N}$ , and  $v_k = r$  for sufficiently large  $k \in \mathbb{N}$ . The resulting control architecture is displayed in Figure 1.

##### A. Governor Design

The idea behind the FG is straightforward: modify the target reference as little as needed to ensure that the OCP remains feasible. Drawing inspiration from the Command

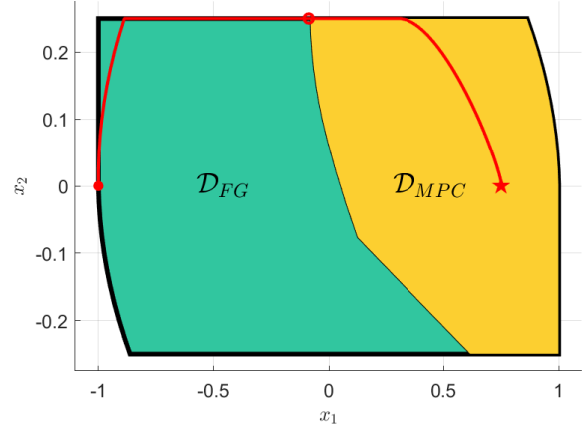


Fig. 3. For the double integrator example in Section VI, the region of attraction of the combined MPC + FG feedback law (green + yellow) is significantly larger than that of the MPC controller alone (yellow), given a prediction horizon of  $N = 10$ . The red trajectory is from the double integrator example in Section VI.

Governor (CG) literature [19], [20], the FG policy can be computed via by solving

$$g(x, r) = \arg \min_{v \in \mathcal{R}_\epsilon} \{\|v - r\|_2^2 \mid (x, v) \in \Gamma_N\}. \quad (15)$$

Given a measurement  $x_k$ , the FG computes a virtual reference  $v_k = g(x_k, r)$  that is passed to the MPC controller to obtain a control action  $u_k = \kappa(x_k, v_k)$ .

##### B. Properties

Given system (1) and the feedback policy (12), the FG is recursively feasible (see Theorem 2), guarantees constraint satisfaction (see Theorem 2), renders the point  $x_r^* = G_x v_r^*$  asymptotically stable (see Theorem 3), and exhibits finite time convergence of  $v_k \rightarrow v_r^*$  (see Theorem 4).

Moreover, the FG expands the ROA of the closed-loop system from  $\mathcal{D}_{MPC} = S_x(\Gamma_N, r^*)$ , i.e., the set of states from which it is possible to reach  $\mathcal{X}(r^*)$  in  $N$ -steps, to

$$\mathcal{D}_{FG} = \bigcup_{v \in \mathcal{R}_\epsilon} S_x(\Gamma_N, v), \quad (16)$$

i.e., the set of states from which it is possible to reach  $\mathcal{X}(v)$  for any  $v \in \mathcal{R}_\epsilon$ . In particular, the addition of the FG guarantees safe transitions between any  $r_1, r_2 \in \mathcal{R}_\epsilon$ . The differences between  $\mathcal{D}_{MPC}$  and  $\mathcal{D}_{FG}$  are illustrated in Figure 3 for the double integrator example in Section VI.

##### C. Implementation

Since  $\Gamma_N$  and  $\mathcal{R}_\epsilon$  are polyhedral, (15) is a strongly convex quadratic program (QP), and can therefore be solved in real-time. Moreover, since the FG problem typically has a small number of variables and many constraints, dual active-set methods [21] are particularly well suited for solving (15) efficiently and reliably due to the limited number of active constraints at any given time.

Implementation of the FG also requires a half-space representation of the feasible set  $\Gamma_N$ . To compute such a

representation, note that the OCP (6) is a QP and can be written in the condensed form [22]

$$\min_{\mu} \quad \frac{1}{2} \mu^T H \mu + \mu^T W \theta \quad (17a)$$

$$\text{s.t.} \quad M \mu + L \theta \leq b, \quad (17b)$$

with parameter  $\theta = (x, v)$ . The feasible set (11) can therefore be expressed as

$$\Gamma_N = \Pi_{\theta} \{(\mu, \theta) \mid M \mu + L \theta \leq b\}. \quad (18)$$

Several toolboxes are available for performing polyhedral calculus (e.g., projections, images, inverse images etc.). In this paper, we compute  $\Gamma_N$  using the `bensolve tools` [23] package. Unfortunately, the complexity of computing  $\Gamma_N$  is dominated by the projection operation. The projection is performed offline but can quickly become intractable even for moderately sized systems as all known projection algorithms suffer from the curse of dimensionality [24]. Thus, the offline computation of  $\Gamma_N$  can quickly become intractable as the size of the state vector, input vector, reference, or prediction horizon grows.

## V. THEORETICAL ANALYSIS

This section analyzes the properties of the closed-loop system under the combined FG and MPC policy. We begin with some definitions. The feasible set of the FG is

$$\Lambda = \Gamma_N \cap (\mathbb{R}^{n_x} \times \mathcal{R}_{\epsilon}). \quad (19)$$

The closed-loop dynamics of (1) under the combined FG and MPC feedback law are

$$x_{k+1} = f(x_k, g(x_k, r)), \quad (20)$$

and the constrained output is

$$y_k = C x_k + D \kappa(x_k, v_k). \quad (21)$$

### A. Safety and Recursive Feasibility

The following theorem provides sufficient conditions under which the Feasibility Governor (FG) is recursively feasible and achieves the *Safety* objective.

**Theorem 2** (Safety & Recursive Feasibility). *Given Assumptions 1–4, consider the closed-loop dynamics (20). Given  $x_0 \in \Pi_x \Lambda$ , then  $(x_k, v_k) \in \Lambda$  and  $y_k \in \mathcal{Y}$ ,  $\forall k \in \mathbb{N}$ .*

*Proof.* The proof is by induction. Given  $x_0 \in \Pi_x \Lambda$ , the FG optimization problem is feasible at time  $k = 0$  and  $(x_0, v_0) \in \Lambda$ . Next, given  $(x_k, v_k) \in \Lambda$ , Theorem 1 implies  $(f(x_k, v_k), v_k) = (x_{k+1}, v_k) \in \Lambda$ . Moreover, since  $(x_{k+1}, v_k) \in \Lambda$  it follows that (15) is feasible at timestep  $k + 1$  and thus  $(x_{k+1}, v_{k+1}) = (x_{k+1}, g(x_{k+1}, r)) \in \Lambda$ . Therefore, by induction,  $(x_k, v_k) \in \Lambda \subset \Gamma_N$ ,  $\forall k \in \mathbb{N}$ . Finally,  $y_k \in \mathcal{Y}$ ,  $\forall k \in \mathbb{N}$  follows from  $(x_k, v_k) \in \Gamma_N$  and Theorem 1.  $\square$

### B. Convergence and Stability

Having established recursive feasibility, we now consider convergence and stability, starting with asymptotic stability. Throughout the section the reference is constant so we suppress all dependencies on  $r$  to simplify the notation.

The proof is via the invariance principle [25, Theorem 2], so we begin by introducing the Lyapunov function candidate

$$V(v) = \|v - r\|_2^2, \quad (22)$$

and the increment function

$$\Delta V(x, v) = V(g(f(x, v))) - V(v). \quad (23)$$

The first step is to characterize the set

$$\Omega = \{(x, v) \in \Lambda \mid \Delta V(x, v) = 0\}. \quad (24)$$

**Lemma 2.** *Under Assumptions 1–4, there exists  $m > 0$  such that*

$$\Delta V(x, v) \leq -m \|g(f(x, v)) - v\|^2 \leq 0, \quad (25)$$

for all  $(x, v) \in \Lambda$  and thus

$$\Omega = \{(x, v) \in \Lambda \mid g(f(x, v)) = v\}.$$

*Proof.* The function  $V$  is strongly convex and continuously differentiable. Thus, there exists  $m > 0$  such that

$$V(v) \geq V(v') + \nabla V(v')^T (v - v') + m \|v - v'\|_2^2$$

for all  $v', v \in \mathbb{R}^{n_v}$ . Letting  $x^+ = f(x, v)$  and  $v' = g(x^+) \in S_v(\Lambda, x^+)$ , we have that

$$\nabla V(v')^T (v - v') \geq 0, \quad \forall v \in S_v(\Lambda, x^+). \quad (26)$$

By using  $v' = g(f(x, v))$ , and recalling that  $(x, v) \in \Lambda$  implies  $(x^+, v) \in \Lambda$  we obtain

$$V(g(f(x, v))) - V(v) \leq -m \|g(f(x, v)) - v\|^2 \leq 0,$$

for all  $(x, v) \in \Lambda$ . Thus  $\Delta V(x, v) \leq 0$  for all  $(x, v) \in \Lambda$  and  $\Omega = \{(x, v) \in \Lambda \mid g(f(x, v)) = v\}$ .  $\square$

The next Lemma shows that, if  $x$  is close enough to  $\bar{x}_v$ , then the FG is able to make progress towards  $r^*$ .

**Lemma 3.** *Given Assumptions 1–4, define*

$$\mathcal{B}_{\delta}(\Sigma) = \{(x, v) \mid v \in \mathcal{R}_{\epsilon}, \|x - G_x v\| \leq \delta\}, \quad (27)$$

where  $\Sigma = \{(x, v) \mid x = G_x v, v \in \mathcal{R}_{\epsilon}\}$  is the equilibrium manifold. Then, there exists  $\delta^* > 0$  such that  $g(x) \neq v$  for all  $(x, v) \in \mathcal{B}_{\delta}(\Sigma)$ ,  $v \neq r^*$  and  $\delta \in [0, \delta^*]$ .

*Proof.* Lemma 1 implies that  $\mathcal{B}_0(\Sigma) = \Sigma \subset \text{Int } \Gamma_N$  hence there exists  $\delta^* > 0$  such that  $\mathcal{B}_{\delta}(\Sigma) \subset \text{Int } \Gamma_N$  for all  $\delta \in [0, \delta^*]$ . Since  $\mathcal{B}_{\delta}(\Sigma) \subset \text{Int } \Gamma_N$ , for any  $(x, v) \in \mathcal{B}_{\delta}(\Sigma)$  there exists  $\alpha = \alpha(\delta) > 0$  such that  $(x, v') \in \Gamma_N$  for all  $v' \in \mathcal{B}_{\alpha}(v)$ , or, equivalently,  $\mathcal{B}_{\alpha}(v) \subseteq S_v(\Gamma_N, x)$ .

The goal is to show that there exists  $v' \in S_v(\Lambda, x)$  such that  $V(v') < V(v)$  whenever  $v \neq r^*$ . First, fix any  $\delta \in [0, \delta^*]$  and  $\alpha = \alpha(\delta)$ , define the set  $\mathcal{C}_{\alpha} = \mathcal{R}_{\epsilon} \cap \mathcal{B}_{\alpha}(v)$  and the ray  $v'(t) = v + t(r^* - v)$  with  $t \geq 0$ , and assume  $v \neq r^*$ . It is evident that  $v'(t) \in \mathcal{C}_{\alpha}$  for  $t \in [0, \min(1, \beta)]$ , with  $\beta = \frac{\alpha}{\|v - r^*\|}$ , because  $v$  and  $r^*$  are in the convex set  $\mathcal{R}_{\epsilon}$

and  $\|v'(t) - v\| \leq \alpha$  for all  $t \leq \beta$ . Further,  $V$  is strongly convex and  $\min_{s \in \mathcal{R}_\epsilon} V(s) = V(r^*) \leq V(v)$  for all  $v \in \mathcal{R}_\epsilon$ , implying that

$$\begin{aligned} V(v'(\beta)) &= V((1 - \beta)v + \beta r^*) \\ &< V(v) + \beta[V(r^*) - V(v)] \\ &< V(v) \end{aligned}$$

for all  $v \in \mathcal{R}_\epsilon \setminus r^*$ . Since  $v'(\beta) \in \mathcal{C}_\alpha$  we have

$$\min_{s \in \mathcal{C}_\alpha} V(s) \leq V(v'(\beta)) < V(v), \quad (28)$$

and, since  $\mathcal{C}_\alpha \subseteq S_v(\Lambda, x) = \mathcal{R}_\epsilon \cap S_v(\Gamma_N, x)$ ,

$$V(g(x)) = \min_{s \in S_v(\Lambda, x)} V(s) \leq \min_{s \in \mathcal{C}_\alpha} V(s). \quad (29)$$

Combining (28) and (29) we conclude

$$V(g(x)) < V(v) \quad \forall (x, v) \in \mathcal{B}_\alpha(\Sigma), \quad v \neq r^* \quad (30)$$

and thus, because  $V$  is strongly convex,  $g(x) \neq v$  for all  $(x, v) \in \mathcal{B}_\delta(\Sigma)$ ,  $v \neq r^*$ , and  $\delta \in [0, \delta^*]$ .  $\square$

Lemma 4 characterizes the largest invariant set in  $\Omega$ , an essential step in the application of the invariance principle.

**Lemma 4.** *Let Assumptions 1–4 hold. Then the point*

$$\mathcal{I}_0 = \{(\bar{x}_r^*, r^*)\}, \quad \bar{x}_r^* = G_x r^* \quad (31)$$

*is the largest invariant set in  $\Omega \subset \Lambda$ .*

*Proof.* Let  $\mathcal{I} \subset \Omega$  be the largest invariant set in  $\Omega$ . Our approach is to erode  $\mathcal{I}$  in several steps.

First, define the function  $\phi : \mathbb{N}_{\geq 0} \times \Gamma_N \rightarrow \mathbb{R}^{n_x}$  such that  $\phi(\ell, x, v)$  denotes the solution of  $x_{k+1} = f(x_k, v)$  starting from  $x_0 = x$  at timestep  $\ell \geq 0$ . Let

$$\tilde{\Omega} = \{(x, v) \in \Lambda \mid g(\phi(\ell, x, v)) = v, \quad \forall \ell \geq 0\} \subseteq \Omega,$$

be the set of all initial conditions for which  $v$  remains constant for all time. Since the set

$$\Omega \setminus \tilde{\Omega} = \{(x, v) \in \Lambda \mid \exists \ell > 0, \quad g(\phi(\ell, x, v)) \neq v\},$$

cannot be invariant,  $\mathcal{I} \subseteq \tilde{\Omega}$ . To see that  $\Omega \setminus \tilde{\Omega}$  isn't invariant, consider any initial condition  $(x_0, v_{-1}) \in \Omega \setminus \tilde{\Omega}$  and consider the point  $(x_{\ell-1}, v_{\ell-1})$  on the resulting trajectory. By the definition of  $\Omega \setminus \tilde{\Omega}$ ,  $g(x_{\ell-1}, v_{\ell-1}) = g(f(x_{\ell-1}, v_{\ell-1}), v_{\ell-1}) \neq v_{\ell-1}$ , which implies  $(x_{\ell-1}, v_{\ell-1}) \notin \Omega$ . Thus, for any  $(x_0, v_{-1}) \in \Omega \setminus \tilde{\Omega}$ , the resulting trajectory exits  $\Omega \setminus \tilde{\Omega}$ .

The inclusion  $\mathcal{I} \subseteq \tilde{\Omega}$  implies that the auxiliary reference must remain constant in  $\mathcal{I}$ , so we focus on the evolution of  $x$  given a constant  $v$ . Recall the set  $\mathcal{B}_\delta(\Sigma)$  defined in (27). Due to Theorem 1, for any  $(x, v) \in \Lambda$  and  $\delta > 0$ , there exists a finite  $t = t(\delta) \geq 0$  such that  $(\phi(t, x, v), v) \in \mathcal{B}_\delta(\Sigma)$ . Further, by Lemma 3, there also exists  $\delta^* > 0$  such that  $(\phi(t, x, v), v) \in \mathcal{B}_\delta(\Sigma)$  implies  $g(\phi(t, x, v)) \neq v$  for any  $\delta \in [0, \delta^*]$  and  $v \neq r^*$ . Thus,  $\tilde{\Omega} \setminus \mathcal{B}_\delta(\Sigma)$  is not invariant and  $\mathcal{I} \subseteq (\tilde{\Omega} \cap \mathcal{B}_\delta(\Sigma))$  for arbitrarily small  $\delta > 0$ .

The inclusion  $\mathcal{I} \subseteq (\tilde{\Omega} \cap \mathcal{B}_\delta(\Sigma))$  for arbitrarily small  $\delta > 0$  implies  $\mathcal{I} \subseteq \mathcal{B}_0(\Sigma) = \Sigma$ . Thanks to Lemma 1,  $\Sigma \subset \text{Int } \Gamma_N$ , which implies

$$g(G_x r^*) = \arg \min_{s \in \mathcal{R}_\epsilon} V(s) = r^*. \quad (32)$$

Thus,  $v = r^* \implies g(G_x v) = v$ . Next, applying Lemma 3 with  $\delta = 0$ , we have that  $v \neq r^* \implies g(G_x v) \neq v$  which is equivalent to  $g(G_x v) = v \implies v = r^*$ . As a result,  $g(G_x v) = v \iff v = r^*$  and, therefore,

$$\mathcal{I} = \{(x, v) \mid x = G_x v, \quad v = r^*\} = (\bar{x}_r^*, r^*), \quad (33)$$

completing the proof.  $\square$

Having assembled all the components, we can now invoke the invariance principle to show asymptotic stability and finite-time convergence.

**Theorem 3** (Asymptotic Stability). *Let Assumptions 1–4 hold. Then,  $(\bar{x}_r^*, r^*)$  is an asymptotically stable equilibrium point of the closed-loop system (20), with domain of attraction  $\mathcal{D} = \Pi_x \Lambda \times \mathbb{R}^{n_v}$ .*

*Proof.* Consider the candidate Lyapunov function  $V : \mathcal{R}_\epsilon \rightarrow \mathbb{R}$  defined in (22) and note that  $V$  is continuous, bounded below, and,  $V(v_{k+1}) \leq V(v_k)$  for all  $(x_k, v_k) \in \Lambda$  by Lemma 2. Moreover,  $x_0 \in \Pi_x \Lambda \implies (x_k, v_k) \in \Lambda$  for all  $k \geq 0$  (Theorem 2), thus (20) is Lyapunov stable. Further, invoking Lemma 4, the largest invariant subset of  $\Omega = \{(x, v) \mid \Delta V(x, v) = 0\}$  is  $(\bar{x}_r^*, r^*)$ . Therefore, by the invariance principle [25, Theorem 2],  $(x_k, v_k) \rightarrow (\bar{x}_r^*, r^*)$  as  $k \rightarrow \infty$  for all  $x_0 \in \Pi_x \Lambda$ .  $\square$

**Theorem 4** (Finite-time Convergence). *Let Assumptions 1–4 hold and consider the closed-loop system (20). Then,  $\forall x_0 \in \Pi_x \Lambda$ , there exists  $t \geq 0$  such that  $v_k = r^*$ ,  $\forall k \geq t$ .*

*Proof.* Due to Lemma 1,  $(\bar{x}_r^*, r^*) \in \Sigma \subset \text{Int } \Gamma_N$ . Thus,  $\bar{x}_r^* \in \text{Int } S_x(\Gamma_N, r^*)$ . In addition, the definition of  $\Lambda = \Gamma_N \cap (\mathbb{R}^{n_x} \times \mathcal{R}_\epsilon)$  implies  $S_x(\Lambda, v) = S_x(\Gamma_N, v)$  for all  $v \in \mathcal{R}_\epsilon$  and therefore  $\bar{x}_r^* \in \text{Int } S_x(\Lambda, r^*)$ . Since  $\bar{x}_r^* \in \text{Int } S_x(\Lambda, r^*)$  and  $x_k \rightarrow \bar{x}_r^*$  as  $k \rightarrow \infty$  (Theorem 3) there exists a finite  $t \geq 0$  such that  $x_t \in S_x(\Lambda, r^*)$ . By strong convexity of  $V$ , it follows from (15) that  $g(x) = r^*$  for all  $x \in S_x(\Lambda, r^*)$ , which implies  $v_k = r^*$  for all  $k \geq t$ .  $\square$

## VI. NUMERICAL EXAMPLES

We consider a double integrator example, which allows us to represent the geometries of the various sets. The system matrices are

$$A = \begin{bmatrix} 1 & 0.1 \\ 0 & 1 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0.1 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, D = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix},$$

$E = \begin{bmatrix} 1 & 0 \end{bmatrix}$ , and  $F = 0$ , and the sampling time is  $t_s = 0.1$ . The default constraint set is

$$\mathcal{Y}_1 = [-1, 1] \times [-0.25, 0.25] \times [-0.25, 0.25],$$

and the MPC parameters are  $Q = I$ ,  $R = 1$ , and  $N = 10$  unless otherwise specified. The reference  $r = 0.75$  and initial condition  $x_0 = [-1, 0]^T$  are chosen such that  $r \in \mathcal{R}_{0.01}$  and  $x_0 \notin S_x(\Gamma_{10}, r)$ . For all the following figures, the terminal set  $\mathcal{T} = \tilde{\mathcal{O}}_\infty^{0.01}$  is computed using the procedure in [18].

Figure 4 illustrates the sets  $\mathcal{T}$  and  $\Gamma_{10}$ . The terminal set  $\mathcal{T}$  is entirely contained in the feasible set, and in both cases  $v$  is

implicitly bounded by the constraints on  $x_1$ . The trajectory of the double integrator is displayed in Figure 5 and illustrates how the MPC + FG navigates  $\Gamma_{10}$ . By the time  $v_k = r$ , the current state  $x_k$  of the system has entered  $\mathcal{D}_{MPC}$  (yellow). From here, the FG holds the auxiliary reference constant and the MPC controller ensures that  $x_k \rightarrow \bar{x}_r$  as  $k \rightarrow \infty$ .

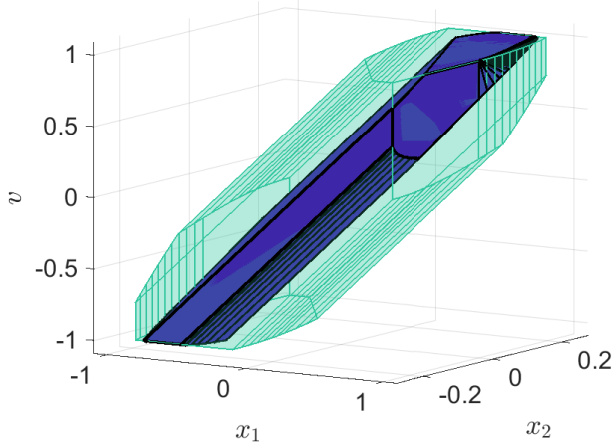


Fig. 4. Terminal set  $\mathcal{T}$  (blue) enclosed in feasible set  $\Gamma_{10}$  (teal) for the double integrator with constraints  $\mathcal{Y}_1$ .

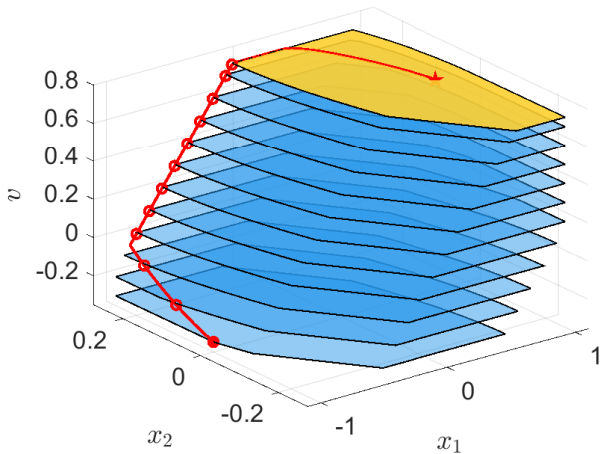


Fig. 5. A closed-loop trajectory of the double integrator over the slices  $S_x(\Gamma_{10}, v)$  for different values of  $v$  with constraints  $\mathcal{Y}_1$ . Circle markers show when the trajectory enters each slice and the star is the point  $(x_r^*, v_r^*)$ .

The feasible sets form an increasing sequence of sets in  $N$ , i.e.,  $\Gamma_N \subseteq \Gamma_{N+1}$  for all  $N \geq 0$ . This is illustrated in Figure 6 which uses a modified constraint set

$$\mathcal{Y}_2 = [-1, 1] \times [-1, 1] \times [-0.05, 0.05]$$

for clarity. The set  $\Gamma_N$  appears to be approaching some  $\Gamma_\infty \supseteq \Gamma_N$ , we hypothesize that this occurs whenever  $\mathcal{Y}$  is compact.

Figure 7 compares the MPC + FG feedback law with  $N = 10$  to an un-governed MPC controller with  $N = N^* = 236$

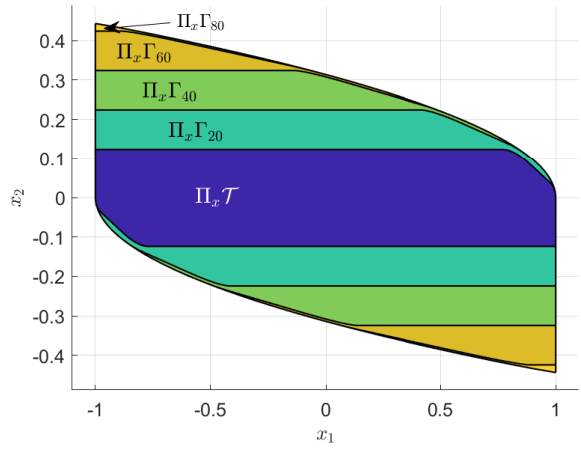


Fig. 6. Increasing the control horizon  $N$  expands the size of the feasible set while the terminal set stays constant. Here  $\mathcal{T} = \Gamma_0 = \bar{O}_{\infty}^{0,01}$  with constraints  $\mathcal{Y}_2$

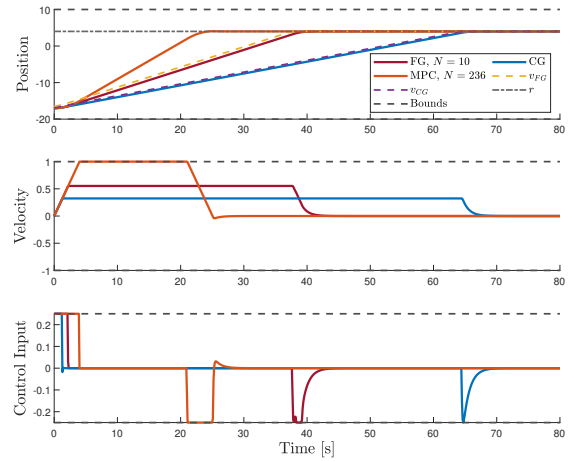


Fig. 7. Closed-loop double integrator dynamics for various control laws with constraints  $\mathcal{Y}_3$ . The FG out performs the CG, and although the MPC has the best performance, its control horizon is too large for real-time applications.

where

$$N^* = N^*(x_0, r, \mathcal{T}) = \inf_i \{i \mid (x_0, r) \in \Gamma_i\} \quad (34)$$

is the smallest horizon length such that the MPC policy is feasible for the chosen  $x_0$ . Both these control laws are also compared to a CG applied to the LQR gain. All three controllers use  $Q = 100I$  and  $R = 1$ . The constraint set

$$\mathcal{Y}_3 = [-20, 20] \times [-1, 1] \times [-0.25, 0.25]$$

is chosen to illustrate what happens when the initial conditions  $x_0 = [-17, 0]^T$ , and reference  $r = 4$  are chosen far away from each other. As displayed in Figure 7, there is  $\approx 37\%$  increase in rise time using the FG, but the worst case computation time for the combined FG and MPC feedback policy is over 5000 times faster than of the un-governed MPC, as seen in Table I.

TABLE I

EXECUTION TIME DATA FOR THE DOUBLE INTEGRATOR EXAMPLE.

	FG ( $N = 10$ )	MPC ( $N = 10$ )	MPC ( $N = 236$ )	CG
TAVE [ms]	0.0126	0.0884	255	0.00833
TMAX [ms]	0.063	0.345	2170	0.0369

## VII. CONCLUSIONS

This paper introduced the Feasibility Governor (FG), an add-on unit that expands the region of attraction of linear model predictive controllers by filtering the reference input passed to the controller and is designed to interfere minimally with the operation and construction of the nominal controller. It was shown that the FG is safe, converges in finite time, and extends the region of attraction of MPC controllers at a fraction of the computation cost associated with increasing the prediction horizon. Future work includes extending FG theory to handle the case when  $G_z$  is not invertible and introducing an easier to compute approximation of the feasible set  $\Gamma_N$  to address the curse of dimensionality for larger systems.

## REFERENCES

- [1] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model predictive control: Theory, Computation and Design*. Nob Hill Pub., 2018.
- [2] G. Goodwin, M. M. Seron, and J. A. De Doná, *Constrained control and estimation: an optimisation approach*. Springer Science & Business Media, 2006.
- [3] D. Q. Mayne, “Model predictive control: Recent developments and future promise,” *Automatica*, vol. 50, no. 12, pp. 2967–2986, 2014.
- [4] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, “Constrained model predictive control: Stability and optimality,” *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.
- [5] H. Chen and F. Allgöwer, “A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability,” *Automatica*, vol. 34, no. 10, pp. 1205–1217, 1998.
- [6] A. H. González and D. Odloak, “Enlarging the domain of attraction of stable mpc controllers, maintaining the output performance,” *Automatica*, vol. 45, no. 4, pp. 1080–1085, 2009.
- [7] D. Limón, I. Alvarado, T. Alamo, and E. F. Camacho, “Mpc for tracking piecewise constant references for constrained linear systems,” *Automatica*, vol. 44, no. 9, pp. 2382–2387, 2008.
- [8] D. Simon, J. Löfberg, and T. Glad, “Reference tracking mpc using dynamic terminal set transformation,” *IEEE Transactions on Automatic Control*, vol. 59, no. 10, pp. 2790–2795, 2014.
- [9] L. Fagiano and A. R. Teel, “Generalized terminal state constraint for model predictive control,” *Automatica*, vol. 49, no. 9, pp. 2622–2631, 2013.
- [10] D. Limon, T. Alamo, and E. F. Camacho, “Enlarging the domain of attraction of mpc controllers,” *Automatica*, vol. 41, no. 4, pp. 629–635, 2005.
- [11] M. Herceg, M. Kvasnica, C. N. Jones, and M. Morari, “Multi-parametric toolbox 3.0,” in *Control Conference (ECC), 2013 European*, pp. 502–510, IEEE, 2013.
- [12] A. Löhne and B. Weißing, “The vector linear program solver bensolve—notes on theoretical background,” *European Journal of Operational Research*, vol. 260, no. 3, pp. 807–813, 2017.
- [13] L. Chisci and G. Zappa, “Dual mode predictive tracking of piecewise constant references for constrained linear systems,” *International Journal of Control*, vol. 76, no. 1, pp. 61–72, 2003.
- [14] S. Olaru and D. Dumur, “Compact explicit mpc with guarantee of feasibility for tracking,” in *Proceedings of the 44th IEEE Conference on Decision and Control*, pp. 969–974, IEEE, 2005.
- [15] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, “The explicit linear quadratic regulator for constrained systems,” *Automatica*, vol. 38, no. 1, pp. 3–20, 2002.
- [16] M. M. Nicotra, D. Liao-McPherson, and I. V. Kolmanovsky, “Embedding constrained model predictive control in a continuous-time dynamic feedback,” *IEEE Transactions on Automatic Control*, vol. 64, pp. 1932–1946, May 2019.
- [17] S. Di Cairano, A. Goldsmith, U. V. Kalabić, and S. A. Bortoff, “Cascaded reference governor–mpc for motion control of two-stage manufacturing machines,” *IEEE Transactions on Control Systems Technology*, vol. 27, no. 5, pp. 2030–2044, 2018.
- [18] 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, no. 9, pp. 1008–1020, 1991.
- [19] E. Garone, S. Di Cairano, and I. Kolmanovsky, “Reference and command governors for systems with constraints: A survey on theory and applications,” *Automatica*, vol. 75, pp. 306–328, 2017.
- [20] A. Bemporad, A. Casavola, and E. Mosca, “Nonlinear control of constrained linear systems via predictive reference management,” *IEEE transactions on Automatic Control*, vol. 42, no. 3, pp. 340–349, 1997.
- [21] D. Goldfarb and A. Idnani, “A numerically stable dual method for solving strictly convex quadratic programs,” *Mathematical programming*, vol. 27, no. 1, pp. 1–33, 1983.
- [22] D. Liao-McPherson and I. Kolmanovsky, “The fbstab quadratic programming method applied to model predictive control: An implicit condensing approach,” in *2019 IEEE 58th Conference on Decision and Control (CDC)*, pp. 3370–3376, 2019.
- [23] D. Ciripoi, A. Löhne, and B. Weißing, “Calculus of convex polyhedra and polyhedral convex functions by utilizing a multiple objective linear programming solver,” *Optimization*, 2018.
- [24] T. Huynh, C. Lassez, and J.-L. Lassez, “Practical issues on the projection of polyhedral sets,” *Annals of mathematics and artificial intelligence*, vol. 6, no. 4, pp. 295–315, 1992.
- [25] J. Hurt, “Some stability theorems for ordinary difference equations,” *SIAM Journal on Numerical Analysis*, vol. 4, no. 4, pp. 582–596, 1967.