MIT Libraries | DSpace@MIT

MIT Open Access Articles

Robust output feedback control with guaranteed constraint satisfaction

The MIT Faculty has made this article openly available. ***Please share*** how this access benefits you. Your story matters.

Citation: Sadraddini, Sadra and Russ Tedrake. "Robust output feedback control with guaranteed constraint satisfaction." Proceedings of the 23rd International Conference on Hybrid Systems: Computation and Control, April 2020, Sydney, Australia, Association for Computing Machinery, April 2020. © 2020 Association for Computing Machinery

As Published: http://dx.doi.org/10.1145/3365365.3382211

Publisher: Association for Computing Machinery (ACM)

Persistent URL: https://hdl.handle.net/1721.1/130045

Version: Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

Terms of use: Creative Commons Attribution-Noncommercial-Share Alike

Massachusetts Institute of Technology

Robust Output Feedback Control with Guaranteed Constraint Satisfaction

Sadra Sadraddini
Massachusetts Institute of Technology
Cambridge, MA, USA
sadra@mit.edu

Russ Tedrake
Massachusetts Institute of Technology
Cambridge, MA, USA
russt@mit.edu

ABSTRACT

We propose a method to control linear time-varying (LTV) discrete-time systems subject to bounded process disturbances and measurable outputs with bounded noise, and polyhedral constraints over system inputs and states. We search over control policies that map the history of measurable outputs to the current control input. We solve the problem in two stages. First, using the original system, we build a linear system that predicts future observations using the past observations. The bounded errors are characterized using zonotopes. Next, we propose control laws based on affine maps of such output prediction errors, and show that controllers can be synthesized using convex linear/quadratic programs. Furthermore, we can add constraints on trajectories and guarantee their satisfaction for all allowable sequences of observation noise and process disturbances. Our method does not require any assumptions about system controllability and observability. The controller design does not directly take into account the state-space dynamics, and its implementation does not require an observer. Instead, partial observability is often sufficient to design a correct controller. We provide the polytopic representation of observability errors and reachable sets in the form of zonotopes. Illustrative examples are included.

CCS CONCEPTS

• **Mathematics of computing → Mathematical optimization**; • **Computing methodologies → Planning under uncertainty**; **Computational control theory**.

KEYWORDS

Formal Synthesis, Output Feedback, Zonotopes, Model Predictive Control

ACM Reference Format:
Sadra Sadraddini and Russ Tedrake. 2020. Robust Output Feedback Control with Guaranteed Constraint Satisfaction. In *23rd ACM International Conference on Hybrid Systems: Computation and Control (HSCC '20), April 22–24, 2020, Sydney, NSW, Australia.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3365365.3382211

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
HSCC '20, April 22–24, 2020, Sydney, NSW, Australia
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-7018-9/20/04...$15.00
https://doi.org/10.1145/3365365.3382211

1 INTRODUCTION

Full state knowledge is neither realistic nor necessary for a broad range of control problems. Instead, controllers may use the measured system outputs to command inputs. There are, in general, two main approaches to output feedback control: (i) *static* output feedback, where the controller directly maps the current system outputs to control inputs; and (ii) *dynamic* output feedback, where the controls are a map of the latent states, which themselves evolve by measured outputs. Dynamic output feedback can be brought into static output feedback form by a special augmentation of the output space with latent states [26]. Finding the complete algorithmic solution to static output feedback design is one of the classical unsolved problems of control theory [36]. Instead, a wide range of incomplete solutions - methods that may find solutions for particular problems but are not able to verify the nonexistence of a solution - and special classes have been studied. In the most well-studied problem, the latent state space has the same dimension as the actual state space, and the dynamics governing the evolution of latent states are given by an *observer* such that the latent states are steered toward the actual states. For unconstrained linear systems with Gaussian noise and quadratic cost, it is well known that the *separation principle* holds: the optimal controller is Linear Quadratic Regulator (LQR) of the latent state, which itself is the best estimation of the actual state using Kalman filter [18]. Therefore, state estimation and control are decoupled. This scheme is known as Linear Quadratic Gaussian (LQG) control.

Despite its success, LQG control has two major limitations: (1) the need for observing and incorporating the full state into the control loop; and (2) the lack of robustness guarantees [7] as well as the inability to reason about constraint satisfaction. In order to overcome (2), the standard approach is replacing the Gaussian noise with bounded sets, and quadratic regulators with model predictive control (MPC) policies. However, issue (1) becomes even more severe as MPC needs to be implemented for high-dimensional latent state spaces. The majority of works on output feedback MPC are in principle MPC implemented for latent states, while taking into account the estimation error to guarantee robust constraint satisfaction [9, 27, 35, 39]. They deal with infinite-time specifications and require strong assumptions about controllability and observability. Furthermore, as the state dimension increases, these methods become too complicated to implement. For example, control of deformable objects typically have high dimensional state-space models that it seems unnatural to incorporate them in full state-feedback (or full observed-state-feedback) MPC schemes.

In order to not deal with very high-dimensional state spaces, model order reduction methods are ordinarily used. The aim is to

approximate the true dynamics with less complexity than the actual states. A decades old classical problem is reduced-order LQG, where the observer has arbitrarily fewer states than the actual system. It is well known that the separation principle no longer holds in this case as it gives rise to highly nonlinear, rank-constrained, optimal projection equations [17], for which various numerical methods have been proposed to find locally-optimal solutions [37, 38, 41]. However, convergence and global optimality are not, in general, guaranteed. The difficulties associated with reduced-order LQG highlight the grand challenge of simultaneous model order reduction and control.

In this paper, we revisit the problem of output feedback control without full state estimation. Instead of employing standard model-order reduction techniques, or designing an observer with fewer dimensions than the actual system, we build a linear model for output prediction based on the history of outputs and controls - using all the information available. Thus, we circumvent the explicit model order reduction and choices within. We design the predictor that minimizes the mismatch between the output prediction and the actual evolution, in a sense that is made clear in the paper. The error sets are time-varying. For instance, they may be large if some state variables are not measurable but heavily impact the dynamics, but uncertainty shrinks as time progresses and the history of outputs indirectly incorporates them into the prediction. Next, we design control policies based on the abstract output predictor. The only assumptions are the bounding sets for possible initial states, process disturbances, and measurement noises. The contributions and the organization of this paper are as follows:

(1) We characterize the set-valued error between the output predictor and actual full model using zonotopes, which are a highly flexible class of polytopes amenable to convenient propagation with affine transformations and Minkowski sums. We use zonotope order-reduction methods to gain tractability (Section 4).
(2) We parameterize the controller as an affine feedback of the errors of the abstract output predictor. The reachable sets become parameterized zonotopes. We specify the synthesis problem by polyhedral constraints and convex linear/quadratic costs. All the hard constraints become instances of polytope containment problems, which have efficient linear encodings [31]. Furthermore, we extend the results to infinite-time invariance for time-invariant or periodic jump linear systems via a containment approach. The parameters are synthesized in a correct-by-design fashion using a linear/quadratic convex program (section 5).
(3) We provide illustrative numerical examples on robust constrained control and comparisons to LQG, (Section 7).

Related Work

Policy Parameterization. Parameterizing control policies such that the synthesis program becomes convex is well studied. Examples include Youla parameterization [40] and LMI formulations for robust control [4]. In the case of full-state knowledge, it was first observed in [13] that if the policy is parameterized as a feedback of the history of disturbances rather than the state, then the set of parameters that lead to finite-time convex constraint satisfaction is convex. This observation is very useful in designing correct-by-design controllers for constrained linear systems via convex programs [12, 29]. A related parameterization was used in [30] to obtain robust controlled invariance. However, the disturbance-feedback paradigm can not directly be extended to the output-feedback case as the disturbances acting on state can not be measured. This paper takes the idea of disturbance-feedback but applies it to the errors in the output predictor.

The idea of parameterizing the control policy as an affine casual map of the history of outputs for constrained LTV system has also appeared in [33] and related ideas such system level synthesis [1]. The parameterization is carefully designed such that the synthesis program becomes convex. The main difference between our work and [1, 33] is the middle step of introducing the output predictor and the fact that the synthesis program is based on the output predictor - the notion of the full state is absent in the synthesis. We also provide insight to how observability errors are propagated. With the use of zonotope characterization, zonotope order-reduction, and polytope containment, we are able to provide a scalable solution to the worst case synthesis problem in [33] that does not rely on exhaustive approaches such as sampling. Finally, as opposed to [33], we also provide a solution to infinite-time synthesis for time-invariant or periodic systems.

Data-enabled Control. While our approach is not based on data, the abstract output predictor can be learned from Data (see Remark 1). In data-enabled predictive control (DeepC) [6, 16], the input-output data is directly used to find control inputs, skipping the model-identification. The technique exploits the superposition principle of linear systems to characterize the policy as a convex optimization problem. The robust version of [16] requires the past knowledge of disturbances, which are unknown in partially observable environments. Our approach does not suffer from this unrealistic assumption as policies are based on observable errors in the output prediction, not disturbances on state.

MPC for reduced order Models. Few recent works consider MPC methods that do not rely on the full-state estimation using reduced-order models. Ordinarily, the MPC approaches are independent of the method used for order-reduction. For instance, the *balanced-truncation* [32] is a popular method. The main challenge is propagating the order-reduction error dynamics and taking them into account for MPC constraint satisfaction. Earlier works [8, 14, 15, 28] avoided this issue and relied on softening the MPC constraints on-the-fly to accommodate possible violations. Stability was only guaranteed for open-loop stable systems [15]. Later, the authors in [34] provided a very conservative error propagation characterization for systems in which the unmeasured states evolved in an open-loop stable fashion, thus being able to reason about MPC constraint satisfaction by tightening the MPC constraints with the appropriate set-valued error terms. In order to reduce conservativeness, moving-horizon estimators of the unmeasured states were employed on the fly to make the tightened constraints time-varying and less conservative than what was computed offline. However, this later technique required full-state estimation. The same issue is present in [22], where the authors assume the knowledge of full

state but only use it for a linear feedback ancillary controller alongside MPC for the reduced order model. The works in [20, 24] solely rely on reduced-state feedback for MPC. The error bounds, similar to [34], are computed in a worst-case fashion for any control policy, and are computed prior to controller synthesis. In [24], a guess and verify approach is taken to find the polytopic error set, which is shown to be less conservative than [20]. However, these bounds may still be arbitrarily large and heavily depend on the user expertise for guessing the range of possible states/inputs. The aforementioned works take advantage of the infinite-time specifications and Linear Time Invariant (LTI) systems, and did not take the highly time-varying transient behavior into account.

2 PRELIMINARIES

In this section we provide the notation and necessary background on polytopes, zonotopes, and polytope containment encodings, all which we use in this paper.

2.1 Notation

The set of real, non-negative real, and non-negative integer numbers are denoted by $\mathbb{R}$, $\mathbb{R}_+$, and $\mathbb{N}$, respectively. The set of non-negative integers smaller or equal to T is shown by $\mathbb{N}_{\leq T}$. Given matrix A, we use A' and $A^\dagger$ to denote its transpose and Moore-Penrose inverse, respectively. The infinity norm of matrix A is denoted by $\|A\|_\infty$, which is the maximum absolute row sum. Given matrices $A_i, i = 1, \cdots, N$, with appropriate dimensions, we use the following stacking notations:

- $[A_1, \cdots, A_N]$ to stack matrices vertically,
- $(A_1, \cdots, A_N)$ to stack matrices horizontally,
- $\text{blk}(A_1, \cdots, A_N)$ to construct a block-diagonal matrix by stacking them diagonally,
- $[A_1, \cdots, A_N]_\triangle$ is the lower-triangular matrix by stacking them vertically and letting remaining entries to be zero.

Given $A \in \mathbb{R}^{n\times n}$, the trace of the matrix is denoted by $\text{tr}(A)$. For $A \in \mathbb{R}^{n_A\times n}$ and $\mathbb{S} \subset \mathbb{R}^n$, we interpret $A\mathbb{S}$ as $\{As | s \in \mathbb{S}\}$. Given $\mathbb{S}_1, \mathbb{S}_2 \subset \mathbb{R}^n$, their Minkowski sum is $\mathbb{S}_1 \oplus \mathbb{S}_2 = \{s_1 + s_2 | s_1 \in \mathbb{S}_1, s_2 \in \mathbb{S}_2\}$. We interpret $s + \mathbb{S}$, $s \in \mathbb{R}^n$, as $\{s\} \oplus \mathbb{S}$. Given two sets $\mathcal{A}, \mathcal{B}$, their Cartesian product is $\mathcal{A} \otimes \mathcal{B} := \{(a, b) | a \in \mathcal{A}, b \in \mathcal{B}\}$. The $n\times n$ identity matrix, $n\times m$ zero matrix, and n-dimensional vector of all ones are denoted by I_n, $0_{n\times m}$, and 1_n, respectively. All inequality relations involving matrices are interpreted element-wise in this paper.

With a slight abuse of notation, we call two random variables a and b independent if a and b can take all the elements in $\mathcal{A}$ and $\mathcal{B}$, respectively, then (a, b) can take all the values in $\mathcal{A}\times\mathcal{B}$. We did not use the term *statistically independent* as the underlying probability distributions of a and b may be unknown. We only require the knowledge of their supports.

2.2 Polytopic Objects and Operations

An *H-polyhedron* is a set defined by a finite number of linear inequalities:

$$\mathbb{H} = \{x \in \mathbb{R}^n | Hx \leq h\}, \tag{1}$$

where $H \in \mathbb{R}^{q\times n}$, $h \in \mathbb{R}^q$. A bounded H-polyhedron is called an *H-polytope*. Any set that can be written as a H-polytope is a polytope. The n-dimensional p-norm unit ball is denoted by: $\mathbb{B}_n^p := \{x \in \mathbb{R}^n | \|x\|_p \leq 1\}$. Note that $\mathbb{B}_n^p$ is a polytope for $p \in \{1, \infty\}$. For instance, the ∞-norm unit ball is a hyperbox centered at origin. Its H-polytope form is:

$$\mathbb{B}_n^\infty := \{x \in \mathbb{R}^n | [I_n, -I_n]x \leq 1_{2n}\}. \tag{2}$$

An affine transformation of a polytope is still a polytope. A *zonotope* is an affine transformation of a hyperbox, and is represented as

$$\mathbb{Z} := \langle \bar{x}, G\rangle := \bar{x} + G\mathbb{B}_p^\infty, \tag{3}$$

where $\bar{x}$ is the zonotope centorid and G is the matrix G is the *generator*. Zonotopes are popular because of their properties with affine transformations and Minkowski sums. We have:

$$A\langle \bar{x}, G\rangle + b = \langle A\bar{x} + b, AG_z\rangle, \tag{4a}$$

$$\langle \bar{x}_1, G_1\rangle \oplus \langle \bar{x}_2, G_2\rangle = \langle \bar{x}_1 + \bar{x}_2, (G_1, G_2)\rangle, \tag{4b}$$

where all matrices in (4a) and (4b) have appropriate dimensions. Notice the increase in the number of columns in the zonotope generator.

In general, an affine transformation of an H-polytope can be transformed into a H-polytope but its number of inequalities may be exponentially large [19]. A special case is when the linear map has a left-inverse. Then $\bar{x} + G\mathbb{H} = \{y \in \mathbb{R}^n | HX^\dagger y \leq h + HX^\dagger \bar{x}\}$. s

2.3 Polytope Containment

We state the result in [31]. Given two sets in $\mathbb{R}^n$, $\mathbb{Q}_i = \bar{x}_i + G_i\mathbb{H}_i$, $\mathbb{H}_i = \{x \in \mathbb{R}^{n_i} | H_i x \leq h_i\}$, $i = 1, 2$, a sufficient condition for $\mathbb{Q}_1 \subseteq \mathbb{Q}_2$ is the existence of Γ, Λ, and β, with appropriate dimensions, such that:

$$G_1 = G_2\Gamma, \Lambda H_1 = H_2\Gamma, \tag{5a}$$

$$\bar{x}_2 - \bar{x}_1 = G_2\beta, \Lambda h_1 \leq h_2 + H_2\beta, \Lambda \geq 0. \tag{5b}$$

An implication of (5) is that if any or all of $G_1, \bar{x}_1, \bar{x}_2$, and h_2 are decision variables in a mathematical program, then (5) are a set of linear constraints on those decisions. We exploit this property in this paper to constrain the parameters of controllers by containment of reachable sets in target sets.

The encoding in (5) is based on the duality of linear programs. Another practical application of polytope containment is the following relation:

$$\max_{\zeta \in \mathbb{B}_q^\infty} \|\bar{x} + G\zeta\|_p = \begin{array}{ll} \min & r \\ \text{subject to} & \langle \bar{x}, G\rangle \subseteq r\mathbb{B}_n^p. \end{array} \tag{6}$$

We use (6) to turn min-max optimization problems in model predictive control to standard minimization in Section 5.4. We also note that the subset relation in (6) is often conservative, but there still exist instances that the subset relation is tight (see [31] for more details).

2.4 Zonotope Order Reduction

The zonotope *order* with $G \in \mathbb{R}^{n\times p}$ is defined as the ratio $\frac{p}{n}$. In practice, most zonotopes have order greater than 1. Zonotopes with order smaller than 1 have zero volume. A zonotope with order one that has non-zero volume is also known as *parallelotope*. Having zonotopes with very large orders is undesirable. There exists a variety of methods [21] to over-approximate $\mathbb{Z}$ with $\mathbb{Z}^{\text{Redu.}}$ such that $\mathbb{Z} \subseteq \mathbb{Z}^{\text{Redu.}}$ with the order of $\mathbb{Z}^{\text{Redu.}}$ being arbitrarily small, as long

is greater than or equal to one. In this paper, we use the algebraic method in [10] as it is fast, convenient, and not very conservative. Less conservative but slower optimization-based approaches are also available [21, 31].

3 PROBLEM STATEMENT

Consider the following LTV system:

$$x_{t+1} = A_t x_t + B_t u_t + w_t, w_t \in \mathbb{W}_t, \tag{7a}$$

$$y_t = C_t x_t + v_t, v_t \in \mathbb{V}_t, \tag{7b}$$

$$z_t = D_t x_t + d_t, \tag{7c}$$

where $x_t \in \mathbb{R}^n$, $u_t \in \mathbb{R}^m$, $y_t \in \mathbb{R}^o$, $z_t \in \mathbb{R}^z$, $w_t \in \mathbb{W}_t$ and $v_t \in \mathbb{V}_t$ are system state, control inputs, measured outputs, performance variables, process disturbance, and measurement noise, respectively. The term $d_t \in \mathbb{R}^z$ is the performance variable offset, and is given and deterministic. The initial state x_0 is unknown but restricted to a given set $\mathbb{X}_0$. We later assume that sets $\mathbb{W}_t, \mathbb{V}_t, \mathbb{X}_0$ are given zonotopes and the associated random variables are independent. We use the capital letter notation to refer to the history of variables. For instance, $Y_t := [y_0, y_1, \cdots, y_t]$. The same notation follows for U_t, W_t, and V_t. At time t, the raw information available is Y_t and U_{t-1}. An output feedback control policy takes the form of $\Pi = \{\pi_t | t = 0, 1, \cdots, T-1\}$, where $T \in \mathbb{N} \cup \{\infty\}$ is the horizon and π_t are functions such that:

$$u_t = \pi_t(Y_t, U_{t-1}), t \in \mathbb{N}_{\leq T}. \tag{8}$$

A *trajectory* is defined as the time series of values taken in control input-performance output space:

$$\sigma = (z_0, u_0), (z_1, u_1), \cdots, (z_{T-1}, u_{T-1}), z_T. \tag{9}$$

Given Π, the set of all possible trajectories of the system in the input-output space is denoted by $\Sigma(\Pi)$ and is given as:

$$\{\sigma | \text{Eq.(7)}, \text{Eq.(8)}, x_0 \in \mathbb{X}_0, w_t \in \mathbb{W}_t, v_t \in \mathbb{V}_t, t \in \mathbb{N}_{\leq T}\}.$$

Problem 1. *Given* (7)*, a reference trajectory* σ^{ref}*, polyhedral sets* $\mathbb{H}_t^z \subset \mathbb{R}^z, \mathbb{H}_t^u \subset \mathbb{R}^m, t \in \mathbb{N}_{\leq T}$*, and terminal set* $\mathbb{H}_T^z \subset \mathbb{R}^z$*, find the optimal control policy* Π^* *such that:*

$$\begin{aligned} \Pi^* = \arg\min_{\Pi} \quad & \max_{\sigma \in \Sigma(\Pi)} \quad J(\sigma) \\ & \textit{subject to} \quad z_t \in \mathbb{H}_t^z, u_t \in \mathbb{H}_t^u, t \in \mathbb{N}_{\leq T}, \\ & \qquad\qquad z_T \in \mathbb{H}_T^z, \forall \sigma \in \Sigma(\Pi), \end{aligned} \tag{10}$$

where

$$J(\sigma) = \sum_{t=0}^{T-1} (\|z_t - z_t^{ref}\|_{\mathcal{Z}_t} + \|u_t - u_t^{ref}\|_{\mathcal{U}_t}) + \|z_T - z_T^{ref}\|_{\mathcal{Z}_T},$$

and $\|.\|_.$ *are user defined norms.*

For the next problem, we assume the system is time invariant and the specifications is infinite-time set-invariance.

Problem 2. *Given LTI version of system* (7)*, polyhedral sets* $\mathbb{H}^z \subset \mathbb{R}^z, \mathbb{H}^u \subset \mathbb{R}^m$*, find the infinite-horizon control policy* Π^* *such that:*

$$\forall \sigma \in \Sigma(\pi^*), x_t \in \mathbb{H}_x, u_t \in \mathbb{H}_u, \forall t \in \mathbb{N}. \tag{11}$$

In this paper, the main focus is on the robust constraint satisfaction in (10) and (11) - guaranteeing constraints for all possible closed-loop trajectories produced by Π. For Problem 2, we can consider periodic LTV systems as well but we omit the details. For Problem 1, alternatives or approximations for the cost function are possible. For instance, we may wish to minimize the cost associated with a nominal trajectory instead of the worst-case one. The choice of norms are typically p-norms, with $p \in \{1, 2, \infty\}$. Exact minimax optimization with 2-norms is inefficient [3, 23], but we still make approximations that render it convex and efficient. The details are outlined in Section 5.4.

4 OUTPUT PREDICTOR

In this section, we take the first step toward solving Problem 1. We abstract the system (7) into a linear model in which only outputs are present. We make the abstraction sound by considering time-varying error sets such that the output predictor contains all the output behaviors of (7).

4.1 Observability

We can not measure x_0, W_t or V_t. The only information we have at time t is Y_t and U_t (after making the decision of u_t). Using this information, we are able to build a model for the next measurable output and the current performance variable. We propose the following linear model:

$$y_{t+1} = \mathbf{M}_t Y_t + \mathbf{N}_t U_t + e_t, e_t \in \mathbb{E}_t, \tag{12a}$$

$$z_t = \mathbf{R}_t Y_t + \mathbf{S}_t U_t + f_t, f_t \in \mathbb{F}_t, \tag{12b}$$

where $\mathbf{M}_t, \mathbf{N}_t, \mathbf{R}_t, \mathbf{S}_t$ are matrices of parameters with appropriate dimensions, and $e_t \in \mathbb{E}_t, f_t \in \mathbb{F}_t$ are the error terms. The bolded notation in this section highlights the decision variables that we aim to compute in this section. For any values chosen for $\mathbf{M}_t, \mathbf{N}_t, \mathbf{R}_t, \mathbf{S}_t$, there exist error sets that satisfy (12). The aim is to choose the aforementioned matrices such that these error sets are small, so their propagation over time in (12) does not lead to very large sets for possible performance outputs.

First, we introduce additional notation. Given an initial state x_0, control sequence $u_0, u_1, \cdots, u_t$, and disturbances $w_0, w_1, \cdots, w_t$, the state at time t is given by:

$$x_t = P_t^x x_0 + P_t^u U_{t-1} + P_t^w W_{t-1}, t > 0, \tag{13}$$

where the matrices are recursively constructed as follows:

$$P_{t+1}^x = A_t P_t^x, P_1^x = A_0, \tag{14a}$$

$$P_{t+1}^u = (A_t P_t^u, B_t), P_1^u = B_0, \tag{14b}$$

$$P_{t+1}^w = (A_t P_t^w, I_n), P_1^w = I_n, \tag{14c}$$

Note that the sizes of P^u, P^w grow linearly with time. We have $y_0 = C_0 x_0 + v_0$. The measured output at $t \geq 1$ is:

$$y_t = C_t P_t^x x_0 + C_t P_t^u U_{t-1} + C_t P_t^w W_{t-1} + v_t. \tag{15}$$

For the ease of notation used later in the paper, we introduce the following compact form:

$$Y_t = Q_t^x x_0 + Q_t^u U_t + Q_t^w W_t + Q_t^v V_t, \tag{16}$$

where

$$Q_{t+1}^x = [Q_t^x, C_{t+1} P_{t+1}^x], Q_t^0 = C_0 \tag{17a}$$

$$Q_{t+1}^u = [Q_t^u, (C_{t+1} P_{t+1}^u, 0_{o \times m})]_\Delta, Q_0^u = 0_{o \times m}, \tag{17b}$$

$$Q^w_{t+1} = [Q^w_t, (C_{t+1}P^w_{t+1}, 0_{o\times n})]_\Delta, Q^w_0 = 0_{o\times n}, \tag{17c}$$

Note that Q^u_t, Q^w_t are strictly lower-triangle matrices since y_t does not depend on u_t and w_t. The matrix Q^v_t is $I_{o(t+1)}$ as $Q^v_t V_t = V_t$. The performance variable at time t is:

$$z_t = D_t P^x_t x_0 + D_t P^u_t U_{t-1} + D_t P^w_t W_{t-1} + d_t. \tag{18}$$

Replacing (16) in (12) will yield the following:

$$y_{t+1} = \mathbf{M}_t Q^x_t x_0 + \mathbf{M}_t Q^u_t U_t + \mathbf{M}_t Q^w_t W_t + \mathbf{M}_t Q^v_t V_t + \mathbf{N}_t U_t + e_t, \tag{19a}$$

$$z_t = \mathbf{R}_t Q^x_t x_0 + \mathbf{R}_t Q^u_t U_t + \mathbf{R}_t Q^w_t W_t + \mathbf{R}_t Q^v_t V_t + \mathbf{S}_t U_t + f_t. \tag{19b}$$

By comparing it to (15) and (18), the error terms become:

$$\begin{aligned} e_t &= (C_{t+1}P^x_{t+1} - \mathbf{M}_t Q^x_t)x_0 + (C_{t+1}P^u_{t+1} - \mathbf{N}_t - \mathbf{M}_t Q^u_t)U_t \\ &\quad + (C_{t+1}P^w_{t+1} - \mathbf{M}_t Q^w_t)W_t - \mathbf{M}_t Q^v_t V_t + v_{t+1}. \\ f_t &= \big(D_t P^x_t - \mathbf{R}_t Q^x_t\big)x_0 + \big((D_t P^u_t, 0_{z\times m}) - \mathbf{S}_t - \mathbf{R}_t Q^u_t\big)U_t \\ &\quad + \big((D_t P^w_t, 0_{z\times n}) - \mathbf{R}_t Q^w_t\big)W_t - \mathbf{R}_t Q^v_t V_t + d_t. \end{aligned}$$

We investigate (4.1) and the characterization of $\mathbb{E}_t$. The analysis for $\mathbb{F}_t$ is similar. We choose $\mathbf{N_t} = C_{t+1}P^u_{t+1} - \mathbf{M}_t Q^u_t$ to cancel the second term. We first make a common mild assumption.

Assumption 1. *The random values $x_0 \in \mathbb{X}_0, w_t \in \mathbb{W}_t, v_t \in \mathbb{V}_t, t \in \mathbb{N}_{\leq T}$, are independent.*

Using Assumption 1, the error set becomes:

$$\begin{aligned} \mathbb{E}_t = \; & (C_{t+1}P^x_{t+1} - \mathbf{M}_t Q^x_t)\mathbb{X}_0 \\ & \oplus (C_{t+1}P^w_{t+1} - \mathbf{M}_t Q^w_t) \textstyle\bigotimes_{\tau=0}^{t} \mathbb{W}_\tau \\ & \oplus \mathbf{M}_t Q^v_t \textstyle\bigotimes_{\tau=0}^{t} \mathbb{V}_\tau \\ & \oplus \mathbb{V}_{t+1}. \end{aligned} \tag{20}$$

Notice that we identify three additive, assumed to be independent, sources of error:

The Initial State. The term $(C_{t+1}P^x_{t+1} - \mathbf{M}_t Q^x_t)\mathbb{X}_0$ is the effect of the uncertainty in initial state at time t. If there exists $\mathbf{M}_t$ such that $(C_{t+1}P^x_{t+1} - \mathbf{M}_t Q^x_t) = 0$, then this uncertainty vanishes. This condition is equivalent to observability. In the LTI case, we have $Q^x_t = [C, CA, CA^2, \cdots, CA^t]$, which has a left inverse if it is full column rank, thus enabling $(CP^x_{t+1} - \mathbf{M}_t Q^x_t) = 0$ by choosing $\mathbf{M}_t = Q^{x\dagger}_t CP^x_{t+1}$. Even when observability does not hold, this term still captures the uncertainty that is used to compute the reachable sets.

Process Disturbances. These act similarly to the uncertainty in the initial state as they may become observable after some time. However, it is not possible to drive this term to zero as the most recent disturbances are not observable unless $C_.$ matrices are invertible.

Measurement Noise. The third term acts as a regularizer for choices of $\mathbf{M_t}$, penalizing large values as they amplify the effect of measurement noises in (12).

4.2 Zonotope Characterization

Assumption 2. *The sets $\mathbb{X}_0, \mathbb{W}_t, \mathbb{V}_t, t \in \mathbb{N}_{\leq T}$, are zonotopes.*

Assumption 2 is reasonable in most applications. Even when it is not the case, any set can be over-approximated by a zonotope. The assumption is inappropriate when the uncertainty sets are highly asymmetric sets. Our framework remains valid for any representation as affine maps of H-polytopes for the sets above. The zonotope assumption is mainly for convenient notation and computation, as we use zonotope order reduction techniques later in the paper.

The sets $\mathbb{E}_t$ and $\mathbb{F}_t$ becomes zonotopes. In order to provide their representation, we provide the following extra notation:

$$\mathbb{X}_0 = \langle \bar{x}_0, G^{X_0} \rangle, \mathbb{W}_t = \langle \bar{w}_t, G^{w_t} \rangle, \mathbb{V}_t = \langle \bar{v}_t, G^{v_t} \rangle. \tag{21}$$

Then we have:

$$\mathbb{E}_t = \langle \bar{e}_t, G^{e_t} \rangle, \mathbb{F}_t = \langle \bar{f}_t, G^{f_t} \rangle, \tag{22}$$

where:

$$\begin{aligned} \bar{e}_t = \; & (C_{t+1}P^x_{t+1} - \mathbf{M}_t Q^x_t)\bar{x}_0 + \\ & (C_{t+1}P^w_{t+1} - \mathbf{M}_t Q^w_t)\bar{W}_t + \mathbf{M}_t Q^v_t \bar{V}_t + \bar{v}_{t+1}, \\ G^{e_t} = \; & \big((C_{t+1}P^x_{t+1} - \mathbf{M}_t Q^x_t)G^{X_0}, \\ & (C_{t+1}P^w_{t+1} - \mathbf{M}_t Q^w_t)G^{W_t}, \mathbf{M}_t Q^v_t G^{V_t}, G^{v_{t+1}}\big), \\ \bar{f}_t = \; & (D_t P^x_t - \mathbf{R}_t Q^x_t)\bar{x}_0 + ((D_t P^w_t, 0_{z\times n}) \\ & - \mathbf{R}_t Q^w_t)\bar{W}_t - \mathbf{R}_t Q^v_t \bar{V}_t + d_t, \\ G^{f_t} = \; & \big((D_t P^x_t - \mathbf{R}_t Q^x_t)G^{X_0}, \\ & ((D_t P^w_t, 0_{z\times n}) - \mathbf{R}_t Q^w_t)G^{W_t}, -\mathbf{R}_t Q^v_t G^{V_t}\big), \end{aligned}$$

and $\bar{W}_t = [\bar{w}_0, \cdots, \bar{w}_t]$ and $G^{W_t} = \mathrm{blk}(G^{w_0}, \cdots, G^{w_t})$. The same notation follows for $\bar{V}_t$ and G^{V_t}.

4.3 Optimal Parameters for Output Predictor

Now we pick the best model such that the entries in G^{e_t} and G^{f_t} are small values. We wish to solve the following optimization problem:

$$\mathbf{M}_t, \mathbf{N}_t = \arg\min J(G^{e_t}), \tag{23a}$$

$$\mathbf{R}_t, \mathbf{S}_t = \arg\min J(G^{f_t}), \tag{23b}$$

where J is a cost function promoting smaller volumes for $\mathbb{E}_t$ and $\mathbb{F}_t$. We may consider the following cost functions.

4.3.1 Analytical Least Squares. A simple choice is the Frobenius norm:

$$J(G) = \mathrm{tr}(G'G). \tag{24}$$

Then we have the following closed-form minimizers:

$$\begin{aligned} \mathbf{M}_t = \; & \big((C_{t+1}P^x_{t+1}G^{X_0}, C_{t+1}P^w_{t+1}G^{W_t}, 0_{o\times q_1}, G^{v_{t+1}}\big) \\ & \big(Q^x_t G^{X_0}, Q^w_t G^{W_t}, Q^v_t G^{V_t}, 0_{(t+1)o\times q_2}\big)^\dagger, \end{aligned} \tag{25a}$$

$$\mathbf{N}_t = C_{t+1}P^u_{t+1} - \mathbf{M}_t Q^u_t, \tag{25b}$$

$$\begin{aligned} \mathbf{R}_t = \; & \big((D_t P^x_t G^{X_0}, (D_t P^w_t, 0_{z\times n})G^{W_t}, 0_{z\times q_1}\big) \\ & \big(Q^x_t G^{X_0}, Q^w_t G^{W_t}, -Q^v_t G^{V_t},\big)^\dagger, \end{aligned} \tag{25c}$$

$$\mathbf{S}_t = (D_t P^u_t, 0_{z\times n}) - \mathbf{R}_t Q^u_t, \tag{25d}$$

where q_1 and q_2 are the number of columns in G^{v_t} and $G^{v_{t+1}}$, respectively.

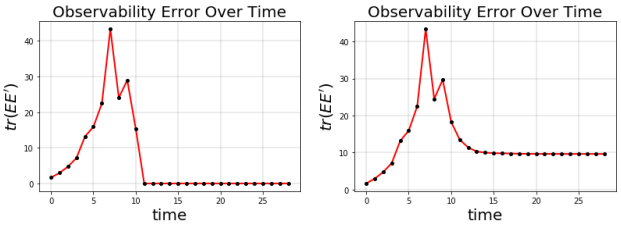

Figure 1: Example 2: Observavility error as characterized by the Frobenious norm of the generator of $\mathbb{E}_t$ for an open-loop unstable system with $n = 12$. [left] without disturbances/noise. The error is zero for $t \geq 11$, when observability holds. [Right] with disturbances/noise.

4.3.2 Convex Programming with Zonotope Containment. An alternative is to propose a zonotope $\mathbb{Z}^{\text{proposed}}$ such that $\mathbb{E}_t \subseteq \alpha\mathbb{Z}^{\text{proposed}}$ for some scalar α. Then we find can find $\mathbf{M}_t$ using the following polytope containment problem:

$$\mathbf{M}_t = \begin{array}{ll} \arg\min & \alpha \\ \text{subject to} & \mathbb{E}_t \subseteq \alpha\mathbb{Z}^{\text{proposed}}. \end{array} \tag{26}$$

By the virtue of encoding in (5), (26) is a convex linear program. A similar optimization problem to (26) can be set up for finding $\mathbf{R}_t$ by containing $\mathbb{E}_t$ in a proposed zonotope. An advantage of optimization-based approach is that we can write it for multiple instances of system (7), hence introducing robustness. The major disadvantage is relying on user expertise to choose $\mathbb{Z}^{\text{proposed}}$. Simple guesses, such as hyperboxes, may be too conservative.

Remark 1 (Learning parameters in (12) directly from data). *The way we find the parameters* (12) *is from the original state-space dynamics* (7). *For unknown systems, the state-space model is identified from the data. One possibility is to circumvent learning* (7) *is to directly learn* (12) *from data - removing the notion of full-state in the whole identification and synthesis process. A potential advantage is generalizability to classes of systems where* (7) *may be very different among instances but* (12) *may be relatively similar. We leave the investigation of this research direction to future work.*

5 FINITE-TIME POLICY SYNTHESIS

In this section, we solve Problem 1 by designing a control policy for the linear predictor (12) from the optimization problem (10). The technique is based on polytopic trajectories [31] and is the closely related to convex optimization for disturbance-feedback synthesis in [29] and system level synthesis in [1]. We provide polytopic representation of possible reachable sets - represented by zonotopes in this paper. These sets are pre-computed.

5.1 Control Policy Parameterization

We propose the parametric structure of the controller such that we synthesize the parameters using convex optimization. We propose the control input to be an affine feedback of the past errors in (12a). We show that this makes all the constraints linear, and hence convex. The policy is:

$$u_t = \tilde{u}_t + \theta_t[y_0, e_0, e_1, \cdots, e_{t-1}], \tag{27}$$

where $\tilde{u}_t \in \mathbb{R}^m$ and $\theta_t \in \mathbb{R}^{m\times o(t+1)}$ are the offset vector and matrix parameters that we wish to synthesize. The vector $[y_0, e_0, e_1, \cdots, e_{t-1}]$ is going to appear often in this section. Thus, we introduce the notation:

$$\xi_0 := y_0, \xi_t := [y_0, e_0, e_1, \cdots, e_{t-1}], t \in \mathbb{N}_{\leq T}. \tag{28}$$

The values in ξ_t belongs to the following zonotope

$$\Xi_t := \langle \bar{\xi}_t, G^{\xi_t} \rangle. \tag{29}$$

The way to find expressions for $\bar{\xi}_t, G^{\xi_t}$ is a bit subtle because the error terms through time are **not** independent variables. Treating them independently is very conservative - it is analogous to restarting the intial condition and all previous disturbance and noises at every time step. Instead, we characterize ξ_t based on the independent random variables x_0, w_t, v_t. Let:

$$\xi_t = \Xi_t^x x_0 + \Xi_t^w W_t + \Xi_t^v V_t, \tag{30}$$

where matrices $\Xi_t^x, \Xi_t^w, \Xi_t^v$ are constructed as follows:

$$\Xi_{t+1}^x = [\Xi_t^x, C_{t+1}P_{t+1}^x - M_t Q_t^x]_\Delta, \Xi_0^x = C_0. \tag{31}$$

$$\Xi_{t+1}^w = [\Xi_t^w, (C_{t+1}P_{t+1}^w - M_t Q_t^w, 0_{o\times n})]_\Delta, \Xi_0^w = 0_{o\times n}. \tag{32}$$

$$\Xi_{t+1}^v = [\Xi_t^v, (M_t Q_t^v, I_o)]_\Delta, \Xi_0^v = I_o. \tag{33}$$

Then zonotope characterization of Ξ_t is:

$$\bar{\xi}_t = \Xi_t^x \bar{x}_0 + \Xi_t^w \bar{W}_{t-1} + \Xi_t^v \bar{V}_t, \tag{34}$$

$$G^{\xi_t} = (\Xi_t^x G^{x_0}, \Xi_t^w G^{W_{t-1}}, \Xi_t^v G^{V_t}). \tag{35}$$

5.2 Encoding the dynamics

Under the control policy (27), the closed-loop response is an affine map of ξ_t. We introduce the following:

$$y_t = \tilde{y}_t + \phi_t \xi_t, Y_t = \tilde{Y}_t + \Phi_t \xi_t, \tag{36a}$$

$$u_t = \tilde{u}_t + \theta_t \xi_t, U_t = \tilde{U}_t + \Theta_t \xi_t, \tag{36b}$$

where $\tilde{Y}_t = [\tilde{y}_0, \tilde{y}_1, \cdots, \tilde{y}_t]$ (the same notation stands for $\tilde{U}_t$), and:

$$\Phi_t = [\phi_0, \phi_1, \cdots, \phi_t]_\Delta, \tag{37a}$$

$$\Theta_t = [\theta_0, \theta_1, \cdots, \theta_t]_\Delta, \tag{37b}$$

Plugging the equations into (12a) and rearranging terms produces the following encoding of the dynamics:

$$\tilde{y}_{t+1} = M_t \tilde{Y}_t + N_t \tilde{U}_t, \tag{38a}$$

$$\phi_{t+1} = (M_t \Phi_t + N_t \Theta_t, I_o). \tag{38b}$$

Finally, we have the following initial conditions:

$$\tilde{y}_0 = 0, \phi_0 = I_o. \tag{39}$$

5.3 Encoding the Performance Outputs

We have:

$$z_t = R_t(\tilde{Y}_t + \Phi_t\zeta_t) + S_t(\tilde{U}_t + \Theta_t\zeta_t) + f_t, \tag{40}$$

Then the performance variable and control input falls into the following zonotope:

$$z_t \in \mathbb{Z}_t, \mathbb{Z}_t = \langle \bar{z}_t, G^{z_t} \rangle, \tag{41a}$$

$$u_t \in \mathbb{U}_t, \mathbb{U}_t = \langle \bar{u}_t, G^{u_t} \rangle, \tag{41b}$$

By writing (41b) using the independent uncertain variables, we have:

$$\begin{aligned} z_t = \quad & (R_t\Phi_t\Xi_t^x + S_t\Theta_t\Xi_t^x + D_tP_t^x - R_tQ_t^x)x_0 + \\ & (R_t\Phi_t\Xi_t^w + S_t\Theta_t\Xi_t^w + (D_tP_t^w, 0_{z\times n}) - R_tQ_t^w))W_t + \\ & (R_t\Phi_t\Xi_t^v + S_t\Theta_t\Xi_t^v - R_tQ_t^v)V_t \end{aligned} \tag{42}$$

Therefore, we have the following compact form for the zonotope characterization of $\mathbb{Z}_t$:

$$\bar{z}_t = R_t(\tilde{Y}_t + \Phi_t\bar{\xi}_t) + S_t(\tilde{U}_t + \Theta_t\bar{\xi}_t) + \bar{f}_t, \tag{43a}$$

$$\begin{aligned} G^{z_t} = \quad & ((R_t\Phi_t\Xi_t^x + S_t\Theta_t\Xi_t^x + D_tP_t^x - R_tQ_t^x)G^{x_0}, \\ & (R_t\Phi_t\Xi_t^w + S_t\Theta_t\Xi_t^w + (D_tP_t^w, 0_{z\times n}) - R_tQ_t^w))G^{W_t} \\ & (R_t\Phi_t\Xi_t^v + S_t\Theta_t\Xi_t^v - R_tQ_t^v)G^{V_t}) \end{aligned} \tag{43b}$$

The parameters of the control zonotope are:

$$\bar{u}_t = \tilde{u}_t + \theta_t\bar{\xi}_t, \tag{44a}$$

$$G^{u_t} = \theta_t G_t^{\xi}. \tag{44b}$$

5.4 Control Policy Synthesis via Convex Linear/Quadratic Programming

Now we are in the position to synthesize the parameters of the controller. The policy decision variables are $\tilde{u}_\tau, \theta_\tau, \tau \in \mathbb{N}_{\leq T-1}$. The rest are variables for encoding.

Theorem 1 (Solution to Problem 1). *Problem* (10) *is cast as the following convex optimization problem*

$$\begin{aligned} \pi^* = \quad & \arg\min_\pi \quad r_T^z + \sum_{t=0}^{T-1}(r_t^z + r_t^u), \\ & \textit{subject to} \quad \langle \bar{z}_t - z_t^{ref}, G^{z_t} \rangle \subseteq r_t^z\mathbb{B}_p, t \in \mathbb{N}_{\leq T} \\ & \qquad \langle \bar{u}_t - u_t^{ref}, G^{u_t} \rangle \subseteq r_t^u\mathbb{B}_p, \\ & \qquad \langle \bar{z}_t, G^{z_t} \rangle \in \mathbb{H}_t^z, \langle \bar{u}_t, G^{u_t} \rangle \in \mathbb{H}_t^u, \\ & \qquad t \in \mathbb{N}_{\leq T-1}, \langle \bar{z}_T, G^{z_T} \rangle \in \mathbb{H}_T^z, \\ & \qquad \textit{Eq.}(43), (44). \end{aligned} \tag{45}$$

Proof. The proof is constructive from the introduced encoding and the min-max to min-min duality (6), □

Note that one may replace the r values in the cost with quadratic r^2 to obtain a convex quadratic program.

The optimization problem (45) is solved offline. Once it is solved, implementing the linear error-feedback policy online is very efficient as it boils down to matrix multiplications and summations in (8) - no real-time optimization is necessary. Resolving (45) online to adjust it to online measurements (such as an updated knowledge of possible initial conditions) can potentially improve performance, but its empirical investigation is beyond the scope of this paper.

5.5 Computational Complexity

5.5.1 Encoding Dynamics. The number of decision variables corresponding to the control policy in (45) is $mT(T+1)/2 + mT$. The number of variables in Φ_T is of order $O(oT^2)$. The number of variables and constraints corresponding to dynamics encoding grows with the order $O((o+m)T^2)$. A simple technique to decrease the number of decision variables is making Θ *block-banded* [33], where the values of θ_t that correspond to errors older than τ steps before are set to zero. By assuming fixed band τ, the number of policy decision variables becomes linear in τ, T.

5.5.2 Encoding Min-Max Optimization. The rest of the complexity roots in the polytope containment encoding, which depend on the order of the zonotopes that represent uncertainty sets. Let the zonotope order $\rho_x n$ be the order of $\mathbb{X}_0$, and $\rho_w n, \rho_v o$ be the average zonotope order of $\mathbb{W}_t$ and $\mathbb{V}_t$ sets over $t = 0, 1, \cdots, T-1$. Then the number of generators in both $\mathbb{Z}_t$ and $\mathbb{U}_t$ is $O\big(n\rho_x + t(n\rho_w + o\rho_v)\big)$. Given the average number of hyperplanes in the constraint sets be $\rho_h^u m$ for control constraints and $\rho_h^z z$ performance variables, the encoding of polytope containment from (5) requires the following order of variables and constraints:

$$O\big(\max(n\rho_h^x, m\rho_h^u, n, m)(n\rho_w + o\rho_v)T^2\big). \tag{46}$$

If all zonotopes are paralleltopes, then all values related to indices of ρ are one. The more complex the zonotopes are beyond paralleltopes, the complexity of (45) grows. Note that the complexity is quadratic in horizon T, instead of linear that is typical in traditional MPC.

5.6 Proxy Constraint via Zonotope Order Reduction

In order to reduce complexity, we propose the following method that preserves correctness, but may degrade optimality. A very effective way to decrease complexity is performing zonotope order reduction for Ξ_t prior to solving (45). Let $\Xi_t^{\text{Redu.}}$ be the reduced zonotope of Ξ_t. However, the performance variable is not a function of this zonotope. By treating ξ_t and f_t being independent (which are not) in (40), we conservatively have the following approximation:

$$\mathbb{Z}_t \subseteq R_t\tilde{Y}_t + S_t\tilde{U}_t + (R_t\Phi_t + S_t\Theta_t)\Xi_t^{\text{Redu.}} \oplus \mathbb{F}_t, \tag{47}$$

Instead of using $\mathbb{Z}_t$, we use the right hand side in (47) to alleviate complexity. The main advantage of such a order reductions is that the dependency of the complexity on n - the original state space dimension - can be removed as the number of the columns of the generator in $\Xi_t^{\text{Redu.}}$ can be as small as $o(t+1)$ (order 1 zonotope). We always use this technique when synthesizing policies for systems with very large state-spaces in a scalable way.

6 INFINITE-TIME INVARIANCE

In this section, we provide a solution to Problem 2. We assume the system to be time invariant. The method is based on the following idea: if a finite-time policy after time T_∞ manages to bring all the possible state values x_{T_∞} to fully lie in X_0, then we can repeat the same policy starting from T_∞ and so on. Thus, we obtain a stabilizing periodic output feedback policy that maps at most T_∞ recent

history to a control input. We formalize this in the following theorem.

Theorem 2 (Solution to Problem 2). *If there exists T_∞ such that $X_{T_\infty} \subseteq X_0$, where:*

$$\begin{aligned} X_t = \quad & P^u_{T_\infty} \tilde{U}_{t-1} + (P^x_t + P^u_{T_\infty} \Theta_{t-1} \Xi^x_{t-1}) X_0 \\ & \oplus (P^w_t + P^u_t \Theta_{t-1} \Xi^w_{t-1}) \bigotimes_{\tau=0}^{t-1} W_\tau \oplus \\ & P^v_t \Theta_{t-1} \Xi^v_t \bigotimes_{\tau=0}^{t-1} V_\tau \end{aligned} \tag{48}$$

and $X_t \in \mathbb{H}_x, \mathbb{U}_t \in \mathbb{U}_t, t \in \mathbb{N}_{\leq T_\infty}$, then the following control policy is a solution to Problem 2:

$$\pi(t) = \pi(t \mod T_\infty) \tag{49}$$

Proof. We plug the control policy (8) in (13) to find the state-response. When $X_{T_\infty} \subseteq X_0$, then the history can be cleaned with the reset of time to 0 and the repetition of the same policy. It is straightforward to observe that $X_{kT_\infty} \subseteq X_{(k-1)T_\infty} X_0 \subset X_0 \forall k \in \mathbb{N}, t \geq 1$. □

Our solution based on Theorem 2 is sound but it is not as flexible as state-feedback methods such as in [30]. The main reason is the dependence of the solutions on X_0. If X_0 is a decision variable, then the containment encoding leads to bilinear constraints.

7 EXAMPLES

We provide multiple examples to demonstrate the usefulness of our methods. First, we consider unconstrained problems and compare our controller to LQG. Next, we consider constrained problems. The scripts for all the examples are publicly available [1].

7.1 Comparison against LQG

Our method is different in nature from LQG, as it is mainly designed to handle constraints and provide worst-case guarantees rather than average performance. Nevertheless, both methods can be applied to unconstrained problems problem. It is anticipated that for some cases our method achieves smaller cost than LQG, which is mainly due to the fact that LQG is optimized for Gaussian noise, but ours is indifferent to the distribution. In both examples the systems are LTI.

Example 1. *We chose $n = 6, m = 1, o = 1, z = 1$, yielding a problem of the same size as in [15] and [24]. We randomly generate matrix A. The location of its eigenvalues in the complex plane are shown in Figure 2[Top]. It is observed that the 4 of the eigenvalues are outside of the unit circle, implying open-loop instability. The systems in [24] were, on the other hand, open-loop stable. We set $B = (0, 1, 1, 1, 1, 0)$. Only the first entry of matrices C, D are 1 and the rest are zero (we only measure and track the first state). The zonotope of $X_0 = \langle 0, I_6 \rangle$ is centered at origin and its generator is I_6, meaning the initial state can take any value in the unit hyperbox. We set $\mathbb{W}_t = \langle 0, 0.01 I_6 \rangle, \mathbb{V}_t = \langle 0, 0.01 \rangle, t = 0, 1, \cdots, T-1$. The horizon is set to $T = 40$. With no additional constraints, we synthesize our output feedback control policy using (45). Comparisons are made against time-varying LQG (TV-LQG), which uses Riccati difference equation to update control and observer gains over time [2].*

[1] https://github.com/sadraddini/polytrajectory

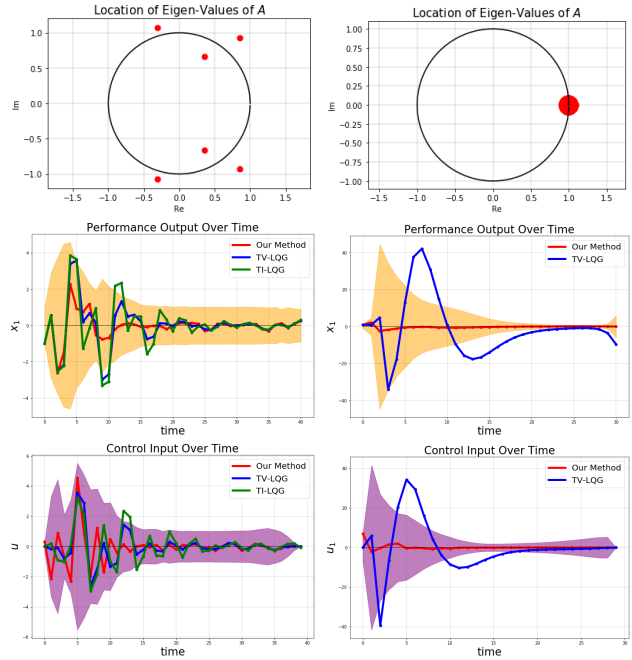

Figure 2: [Left] Example 1 and [Right] Example 2. The location of eigenvalues of A (top figures), sample trajectories, and reachable sets shown by the shaded areas, which are obtained from projecting the associated zonotopes.

The time-invariant LQG (TI-LQG) uses discrete-time algebraic Riccati equation (DARE) to find fixed gains. To excite the system the most, we randomly sample the measurement noise and process disturbance from the vertices of their respective zonotopes. The LQG gains are computed from the covariance matrices fit to 1000 randomly generated samples. The cost function is the 2-norm summation of inputs and outputs. However, in using (45), we replace the cost with ∞-norm-squared and use zonotope order-reduction (to make order zonotope orders equal to 1) to improve tractability. The synthesis optimization problem (45) takes 0.4 seconds on a personal computer. A sample trajectory is shown in Figure 2. The reachable sets corresponding to $\mathbb{Z}_t$ and $\mathbb{U}_t$ are shown in shaded colors. On over 1000 runs, the cost associated with our method was about 20% less than the one corresponding to TV-LQG, and 28% less than the cost of TI-LQG. When we changed the disturbance and noise distribution to Gaussian, we found that over 1000 runs, the cost of TV-LQG is on average about 4% smaller than our method, which has roughly the same average cost as TI-LQG. Note that LQG is provably optimal on average for Gaussian setting [2].

Example 2. *We chose $n = 1000, m = 5, o = 1, z = 1$. Solving DARE for this problem size is numerically difficult. Therefore, we only solve the problem using our method and TV-LQG. The location of eigenvalues and a sample trajectory are shown in Figure 2. The rest of the setup is similar to Example 1. In this numerical example and in many other cases that only a small portion of state values are measured, LQG becomes very sensitive with respect to numerics in covariance matrices. In this example, our method overperforms LQG.*

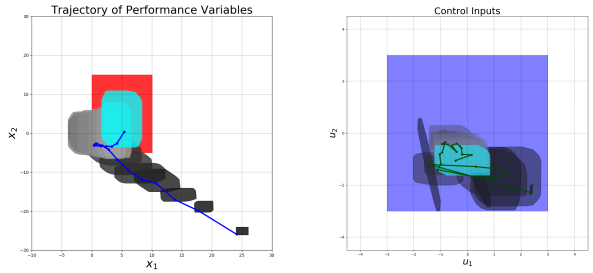

Figure 3: Example 3: [Left] Sample trajectory in $x_{[1]}, x_{[2]}$ and the zonotopes $\mathbb{Z}_t$ through time. The red shaded area is $\mathbb{Z}_T$. [Right] The corresponding trajectory of inputs and the zonotopes of $\mathbb{U}[t]$ through time. The shaded box is the set of valid inputs. In both plots, time progression is depicted by brighter colors. The cyan zonotopes are for final time.

The noise variables are spanned over ± 0.01 in all dimensions and the initial state belongs to unit hyperbox in $\mathbb{R}^{1000}$.

7.2 Constrained Control

Example 3. *We pick $n = 10, m = 2, o = 1, z = 2$ and randomly generated matrices in (7). We only observe $x_{[1]}$, and the performance variables are $x_{[1]}$ and $x_{[2]}$. We constrain the control input for all times in a box, and the terminal constraint is a box in $x_{[1]}, x_{[2]}$ space. Process and measurement noise are present - their numerical descriptions are available in the scripts. A sample trajectory with $T = 20$ and results are shown in Fig. 3.*

7.3 Control with Fixed but Unknown Parameters

We may consider a system where the state includes static parameters of the environment. These parameters are not observable at all times, but they are measured in a time-varying way. The parameters take values from a polytopic set, and our synthesis method provides a formal certificate of robustness versus all admissible parameters.

Example 4. *Consider a particle moving in 2D space with constant velocity in the horizontal direction and double integrator dynamics with acceleration control in the vertical direction, where gravity faces downward. The environment includes obstacles on the upper side and lower side that remain flat for K segments of length 1 (see Fig. 4. The heights of the obstacles belong to a given zonotope. The state is $2K+3$ dimensions ($2K$ static states for the heights of the obstacles, 3 for the particle position and velocity). At every time, the measured outputs are the difference between the height of the particle and the obstacle in the current and the subsequent segment (4 outputs). We also include small measurement noise and disturbance acting in vertical position. The performance variables encode the distance from the obstacles in each segment. This yields a system with time-varying C and D matrix.*

By solving the convex program in Theorem 45, we obtain a policy that is provably robust against all possible environment parameters.

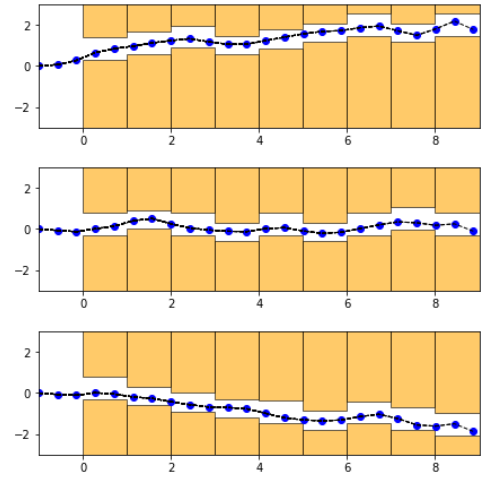

Figure 4: Example 4: The obstacles are shaded in orange. The particle moves to the right.

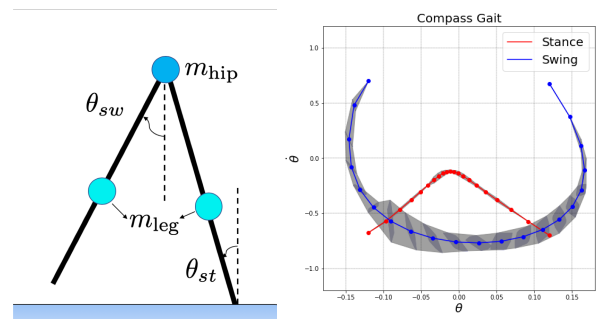

Figure 5: Example 5: Infinite-time invariance with compass gait model. The measured output is the angle between two legs, which is $\theta_{st} + \theta_{sw}$. The areas between zonotopes are filled in a continous-time sense by taking the convex-hull of two time-consecutive zonotopes.

Three samples of such environments and corresponding sample trajectories are shown in Fig. 4, where $K = 10$.

7.4 Periodic Invariance

Example 5. *We consider the compass gait walking model in [11]. By assuming small angles, we characterize the system with a jump linear model, and discretize time with $\Delta t = 0.05s$. Note that limit cycles still exist because of discrete jumps. The parameters are the same as in [5]. The only measurable output is the angle between the stance and the swing leg. The task is obtaining a polytopic trajectory driven with output feedback that repeats itself after the reset map. A slight modification to Theorem 2 is necessary. By considering a linear reset map R, instead of writing $X_{T_\infty} \subseteq X_0$, we have $RX_{T_\infty} \subseteq X_0$, which is still a linear zonotope containment problem. The results are shown in Fig. 5. The results are comparable to the state-feedback LQR region-of-attraction computed in [25] with the note that linearization errors are not considered here.*

8 CONCLUSION AND FUTURE WORK

In this paper, we provided provably correct output feedback policies with explicit zonotope representation of reachable sets. The controllers were found using convex programs.

Future work will focus on the direction suggested in Remark 1, and extension to hybrid systems where unlike example 5, the mode sequence is not known in prior.

9 ACKNOWLEDGEMENTS

This work was partially supported by Department of the Navy, Office of Naval Research (ONR) Award No.: N00014-17-1-2699.

REFERENCES

[1] James Anderson, John C Doyle, Steven H Low, and Nikolai Matni. 2019. System level synthesis. *Annual Reviews in Control* (2019).

[2] Karl J Åström. 2012. *Introduction to stochastic control theory.* Courier Corporation.

[3] Alberto Bemporad, Francesco Borrelli, and Manfred Morari. 2003. Min-max control of constrained uncertain discrete-time linear systems. *IEEE Transactions on automatic control* 48, 9 (2003), 1600–1606.

[4] Stephen Boyd, Laurent El Ghaoui, Eric Feron, and Venkataramanan Balakrishnan. 1994. *Linear matrix inequalities in system and control theory.* Vol. 15. Siam.

[5] Katie Byl and Russ Tedrake. 2008. Approximate optimal control of the compass gait on rough terrain. In *2008 IEEE International Conference on Robotics and Automation.* IEEE, 1258–1263.

[6] Jeremy Coulson, John Lygeros, and Florian Dörfler. 2019. Data-enabled predictive control: in the shallows of the DeePC. In *2019 18th European Control Conference (ECC).* IEEE, 307–312.

[7] John C Doyle. 1978. Guaranteed margins for LQG regulators. *IEEE Transactions on automatic Control* 23, 4 (1978), 756–757.

[8] Stevan Dubljevic, Nael H El-Farra, Prashant Mhaskar, and Panagiotis D Christofides. 2006. Predictive control of parabolic PDEs with state and control constraints. *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal* 16, 16 (2006), 749–772.

[9] Marcello Farina, Luca Giulioni, Lalo Magni, and Riccardo Scattolini. 2015. An approach to output-feedback MPC of stochastic linear discrete-time systems. *Automatica* 55 (2015), 140–149.

[10] Antoine Girard. 2005. Reachability of uncertain linear systems using zonotopes. In *International Workshop on Hybrid Systems: Computation and Control.* Springer, 291–305.

[11] Ambarish Goswami, Bernard Espiau, and Ahmed Keramane. 1997. Limit cycles in a passive compass gait biped and passivity-mimicking control laws. *Autonomous Robots* 4, 3 (1997), 273–286.

[12] Paul James Goulart. 2007. *Affine feedback policies for robust control with constraints.* Ph.D. Dissertation. University of Cambridge.

[13] Paul J Goulart, Eric C Kerrigan, and Jan M Maciejowski. 2006. Optimization over state feedback policies for robust control with constraints. *Automatica* 42, 4 (2006), 523–533.

[14] Svein Hovland and Jan Tommy Gravdahl. 2008. Complexity reduction in explicit MPC through model reduction. *IFAC Proceedings Volumes* 41, 2 (2008), 7711–7716.

[15] Svein Hovland, C Lovaas, Jan Tommy Gravdahl, and Graham C Goodwin. 2008. Stability of model predictive control based on reduced-order models. In *2008 47th IEEE Conference on Decision and Control.* IEEE, 4067–4072.

[16] Linbin Huang, Jeremy Coulson, John Lygeros, and Florian Dörfler. 2019. Data-Driven Wide-Area Control. arXiv:eess.SY/1911.12151

[17] David Hyland and Dennis Bernstein. 1984. The optimal projection equations for fixed-order dynamic compensation. *IEEE Trans. Automat. Control* 29, 11 (1984), 1034–1037.

[18] D Peter Joseph and T Julius Tou. 1961. On linear control theory. *Transactions of the American Institute of Electrical Engineers, Part II: Applications and Industry* 80, 4 (1961), 193–196.

[19] Kai Kellner, Thorsten Theobald, and Christian Trabandt. 2013. Containment problems for polytopes and spectrahedra. *SIAM Journal on Optimization* 23, 2 (2013), 1000–1020.

[20] Markus Kögel and Rolf Findeisen. 2015. Robust output feedback model predictive control using reduced order models. *IFAC-PapersOnLine* 48, 8 (2015), 1008–1014.

[21] Anna-Kathrin Kopetzki, Bastian Schürmann, and Matthias Althoff. 2017. Methods for order reduction of zonotopes. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC).* IEEE, 5626–5633.

[22] Martin Loehning, Marcus Reble, Jan Hasenauer, Shuyou Yu, and Frank Allgoewer. 2014. Model predictive control using reduced order models: Guaranteed stability for constrained linear systems. *Journal of Process Control* 24, 11 (2014), 1647–1659.

[23] Johan Löfberg. 2003. *Minimax approaches to robust model predictive control.* Vol. 812. Linköping University Electronic Press.

[24] Joseph Lorenzetti, Benoit Landry, Sumeet Singh, and Marco Pavone. 2018. Reduced Order Model Predictive Control For Setpoint Tracking. *arXiv preprint arXiv:1811.06590* (2018).

[25] Ian R Manchester, Mark M Tobenkin, Michael Levashov, and Russ Tedrake. 2011. Regions of attraction for hybrid limit cycles of walking robots. *IFAC Proceedings Volumes* 44, 1 (2011), 5801–5806.

[26] Bengt Mårtensson. 1985. The order of any stabilizing regulator is sufficient a priori information for adaptive stabilization. *Systems & Control Letters* 6, 2 (1985), 87–91.

[27] David Q Mayne, SV Raković, Rolf Findeisen, and Frank Allgöwer. 2006. Robust output feedback model predictive control of constrained linear systems. *Automatica* 42, 7 (2006), 1217–1222.

[28] Diogo Narciso and Efstratios Pistikopoulos. 2008. A combined balanced truncation and multi-parametric programming approach for linear model predictive control. In *Computer Aided Chemical Engineering*. Vol. 25. Elsevier, 405–410.

[29] Frauke Oldewurtel, Colin N Jones, and Manfred Morari. 2008. A tractable approximation of chance constrained stochastic MPC based on affine disturbance feedback. In *2008 47th IEEE Conference on Decision and Control*. IEEE, 4731–4736.

[30] SV Raković, Eric C Kerrigan, David Q Mayne, and Konstantinos I Kouramas. 2007. Optimized robust control invariance for linear discrete-time systems: Theoretical foundations. *Automatica* 43, 5 (2007), 831–841.

[31] Sadra Sadraddini and Russ Tedrake. 2019. Linear Encodings for Polytope Containment Problems. *arXiv preprint arXiv:1903.05214* (2019).

[32] Michael G Safonov and RY Chiang. 1989. A Schur method for balanced-truncation model reduction. *IEEE Trans. Automat. Control* 34, 7 (1989), 729–733.

[33] Joelle Skaf and Stephen P Boyd. 2010. Design of affine controllers via convex optimization. *IEEE Trans. Automat. Control* 55, 11 (2010), 2476–2487.

[34] Pantelis Sopasakis, Daniele Bernardini, and Alberto Bemporad. 2013. Constrained model predictive control based on reduced-order models. In *52nd IEEE Conference on Decision and Control*. IEEE, 7071–7076.

[35] Sankaranarayanan Subramanian, Sergio Lucia, and Sebastian Engell. 2017. A novel tube-based output feedback MPC for constrained linear systems. In *2017 American Control Conference (ACC)*. IEEE, 3060–3065.

[36] Vassilis L Syrmos, Chaouki T Abdallah, Peter Dorato, and Karolos Grigoriadis. 1997. Static output feedback—a survey. *Automatica* 33, 2 (1997), 125–137.

[37] LG Van Willigenburg and Willem L De Koning. 1999. Optimal reduced-order compensation of time-varying discrete-time systems with deterministic and white parameters. *Automatica* 35, 1 (1999), 129–138.

[38] LG Van Willigenburg and Willem L De Koning. 2000. Numerical algorithms and issues concerning the discrete-time optimal projection equations. *European Journal of Control* 6, 1 (2000), 93–110.

[39] Zhaoyang Wan and Mayuresh V Kothare. 2004. Efficient scheduled stabilizing output feedback model predictive control for constrained nonlinear systems. *IEEE Trans. Automat. Control* 49, 7 (2004), 1172–1177.

[40] Dante Youla, Hamid Jabr, and Jr Bongiorno. 1976. Modern Wiener-Hopf design of optimal controllers–Part II: The multivariable case. *IEEE Trans. Automat. Control* 21, 3 (1976), 319–338.

[41] Dragan Žigić, Layne T Watson, Emmanuel G Collins Jr, and Dennis S Bernstein. 1992. Homotopy methods for solving the optimal projection equations for the H 2 reduced order model problem. *International Journal of control* 56, 1 (1992), 173–191.