Stability and Performance Verification of Optimization-based Controllers

Milan Korda and Colin N. Jones*

January 15, 2015

Abstract

This paper presents a method to verify closed-loop properties of optimization-based controllers for deterministic and stochastic constrained polynomial discrete-time dynamical systems. The closed-loop properties amenable to the proposed technique include global and local stability, performance with respect to a given cost function (both in a deterministic and stochastic setting) and the $\mathcal{L}_2$ gain. The method applies to a wide range of practical control problems: For instance, a dynamical controller (e.g., a PID) plus input saturation, model predictive control with state estimation, inexact model and soft constraints, or a general optimization-based controller where the underlying problem is solved with a fixed number of iterations of a first-order method are all amenable to the proposed approach.

The approach is based on the observation that the control input generated by an optimization-based controller satisfies the associated Karush-Kuhn-Tucker (KKT) conditions which, provided all data is polynomial, are a system of polynomial equalities and inequalities. The closed-loop properties can then be analyzed using sum-of-squares (SOS) programming.

1 Introduction

This paper presents a computational approach to analyze closed-loop properties of optimization-based controllers for constrained polynomial discrete-time dynamical systems. Throughout the paper we assume that we are given an optimization-based controller that at each time instance generates a control input by solving an optimization problem parametrized by a function of the past measurements of

*Milan Korda and Colin N. Jones are with the Laboratoire d'Automatique , École Polytechnique Fédérale de Lausanne, Switzerland. {milan.korda, colin.jones}@epfl.ch.

the controlled system's output, and we ask about closed-loop properties of this interconnection. This setting encompasses a wide range of control problems including the control of a polynomial dynamical system by a linear controller (e.g., a PID) with an input saturation, output feedback model predictive control with inexact model and soft constraints, or a general optimization-based controller where the underlying problem is solved approximately with a fixed number of iterations of a first-order[1] optimization method. Importantly, the method verifies all KKT points; hence it can be used to verify closed-loop properties of optimization-based controllers where the underlying, possibly nonconvex, optimization problem is solved with a local method with guaranteed convergence to a KKT point only.

The closed-loop properties possible to analyze by the approach include: global and local stability, performance with respect to a discounted infinite-horizon cost (where we provide polynomial upper and lower bounds on the cost attained by the controller over a given set of initial conditions, both in a deterministic and a stochastic setting), the $\mathcal{L}_2$ gain from a given disturbance input to a given performance output (where we provide a numerical upper bound).

The main idea behind the presented approach is the observation that the KKT system associated to an optimization problem with polynomial data is a system of polynomial equalities and inequalities. Consequently, provided that suitable constraint qualification conditions hold (see, e.g., [13]), the solution of this optimization problem satisfies a system of polynomial equalities and inequalities. Hence, the closed-loop evolution of a polynomial dynamical system controlled by an optimization-based controller solving at each time step an optimization problem with polynomial data can be seen as a difference inclusion where the successor state lies in a set defined by polynomial equalities and inequalities. This difference inclusion is then analyzed using sum-of-squares (SOS) techniques (see, e.g., [6, 11] for introduction to SOS programming).

The approach is based on the observation of Primbs [15] who noticed that the KKT system of a constrained linear-quadratic optimization problem is a set of polynomial equalities and inequalities and used the S-procedure (see, e.g., [1]) to derive sufficient linear matrix inequality (LMI) conditions for a given linear MPC controller to be stabilizing. In this work we significantly extend the approach in terms of both the range of closed-loop properties analyzed and the range of practical problems amenable to the method. Indeed, our approach is applicable to general polynomial dynamical systems, both deterministic and stochastic, and allows the analysis not only of stability but also of various performance measures. The approach is not only applicable to an MPC controller with linear dynamics and a quadratic cost function as in [15] but also to a general optimization-based controller, where the optimization

[1] By a first order optimization method we mean a method using only function value and gradient information, e.g., the projected gradient method (see Section 4.4).

problem may not be solved exactly, encompassing all the above-mentioned control problems.

This work is a continuation of [5] where the approach was used to analyze the stability of optimization-based controllers where the optimization problem is solved approximately by a fixed number of iterations of a first order method. The results of [5] are summarized in Section 4.4 of this paper as one of the examples that fit in the presented framework.

The paper is organized as follows. Section 2 gives a brief introduction to SOS programming. Section 3 states the problem to be solved. Section 4 presents a number of examples amenable to the proposed method. Section 5 presents the main verification results: Section 5.1 on global stability, Section 5.2 on local stability, Section 5.3 on performance analysis in a deterministic setting, Section 5.4 on performance analysis in a stochastic setting and Section 5.5 on the analysis of the $\mathcal{L}_2$ gain in a robust setting. Computational aspects are discussed in Section 6. Numerical examples are in Section 7 and some proofs are collected in the Appendix.

2 Sum-of-squares programming

Throughout the paper we will rely on sum-of-squares (SOS) programming, which allows us to optimize, in a convex way, over polynomials with nonnegativity constraints imposed over a set defined by polynomial equalities and inequalities (see, e.g., [11, 6] for more details on SOS programming). In particular we will often encounter optimization problems with constraints of the form

$$\mathcal{L}V(x) \geq 0 \quad \forall\, x \in \mathbf{K}, \tag{1}$$

where $\mathcal{L}$ is a linear operator (e.g., a simple addition or a composition with a fixed function), $V : \mathbb{R}^n \to \mathbb{R}$ a polynomial and

$$\mathbf{K} = \{x \in \mathbb{R}^n \mid g(x) \geq 0,\ h(x) = 0\},$$

where the functions $g : \mathbb{R}^n \to \mathbb{R}^{n_g}$ and $h : \mathbb{R}^n \to \mathbb{R}^{n_h}$ are vector polynomials (i.e., each component is a polynomial). A *sufficient* condition for (1) to be satisfied is

$$\mathcal{L}V = \sigma_0 + g^\top \sigma + h^\top p = \sigma_0 + \sum_{i=1}^{n_g} \sigma_i g_i + \sum_{i=1}^{n_h} p_i h_i, \tag{2}$$

where σ_0 and σ_i, $i = 1, \ldots, n_g$, are SOS polynomials and p_i, $i = 1, \ldots, n_h$, are arbitrary polynomials. A polynomial σ is SOS if it can be written as

$$\sigma(x) = \beta(x)^\top \mathcal{Q} \beta(x), \quad \mathcal{Q} \succeq 0, \tag{3}$$

where $\beta(x)$ is a vector of polynomials and $\mathcal{Q} \succeq 0$ signifies that $\mathcal{Q}$ is a positive semidefinite matrix. The condition (3) trivially implies that $\sigma(x) \geq 0$ for all $x \in \mathbb{R}^n$. Importantly, the condition (3) translates to a set of linear constraints and a positive semidefiniteness constraint and therefore is equivalent to a semidefinite programming (SDP) feasibility problem. In addition, the constraint (2) is affine in the coefficients of V, σ and p; therefore (2) also translates to an SDP feasibility problem and, crucially, it is possible to *optimize* over the coefficients of V (as long as they are affinely parametrized in the decision variables) subject to the constraint (2) using semidefinite programming.

In the rest of the paper when we say that a constraint of the form (1) is replaced by sufficient SOS constraints, then we mean that (1) is replaced with (2).

In addition we will often encounter optimization problems with objective functions of the form

$$\min/\max \int_{\mathbf{X}} V(x)\, dx, \tag{4}$$

where V is a polynomial and $\mathbf{X}$ a simple set (e.g., a box). The objective function is linear in the coefficients of the polynomials V. Indeed, expressing $V(x) = \sum_{i=1}^{n_\beta} v_i \beta_i(x)$, where $(\beta_i)_{i=1}^{n_\beta}$ are fixed polynomial basis functions and $(v_i)_{i=1}^{n_\beta}$ the corresponding coefficients, we have

$$\int_{\mathbf{X}} V(x)\, dx = \sum_{i=1}^{n_\beta} v_i \int_{\mathbf{X}} \beta_i(x)\, dx = \sum_{i=1}^{n_\beta} v_i m_i,$$

where the moments $m_i := \int_{\mathbf{X}} \beta_i(x)\, dx$ can be precomputed (in a closed form for simple sets $\mathbf{X}$). We see that the objective (4) is linear in the coefficients $(v_i)_{i=1}^{n_\beta}$ and hence optimization problems with objective (4) subject to the constraint (1) enforced via the sufficient constraint (2) translate to an SDP.

3 Problem statement

We consider the nonlinear discrete-time dynamical system

$$x^+ = f_x(x, u), \tag{5a}$$

$$y = f_y(x), \tag{5b}$$

where $x \in \mathbb{R}^{n_x}$ is the state, $u \in \mathbb{R}^{n_u}$ the control input, $x^+ \in \mathbb{R}^{n_x}$ the successor state, $f_x : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ a transition mapping and $f_y : \mathbb{R}^{n_x} \to \mathbb{R}^{n_y}$ an output mapping. We assume that each component of f_x and f_y is a multivariate polynomial in (x, u) and x, respectively.

We assume that the system is controlled by a given set-valued controller

$$u \in \kappa(\mathbf{K}_s), \tag{6}$$

where $\kappa : \mathbb{R}^{n_\theta} \to \mathbb{R}^{n_u}$ is polynomial and

$$\mathbf{K}_s := \{\theta \in \mathbb{R}^{n_\theta} \,|\, \exists \lambda \in \mathbb{R}^{n_\lambda} \text{ s.t. } g(s,\theta,\lambda) \geq 0, h(s,\theta,\lambda) = 0\}, \tag{7}$$

where each component of the vector-valued functions $g : \mathbb{R}^{n_s} \times \mathbb{R}^{n_\theta} \times \mathbb{R}^{n_\lambda} \to \mathbb{R}^{n_g}$ and $h : \mathbb{R}^{n_s} \times \mathbb{R}^{n_\theta} \times \mathbb{R}^{n_\lambda} \to \mathbb{R}^{n_h}$ is a polynomial in the variables (s, θ, λ). The set $\mathbf{K}_s$ is parametrized by the output of a dynamical system

$$z^+ = f_z(z, y), \tag{8a}$$
$$s = f_s(z, y), \tag{8b}$$

where $f_z : \mathbb{R}^{n_z} \times \mathbb{R}^{n_y} \to \mathbb{R}^{n_z}$ and $f_s : \mathbb{R}^{n_z} \times \mathbb{R}^{n_y} \to \mathbb{R}^{n_s}$ are polynomial. The problem setup is depicted in Figure 1. In the rest of the paper we develop a method to analyze the closed-loop stability and performance of this interconnection. Before doing that we present several examples which fall into the presented framework.

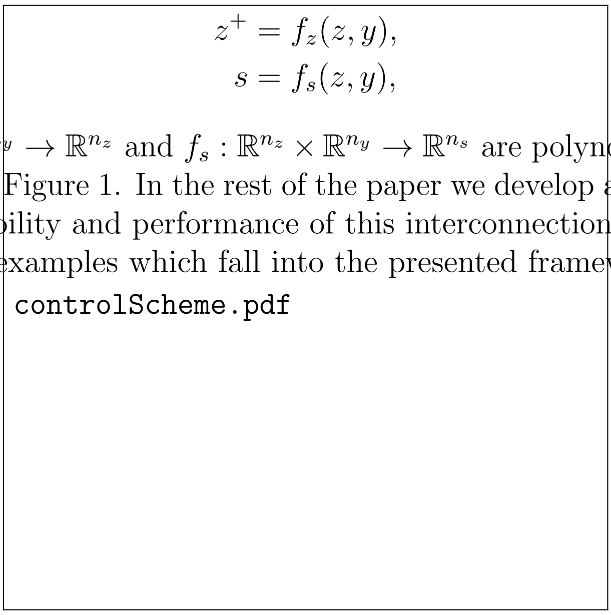

Figure 1: Control scheme

4 Examples

The framework considered allows for the analysis of a large number of practical control scenarios. The common idea is to write the control input u as the output of an optimization problem with polynomial data parametrized by the state of the dynamical system (8). The control input u then belongs to the associated KKT system (provided that mild regularity conditions are satisfied) which is of the form (7).

4.1 Polynomial dynamical controller + input saturation

Any polynomial dynamical controller (e.g., a PID controller) plus an input saturation can be written in the presented form provided that the input constraint set is defined by finitely many polynomial inequalities satisfying mild constraint qualification conditions (see, e.g., [13]). Indeed, regarding z as the state and s as the output of the controller and generating u according to $u = \mathrm{proj}_U(s)$, where

$$\mathrm{proj}_U(s) = \min_{\theta \in U} \frac{1}{2} \|\theta - s\|_2^2 \tag{9}$$

is the Euclidean projection of s on the constraint set U. Assuming that the input constraint set is of the form

$$U = \{v \in \mathbb{R}^{n_u} \mid g_U(v) \geq 0\}, \tag{10}$$

where $g_U : \mathbb{R}^{n_u} \to \mathbb{R}^{n_{g_U}}$ has polynomial entries, the KKT conditions associated to the optimization problem (9) read

$$\begin{aligned} \theta - s - \nabla g_U(\theta)\lambda &= 0 && \text{(11a)} \\ \lambda^\top \theta &= 0 && \text{(11b)} \\ \lambda &\geq 0 && \text{(11c)} \\ g_U(\theta) &\geq 0, && \text{(11d)} \end{aligned}$$

where $\lambda \in \mathbb{R}^{n_U}$ is the vector of Lagrange multipliers associated with the constraints defining U and ∇g_U is the transpose of the Jacobian of g_U (i.e., $[\nabla g_U]_{i,j} = \frac{\partial [g_U]_j}{\partial x_i}$). Assuming that constraint qualification conditions hold such that any minimizer of (9) satisfies the KKT conditions (11) we conclude that

$$u = \mathrm{proj}_U(s) \in \kappa(\mathbf{K}_s)$$

with κ being the identity (i.e., $\kappa(\theta) = \theta$),

$$h(s, \theta, \lambda) = \begin{bmatrix} \theta - s - \nabla g_U(\theta)\lambda \\ \lambda^\top \theta \end{bmatrix}$$

and

$$g(s, \theta, \lambda) = \begin{bmatrix} \lambda \\ g_U(\theta) \end{bmatrix},$$

where h and g are polynomials in (s, θ, λ) as required.

Note that the description of the input constraint set (10) is not unique. For example, if the input constraint set is $[-1, 1]$, then the function g_U can be

$$g_U(\theta) = (1 - \theta)(1 + \theta) \tag{12}$$

or

$$g_U(\theta) = \begin{bmatrix} 1 - \theta \\ 1 + \theta \end{bmatrix} \tag{13}$$

or any odd powers of the above. Depending on the particular description, the constraint qualification conditions may or may not hold. It is therefore important to choose a suitable description of U which is both simple and such that the constraint qualification conditions hold. We remark that in the case of $U = [-1, 1]$ both (12) and (13) satisfy these requirements.

Note also that the KKT system (11) may be satisfied by points which are not global minimizers of (9) if the set U is nonconvex; this is an artefact of the presented method and cannot be avoided within the presented framework. We note, however, that the input constraint set U is in most practical cases convex (note that the concavity of the components of g_U is not required; what matters is the convexity of the set U defined by g_U [7]).

4.2 Output feedback nonlinear MPC with model mismatch and soft constraints

This example shows how to model within the presented framework a nonlinear MPC controller with state estimation, a model mismatch[2], soft constraints and no a priori stability guarantees (enforced, e.g., using a terminal penalty and/or terminal set) and possibly only locally optimal solutions delivered by the optimization algorithm. In this case the system (8) is an estimator of the state of the dynamical system (5) and in each time step the following optimization problem is solved

$$\begin{array}{ll} \underset{\hat{\boldsymbol{u}},\hat{\boldsymbol{x}},\boldsymbol{\varepsilon}}{\text{minimize}} & l_{\mathrm{s}}(\boldsymbol{\varepsilon}) + \sum_{i=0}^{N} l_i(s, \hat{x}_i, \hat{u}_i) \\ \text{subject to} & \hat{x}_{i+1} = \hat{f}(\hat{x}_i, \hat{u}_i), \quad i = 0, \ldots, N-1 \\ & a(s, \hat{\boldsymbol{x}}, \hat{\boldsymbol{u}}) + \boldsymbol{\varepsilon} \geq 0 \\ & b(s, \hat{\boldsymbol{x}}, \hat{\boldsymbol{u}}) = 0, \end{array} \tag{14}$$

where $\hat{\boldsymbol{x}} = (\hat{x}_0, \ldots, \hat{x}_N)$, $\hat{\boldsymbol{u}} = (\hat{u}_0, \ldots, \hat{u}_{N-1})$, $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_{n_\varepsilon})$ are slack variables for the inequality constraints, $\hat{f}$ is a polynomial model of the true transition mapping f_x, l_{s} is a polynomial penalty for violations of the inequality constraints, l_i, $i = 0, \ldots, N$, are polynomial stage costs and a and b (vector) polynomial constraints parametrized by the state estimate s produced by (8). If the dimension of the state estimate s is equal to the dimension of the state of the model $\hat{x}$ (which we do not require), then most MPC formulations will impose $\hat{x}_0 = s$, which is encoded by making one of the components of b equal to $\hat{x}_0 - s$. The formulation (14) is, however, not restricted to

[2] In this example we assume that we know the true model of the system but in the MPC controller we intentionally use a different model (e.g., we use a linearized or otherwise simplified model for the sake of computation speed); the true model is used only to *verify* closed-loop properties of the true model controlled by the MPC controller. See Section 5.5 for the case where the true model is not known exactly even for the verification purposes and the model mismatch is captured by an exogenous disturbance.

this scenario and allows arbitrary dependence of the constraints (along the whole prediction horizon) on the state estimate s. The control input applied to the system is then $u = \hat{u}_0^\star$, where $\hat{u}_0^\star$ is the first component of any vector $\hat{\boldsymbol{u}}^\star$ optimal in (14).

The KKT system associated to (14) reads

$$\begin{aligned} \nabla_{\hat{\boldsymbol{x}},\hat{\boldsymbol{u}},\boldsymbol{\varepsilon}} \mathcal{L}(s,\hat{x},\hat{u},\boldsymbol{\varepsilon},\lambda) &= 0 && \text{(15a)}\\ \lambda_a^\top a(s,\hat{\boldsymbol{x}},\hat{\boldsymbol{u}}) &= 0 && \text{(15b)}\\ b(s,\hat{\boldsymbol{x}},\hat{\boldsymbol{u}}) &= 0 && \text{(15c)}\\ \lambda_a &\geq 0 && \text{(15d)}\\ a(s,\hat{\boldsymbol{x}},\hat{\boldsymbol{u}}) &\geq 0, && \text{(15e)} \end{aligned}$$

where $\lambda := (\lambda_a, \lambda_b, \lambda_{\hat{f}}^0, \ldots, \lambda_{\hat{f}}^{N-1})$ and

$$\begin{aligned} \mathcal{L}(s,\hat{x},\hat{u},\boldsymbol{\varepsilon},\lambda) := l_{\mathrm{s}}(\boldsymbol{\varepsilon}) + \sum_{i=0}^{N} l_i(s,\hat{x}_i,\hat{u}_i) - \lambda_a^\top(a(s,\hat{\boldsymbol{x}},\hat{\boldsymbol{u}}) + \boldsymbol{\varepsilon})\\ + \lambda_b^\top b(s,\hat{\boldsymbol{x}},\hat{\boldsymbol{u}}) + \sum_{i=0}^{N-1} \lambda_{\hat{f}}^i (\hat{x}_{i+1} - \hat{f}(\hat{x}_i,\hat{u}_i)) \end{aligned}$$

is the Lagrangian of (14). The KKT system (15) is a system of polynomial equalities and inequalities. Consequently, setting

$$\theta := (\hat{\boldsymbol{u}}, \hat{\boldsymbol{x}}, \boldsymbol{\varepsilon}), \quad \kappa(\theta) = \hat{u}_0$$

and assuming that constraint qualification conditions hold such that every optimal solution to (14) satisfies the KKT condition (15), there exist polynomial functions $h(s,\theta,\lambda)$ and $g(s,\theta,\lambda)$ such that $\hat{u}_0^\star \in \kappa(\mathbf{K}_s)$ for every $\hat{u}_0^\star$ optimal in (14).

Finally, let us mention that, provided suitable constraint qualification conditions hold, not only every globally optimal $\hat{u}_0^\star$ will satisfy the KKT system but also every *locally optimal* solution to (14) and every critical point of (14) will; hence the proposed method can be used to verify stability and performance properties even if only local solutions to (14) are delivered by the optimization algorithm.

Remark 1. *Note that the situation where the optimization problem (14) is not solved exactly can be handled as well. One way to do so is to include an auxiliary variable δ capturing the inaccuracy in the solution, either in the satisfaction of the KKT system (15) or directly as an error on the delivered control action $\hat{u}_0$ (e.g., defining $\kappa(\theta) = \hat{u}_0(1+\delta)$ with $\theta = (\hat{\boldsymbol{u}}, \hat{\boldsymbol{x}}, \boldsymbol{\varepsilon}, \delta)$) and imposing $|\delta| \leq \Delta$, where $\Delta > 0$ is a known bound on the solution accuracy. If the solution inaccuracy is due to a premature termination of a first-order optimization method used to solve (14), a more refined analysis can be carried out within the presented framework; this is detailed in Section 4.4.*

4.3 General optimization-based controller

Clearly, there was nothing specific about the MPC structure of the optimization problem solved in the previous example and therefore the presented framework can be used to analyze arbitrary optimization-based controllers which at each time step solve an optimization problem parametrized by the output of the dynamical system (8):

$$\begin{array}{ll} \underset{\theta \in \mathbb{R}^{n_\theta}}{\text{minimize}} & J(s,\theta) \\ \text{subject to} & a(s,\theta) \geq 0 \\ & b(s,\theta) = 0, \end{array} \tag{16}$$

with J, a and b polynomial. The associated KKT system reads

$$\nabla_\theta J(s,\theta) - \nabla_\theta a(s,\theta)\lambda_a + \nabla_\theta b(s,\theta)\lambda_b = 0 \tag{17a}$$

$$\lambda_a^\top a(s,\theta) = 0 \tag{17b}$$

$$\lambda_a \geq 0 \tag{17c}$$

$$a(s,\theta) \geq 0 \tag{17d}$$

$$b(s,\theta) = 0, \tag{17e}$$

which is a system of polynomial equalities and inequalities in (s,θ,λ), where $\lambda = (\lambda_a, \lambda_b)$ and hence can be treated within the given framework. In particular the functions h and g defining the set $\mathbf{K}_s$ read

$$h(s,\theta,\lambda) = \begin{bmatrix} \nabla_\theta J(s,\theta) - \nabla_\theta a(s,\theta)\lambda_a + \nabla_\theta b(s,\theta)\lambda_b \\ \lambda_a^\top a(s,\theta) \\ b(s,\theta) \end{bmatrix}$$

and

$$g(s,\theta,\lambda) = \begin{bmatrix} \lambda_a \\ a(s,\theta) \end{bmatrix}.$$

See Remark 1 for the situation where the problem (16) is not solved exactly.

4.4 Optimization-based controller solved using a fixed number of iterations of a first order method

The presented framework can also handle the situation where the optimization problem (16) is solved using an iterative optimization method, each step of which is either an optimization problem or a polynomial mapping. This scenario was elaborated on in detail in [5], where it was shown that the vast majority of first order optimization methods fall into this category. Here we present the basic idea on one of the simplest optimization algorithms, the projected gradient method. When applied to problem (16), the iterates of the projected gradient method are given by

$$\theta_{k+1} = \mathrm{proj}_{\mathcal{S}}(\theta_k - \eta \nabla_\theta J(s, \theta_k)), \tag{18}$$

where $\mathrm{proj}_{\mathcal{S}}(\cdot)$ denotes the Euclidean projection operator on the constraint set

$$\mathcal{S} = \{\theta \in \mathbb{R}^{n_\theta} \mid a(s,\theta) \geq 0,\ b(s,\theta) = 0\}$$

and $\eta > 0$ is a step size. The update formula (18) can be decomposed into two steps: step in the direction of the negative gradient and projection on the constraint set. The first step is a polynomial mapping and the second step is an optimization problem. Indeed, equation (18) can be equivalently written as

$$\begin{array}{rcll} \theta_{k+1} & = & \underset{\theta \in \mathbb{R}^{n_\theta}}{\arg\min} & \frac{1}{2}\|\theta - (\theta_k - \eta \nabla_\theta J(s,\theta_k))\|_2^2 \\ & & \text{s.t.} & a(s,\theta) \geq 0 \\ & & & b(s,\theta) = 0. \end{array} \tag{19}$$

For each $k \in \{0, 1, \ldots\}$, the KKT system associated to (19) reads

$$\begin{aligned} \theta_{k+1} - (\theta_k - \eta \nabla_\theta J(s,\theta_k)) + \nabla_\theta\, a(s,\theta_{k+1})\lambda_a^{k+1} & \\ + \nabla_\theta\, b(s,\theta_{k+1})\lambda_b^{k+1} &= 0 && \text{(20a)} \\ b(s,\theta_{k+1}) &= 0 && \text{(20b)} \\ a(s,\theta_{k+1})^\top \lambda_a^{k+1} &= 0 && \text{(20c)} \\ a(s,\theta_{k+1}) &\geq 0 && \text{(20d)} \\ \lambda_a^{k+1} &\geq 0, && \text{(20e)} \end{aligned}$$

which is a system of polynomial equalities and inequalities. Note in particular the coupling between θ_k and θ_{k+1} in equation (20b). Assuming we apply M steps of the projected gradient method, the last iterated θ_M is therefore characterized by M coupled KKT systems of the form (20), which is a system of polynomial equalities and inequalities as required by the proposed method.

Other optimization methods, in particular most of the first order methods (e.g., fast gradient method [9], AMA [18], ADMM [2] and their accelerated versions [3]), including local non-convex methods (e.g., [4]), and some of the second order methods (e.g., the interior point method with exact line search) are readily formulated in this framework as well; see [5] for more details on first-order methods.

5 Closed-loop analysis

In this section we describe a method to analyze closed-loop properties of the interconnection depicted in Figure 1 and described in Section 3. First, notice that the closed-loop evolution is governed by the difference inclusion

$$x^+ \in f_x(x, \kappa(\mathbf{K}_s)), \tag{21a}$$

$$z^+ = f_z(z, f_y(x)). \tag{21b}$$

Since all problem data is polynomial and the set $\mathbf{K}_s$ is defined by polynomial equalities and inequalities it is possible to analyze stability and performance using sum-of-squares (SOS) programming. This is detailed next.

We will use the following notation:

$$\hat{h}(x, z, \theta, \lambda) := h(f_s(z, f_y(x)), \theta, \lambda) \tag{22a}$$

$$\hat{g}(x, z, \theta, \lambda) := g(f_s(z, f_y(x)), \theta, \lambda). \tag{22b}$$

For the rest of the paper we impose the following standing assumption:

Assumption 1. *The set $\mathbf{K}_s$ is nonempty for all $s \in \mathbb{R}^s$.*

Assumption 1 implies that the control input (6) is well defined for all $s \in \mathbb{R}^{n_s}$.

5.1 Stability analysis – global

A sufficient condition for the state x of the differential inclusion (21) to be stable is the existence of a function V satisfying

$$V(x^+, z^+, \theta^+, \lambda^+) - V(x, z, \theta, \lambda) \le -\|x\|_2^2 \tag{23a}$$

$$V(x, z, \theta, \lambda) \ge \|x\|_2^2 \tag{23b}$$

for all

$$(x, z, \theta, \lambda, x^+, z^+, \theta^+, \lambda^+) \in \mathbf{K},$$

where

$$\begin{aligned} &\mathbf{K} = \{(x, z, \theta, \lambda, x^+, z^+, \theta^+, \lambda^+) \mid x^+ = f(x, \kappa(\theta)) \\ &\hat{h}(x, z, \theta, \lambda) = 0,\ \hat{g}(x, z, \theta, \lambda) \ge 0,\ \hat{h}(x^+, z^+, \theta^+, \lambda^+) = 0, \\ &\hat{g}(x^+, z^+, \theta^+, \lambda^+) \ge 0,\ z^+ = f_z(z, f_y(x))\}. \end{aligned} \tag{24}$$

These equations require that a Lyapunov function V exists which decreases on the basic semialgebraic set $\mathbf{K}$ implicitly characterizing the closed-loop evolution (21). Therefore, we can tractably seek a Lyapunov function for system (21) by restricting V to be a polynomial of a pre-defined degree and replacing the inequalities (23) by sufficient SOS conditions according to Section 2. For better understanding we detail this replacement here and refer to the general treatment in Section 2 in the rest of the paper.

Setting

$$\xi := (x, z, \theta, \lambda, x^+, z^+, \theta^+, \lambda^+),$$

these SOS conditions read

$$\begin{aligned}
&V(x^+, z^+, \theta^+, \lambda^+) - V(x, z, \theta, \lambda) - \|x\|_2^2 = && (25a)\\
&\qquad \sigma_0(\xi) + \sigma_1(\xi)^\top \hat{g}(x, z, \theta, \lambda) + \sigma_2(\xi)^\top \hat{g}(x^+, z^+, \theta^+, \lambda^+)\\
&\qquad + p_1(\xi)^\top \hat{h}(x, z, \theta, \lambda) + p_2(\xi)^\top \hat{h}(x^+, z^+, \theta^+, \lambda^+)\\
&\qquad + p_3(\xi)(x^+ - f_x(x, \kappa(\theta)) + p_4(\xi)(z^+ - f_z(z, f_y(x)),\\
&\|x\|_2^2 - V(x, z, \theta, \lambda) = && (25b)\\
&\qquad \bar{\sigma}_0(\xi) + \bar{\sigma}_1(\xi)^\top \hat{g}(x, z, \theta, \lambda) + \bar{p}_1(\xi)^\top \hat{h}(x, z, \theta, \lambda),
\end{aligned}$$

where $\sigma_i(\xi)$ and $\bar{\sigma}_i(\xi)$ are SOS multipliers and $p(\xi)$ and $\bar{p}(\xi)$ polynomial multipliers of compatible dimensions and pre-specified degrees (selection of the degrees is discussed in Section 6). The satisfaction of (25a) implies the satisfaction of (23a) and the satisfaction of (25b) implies the satisfaction of (23b) for all $\xi \in \mathbf{K}$; this follows readily since σ_i and $\bar{\sigma}_i$ are globally nonnegative, $\hat{g}$ is nonnegative on $\mathbf{K}$ and $\hat{h}$ is zero on $\mathbf{K}$ (see Section 2).

Remark 2. *Note that instead of including the equalities $x^+ - f_x(x, \kappa(\theta))$ and $z^+ - f_z(z, f_y(x))$ in the description of $\mathbf{K}$ we could also directly substitute for x^+ and z^+. In general, direct substitution is preferred if the mappings f_x, f_z and f_y are of low degree, especially linear, in which case there is no increase in the degree of the composition of V with f_x or with f_z and f_y. Otherwise, the formulation (25) is preferred.*

From the previous discussion we conclude that closed-loop stability of the state x of (21) is implied by the feasibility of the following SOS problem:

$$\begin{array}{lll}
\text{find} & V, \sigma_0, \sigma_1, \sigma_2, p_1, p_2, p_3, p_4, \bar{\sigma}_0, \bar{\sigma}_1, \bar{p}_1 & \\
\text{s.t.} & (25a), (25b) & \\
 & \sigma_0, \sigma_1, \sigma_2, \bar{\sigma}_0, \bar{\sigma}_1 & \text{SOS polynomials} \\
 & V, p_1, p_2, p_3, p_4, \bar{p}_1 & \text{arbitrary polynomials,}
\end{array} \qquad (26)$$

where the optimization is over the coefficients of the polynomials

$$(V, \sigma_0, \sigma_1, \sigma_2, p_1, p_2, p_3, p_4, \bar{\sigma}_0, \bar{\sigma}_1, \bar{p}_1).$$

The following theorem summarizes the results of this section.

Theorem 1. *If optimization problem (26) is feasible, then the closed-loop system (21) is globally asymptotically stable.*

5.2 Stability analysis – local

The method presented in Section 5.1 addresses the verification of global asymptotic stability and imposes no restriction on the relationship between x and z, which may result in the problem (26) being infeasible, e.g., if (21) is stable only locally or if there is a known a priori relationship between x and z (e.g., if z is an estimate of x and a bound on the estimation error is known) not taken into account by the formulation of the preceding section.

In order to get a local version and/or incorporate state constraints we enforce the Lyapunov conditions (23) on the intersection of the set $\mathbf{K}$ defined in (24) with a given region of interest or state constraint set

$$\mathbf{X} := \{(x, z) \in \mathbb{R}^{n_x+n_z} \mid \psi_i(x, z) \geq 0,\ i = 1, \ldots, n_\psi\}.$$

Provided that the functions $\psi_i(\cdot)$ are polynomials, the set

$$\bar{\mathbf{K}} := \{(x, z, \theta, \lambda, x^+, z^+, \theta^+, \lambda^+) \mid (x, z, \theta, \lambda, x^+, z^+, \theta^+, \lambda^+) \in \mathbf{K},\ (x, z) \in \mathbf{X}\}$$

is defined by finitely many polynomial equalities and inequalities, leading to a verification SOS problem completely analogous to (26). However, the pitfall here is that the satisfaction of Lyapunov conditions (23) (with $\mathbf{K}$ replaced by $\bar{\mathbf{K}}$) does not ensure the invariance of the closed-loop evolution of (x, z) in the set $\mathbf{X}$. Asymptotic stability is guaranteed only on the largest sub-level set contained in $\mathbf{X}$ of the function

$$\bar{V}(x, z) = \sup_{\theta,\lambda}\{V(x, z, \theta, \lambda) \mid (x, z, \theta, \lambda) \in \hat{\mathbf{K}}\},$$

where

$$\hat{\mathbf{K}} := \{(x, z, \theta, \lambda) \mid\ \hat{g}(x, z, \theta, \lambda) \geq 0,\ \hat{h}(x, z, \theta, \lambda) = 0,\ (x, z) \in \mathbf{X}\}.$$

Note that if we choose V as a function of (x, z) only, then trivially $\bar{V}(x, z) = V(x, z)$. In general, however, the function $\bar{V}(x, z)$ is a solution to a complicated parametric polynomial optimization problem and therefore not known analytically. Therefore, we resort to computing an inner approximation to the largest super level set of $\bar{V}(x, z)$ included in $\mathbf{X}$ by first approximating $\bar{V}(x, z)$ from above by a polynomial function $p(x, z)$ of a prescribed degree and computing the largest sub-level set of $p(x, z)$ included in $\mathbf{X}$. Approximating $\bar{V}(x, z)$ from above can be done using the following SOS optimization problem

$$\begin{array}{ll} \underset{p(x,z)}{\text{minimize}} & \int_{\mathbf{X}} p(x, z)\, d(x, z) \\ \text{s.t.} & p(x, z) \geq V(x, z, \theta, \lambda) \quad \forall\, (x, z, \theta, \lambda) \in \hat{\mathbf{K}}. \end{array} \tag{27}$$

Since the set $\hat{\mathbf{K}}$ is basic semialgebraic, the constraint immediately translates to an SOS constraint (with conservatism inversely proportional to the degree of the

SOS multipliers), and the objective is a linear combination of the coefficients of the polynomial $p(x,z)$, where the coefficients of the linear combination are the moments of the Lebesgue measure over $\mathbf{X}$. Problem (27) therefore immediately translates to an SDP (see Section 2).

Once the polynomial p is found, an inner approximation to the maximum level set of $\bar{V}(x,z)$ contained in $\mathbf{X}$ can be found by solving

$$\begin{array}{ll} \underset{\gamma\in\mathbb{R}_+,\{\sigma_{0,i}\},\{\sigma_{0,i}\}}{\text{maximize}} & \gamma \\ \text{s.t. } \psi_i(x,z) = \sigma_{0,i}(x,z) + \sigma_{1,i}(x,z)(\gamma - p(x,z)), & \\ \quad \{\sigma_{0,i}\},\ \{\sigma_{1,i}\} \quad \text{SOS polynomials} \quad \forall\, i \in \{1,\ldots,n_\psi\}. & \end{array} \tag{28}$$

The satisfaction of the first constraint implies that $\psi_i(x,z) \geq 0$ for all (x,z) such that $p(x,z) \leq \gamma$ and all $i \in \{1,\ldots,n_\psi\}$; therefore $\{(x,z) \mid p(x,z) \leq \gamma\} \subset \mathbf{X}$ for any γ feasible in (28). Maximizing γ then maximizes the size of the inner approximation.

Problem (28) is only quasi-convex because of the bilinearity between σ_1 and γ but can be efficiently solved using a bi-section on γ. Indeed, for a fixed value of γ problem (28) is an SDP, typically of much smaller size than (26).

The following Lemma summarizes the results of this section.

Lemma 1. *If a polynomial p is feasible in (27) and $\gamma \in \mathbb{R}_+$ feasible in (28), then the set $\{(x,z) \mid p(x,z) \leq \gamma\}$ provides an inner approximation to the maximum sub-level set of $\bar{V}$ included in $\mathbf{X}$.*

This immediately leads to the following theorem.

Theorem 2. *If a polynomial V is feasible in (26), polynomial p feasible in (27) and $\gamma \in \mathbb{R}_+$ feasible in (28), then all trajectories of the closed-loop system (21) starting from the set $\{(x,z) \mid p(x,z) \leq \gamma\}$ lie in the set $\mathbf{X}$ and $\lim_{t\to\infty} x_t = 0$.*

5.3 Performance analysis – deterministic setting

In this section we analyze the performance of the controller (6) with respect to a given infinite-time cost function

$$\mathcal{C}(x_0,z_0) = L\alpha^{\tau(x_0,z_0)} + \sum_{t=0}^{\tau(x_0,z_0)-1} \alpha^t l(x_t,u_t), \tag{29}$$

where $\alpha \in (0,1)$ is a discount factor, l is a polynomial stage cost,

$$\tau(x,z) := \inf\{t \in \{1,2,\ldots\} \mid (x_t,z_t) \notin \mathbf{X},\ (x_0,z_0) = (x,z)\} \tag{30}$$

is the first time that the state (x_t,z_t) of (21) leaves $\mathbf{X}$ (setting $\tau(x,z) = +\infty$ if $(x_t,u_t) \in \mathbf{X}$ for all t) and

$$L > \max\{l(x,u) \mid (x,z) \in \mathbf{X}, u \in \kappa(\mathbf{K}_s)\}/(1-\alpha) \tag{31}$$

is a constant upper bounding the stage cost l on $\mathbf{X}$ divided by $1-\alpha$. We assume that $L < \infty$, which is fulfilled if the projection of $\mathbf{X}$ on $\mathbb{R}^{n_x}$ is bounded and the set $\mathbf{K}_s$ is bounded for all $s = f_s(z, f_y(x))$ with $(x,z) \in \mathbf{X}$. A constant L satisfying (31) is usually easily found since $\mathbf{X}$ is known and the controller (6) is usually set up in such a way that it satisfies the input constraints of system (5), which are typically a bounded set of a simple form.

The reason for choosing (29) is twofold. First, $\mathcal{C}(x_0,z_0) = \sum_{t=0}^{\infty} \alpha^t l(x_t,u_t)$ for all (x_0,z_0) such that $(x_t,z_t) \in \mathbf{X}$ for all t; that is, whenever (x_t,z_t) stays in the state constraint set $\mathbf{X}$ for all t, the cost (29) coincides with the standard infinite-horizon discounted cost. Second, $\mathcal{C}(x_0,z_0) \le L$ for all $(x_0,z_0) \in \mathbf{X}$; that is, the cost function is bounded on $\mathbf{X}$, which enables us to obtain polynomial upper and lower bounds on $\mathcal{C}$ (which is not possible if $\mathcal{C}$ is infinite outside the maximum positively invariant subset of (21) included in $\mathbf{X}$ as is the case for the standard infinite-horizon discounted cost).

In the rest of this section we derive polynomial upper and lower bounds on $\mathcal{C}(x,z)$. To this end define

$$\hat{\mathbf{K}}_c := \{(x,z,\theta,\lambda) \mid \hat{g}(x,z,\theta,\lambda) \ge 0,\ \hat{h}(x,z,\theta,\lambda) = 0,\ (x,z) \notin \mathbf{X}\}.$$

The upper bound is based on the following lemma:

Lemma 2. *If*

$$V(x,z,\theta,\lambda) - \alpha V(x^+,z^+,\theta^+,\lambda^+) - l(x,\kappa(\theta)) \ge 0 \qquad (32)$$
$$\forall\,(x,z,\theta,\lambda,x^+,z^+,\theta^+,\lambda^+) \in \bar{\mathbf{K}},$$

$$V(x,z,\theta,\lambda) \ge L \quad \forall\,(x,z,\theta,\lambda) \in \hat{\mathbf{K}}_c \qquad (33)$$

and

$$\overline{V}(x,z) \ge V(x,z,\theta,\lambda) \quad \forall\,(x,z,\theta,\lambda) \in \hat{\mathbf{K}}, \qquad (34)$$

then $\overline{V}(x,z) \ge \mathcal{C}(x,z)$ *for all* $(x,z) \in \mathbf{X}$.

Proof. See the Appendix. □

The lower bound is based on the following lemma:

Lemma 3. *If*

$$V(x,z,\theta,\lambda) - \alpha V(x^+,z^+,\theta^+,\lambda^+) - l(x,\kappa(\theta)) \le 0 \qquad (35)$$
$$\forall\,(x,z,\theta,\lambda,x^+,z^+,\theta^+,\lambda^+) \in \bar{\mathbf{K}},$$

$$V(x,z,\theta,\lambda) \le L \quad \forall\,(x,z,\theta,\lambda) \in \hat{\mathbf{K}}_c \qquad (36)$$

and

$$\underline{V}(x,z) \le V(x,z,\theta,\lambda) \quad \forall\,(x,z,\theta,\lambda) \in \hat{\mathbf{K}}, \qquad (37)$$

then $\underline{V}(x,z) \le \mathcal{C}(x,z)$ *for all* $(x,z) \in \mathbf{X}$.

Proof. Analogous to the proof of Lemma 2. □

The previous two lemmas lead immediately to optimization problems providing upper and lower bounds on $\mathcal{C}(x, z)$.

An upper bound on $\mathcal{C}(x, z)$ is given by the following optimization problem:

$$\begin{array}{ll} \underset{V,\overline{V}}{\text{minimize}} & \int_{\mathbf{X}} \overline{V}(x,z)\rho(x,z)d(x,z) \\ \text{s.t.} & (32), (33), (34), \end{array} \tag{38}$$

where $\rho(x, z)$ is a user-defined nonnegative weighting function allowing one to put a different weight on different initial conditions. Typical examples are $\rho(x, z) = 1$ or $\rho(x, z)$ equal to the indicator function of a certain subset of $\mathbf{X}$ (see Example 7.1).

A lower bound on $\mathcal{C}(x, z)$ is given by the following optimization problem:

$$\begin{array}{ll} \underset{V,\underline{V}}{\text{maximize}} & \int_{\mathbf{X}} \underline{V}(x,z)\rho(x,z)d(x,z) \\ \text{s.t.} & (35), (36), (37). \end{array} \tag{39}$$

In both optimization problems, the optimization is over continuous functions $(V, \overline{V})$ or $(V, \underline{V})$; in order to make the problems tractable we restrict the class of functions to polynomials of predefined degrees and replace the nonnegativity conditions (32), (33), (34) or (35), (36), (37) by sufficient SOS conditions (see Section 2). For (32), (34) and (35), (37), these conditions are completely analogous to (25). For (33) and (36) we have to deal with the condition $(x, z) \notin \mathbf{X}$. A sufficient condition for (33) is

$$\begin{aligned} V(x,z,\theta,\lambda) - L = {} & -\sigma_{\psi_i}(\zeta)\psi_i(x,z) + \sigma_0(\zeta) \\ & + \sigma_1(\zeta)^\top \hat{g}(x,z,\theta,\lambda) + \bar{p}_1(\zeta)^\top \hat{h}(x,z,\theta,\lambda) \quad \forall\, i \in \{1,\ldots,n_\psi\}, \end{aligned} \tag{40}$$

where

$$\zeta := (x, z, \theta, \lambda),$$

σ_0, σ_1 and σ_{ψ_i}'s are SOS and $\bar{p}_1$ is a polynomial. For each $i \in \{1, \ldots, n_\psi\}$ this condition implies that $V(x, z, \theta, \lambda) - L \geq 0$ on

$$\mathbf{K}_{c,i} = \{(x,z,\theta,\lambda) \mid \hat{g}(x,z,\theta,\lambda) \geq 0,\ \hat{h}(x,z,\theta,\lambda) = 0,\ \psi_i(x,z) \leq 0\}.$$

Since $\cup_{i=1}^{n_\psi} \mathbf{K}_{c,i} = \mathbf{K}_c$, the condition (40) indeed implies (33). A sufficient condition for (36) is obtained by replacing the left-hand side of (40) by $L - V(x, z, \theta, \lambda)$.

Therefore, all constraints of the optimization problems (38) and (39) can be enforced through sufficient SOS conditions. The objective function is linear in the coefficients of the polynomials $\overline{V}$ or $\underline{V}$ and can be evaluated in closed form for simple sets $\mathbf{X}$; see Section 2 for details.

In conclusion, by restricting the class of decision variables in (38) and (39) to polynomials of a prescribed degree and replacing the nonnegativity constraints by sufficient SOS conditions, the problems (38) and (39) become SOS problems with linear objective and hence immediately translate to SDPs.

5.4 Performance analysis – stochastic setting

A small modification of the developments from the previous section allows us to analyze the performance in a stochastic setting, where (21) is replaced by

$$x^+ \in f_x(x, \kappa(\mathbf{K}_s), w), \tag{41a}$$
$$z^+ = f_z(z, f_y(x, v)), \tag{41b}$$

where (w, v) is an iid (with respect to time) process and measurement noise with known joint probability distributions $P_{w,v}$, i.e.,

$$\mathbb{P}(w \in A, v \in B) = P_{w,v}(A \times B)$$

for all Borel sets $A \subset \mathbb{R}^{n_w}$ and $B \subset \mathbb{R}^{n_v}$. We analyze the performance with respect to the cost function

$$\mathcal{C}_{\mathrm{s}}(x_0, z_0) = \mathbf{E}\Big\{L\alpha^{\tau(x_0,z_0)}\} + \sum_{t=0}^{\tau(x_0,z_0)-1} \alpha^t l(x_t, u_t)\Big\}, \tag{42}$$

where $\tau(x_0, z_0)$ defined in (30) is now a random variable and L satisfies (31). The expectation in (42) is over the realizations of the stochastic process $(w_t, v_t)_{t=0}^{\infty}$. The rationale behind (42) is the same as behind (29).

The stochastic counterpart to Lemma 2 reads

Lemma 4. *If*

$$\overline{V}(x,z) - \alpha\mathbf{E}\overline{V}(f_x(x,\kappa(\theta),w), f_z(z, f_y(x,v))) - l(x,\kappa(\theta)) \geq 0 \quad \forall\, (x,z,\theta,\lambda) \in \hat{\mathbf{K}}, \tag{43}$$

and

$$\overline{V}(x,z) \geq L \quad \forall\, (x,z) \in \mathbf{X}^c, \tag{44}$$

then $\overline{V}(x,z) \geq \mathcal{C}_{\mathrm{s}}(x,z)$ *for all* $(x,z) \in \mathbf{X}$*,*

where $\mathbf{X}^c$ is the complement of $\mathbf{X}$.

Proof. See the Appendix. □

The stochastic counterpart to Lemma 3 reads

Lemma 5. *If*

$$\underline{V}(x,z) - \alpha\mathbf{E}\underline{V}(f_x(x,\kappa(\theta),w), f_z(z,f_y(x,v))) - l(x,\kappa(\theta)) \le 0 \quad \forall\,(x,z,\theta,\lambda) \in \hat{\mathbf{K}}, \tag{45}$$

and

$$\underline{V}(x,z) \le L \quad \forall\,(x,z) \in \mathbf{X}^c, \tag{46}$$

then $\underline{V}(x,z) \le \mathcal{C}_{\mathrm{s}}(x,z)$ *for all* $(x,z) \in \mathbf{X}$.

Proof. Similar to the proof of Lemma 4. □

Upper and lower bounds on $\mathcal{C}_{\mathrm{s}}$ are then obtained by

$$\begin{array}{ll} \underset{\overline{V}}{\text{minimize}} & \int_{\mathbf{X}} \overline{V}(x,z)\rho(x,z)d(x,z) \\ \text{s.t.} & (43), (44), \end{array} \tag{47}$$

and

$$\begin{array}{ll} \underset{\underline{V}}{\text{maximize}} & \int_{\mathbf{X}} \underline{V}(x,z)\rho(x,z)d(x,z) \\ \text{s.t.} & (45), (46), \end{array} \tag{48}$$

where $\rho(x,z)$ is a given nonnegative weighting function. Polynomial upper and lower bounds on $\mathcal{C}_{\mathrm{s}}$ are obtained by restricting the functions $\overline{V}$ and $\underline{V}$ to polynomials and replacing the nonnegativity constraints by sufficient SOS constraints in exactly the same fashion as in the deterministic setting. The expectation in the constraints (43) and (45) is handled as follows: Given a polynomial $p(x,w,v) = \sum_{\alpha,\beta,\gamma} p_{(\alpha,\beta,\gamma)} x^\alpha w^\beta v^\gamma$ with coefficients $\{p_{(\alpha,\beta,\gamma)}\}$ indexed by multiindices (α,β,γ), we have

$$\mathbf{E}\,p(x,w,v) = \int p(x,w,v)dP_{w,v}(w,v) = \sum_{\alpha,\beta,\gamma} p_{(\alpha,\beta,\gamma)} x^\alpha \int w^\beta v^\gamma dP_{w,v}(w,v),$$

where the moments $\int w^\beta v^\gamma dP_{w,v}(w,v)$ are fixed numbers and can be precomputed offline. Hence, the expectation in (43) and (45) is linear in the decision variables, as required, and is available in closed form provided that the moments of $P_{w,v}$ are known.

Remark 3. *Note that in problems (47) and (48) we use only one function* $\overline{V}$ *and* $\underline{V}$ *instead of pairs of functions* $(V, \overline{V})$ *and* $(V, \underline{V})$ *in problems (38) and (39). Using a pair of functions gives more degrees of freedom and hence smaller conservatism of the upper and lower bounds, but is difficult to use in the stochastic setting because of the need to evaluate the expectation of a function of* (θ^+, λ^+) *which has an unknown dependence on* (w,v). *In order to overcome this, one would either have to impose additional assumptions or resort to a worst-case approach.*

5.5 Robustness analysis – $\mathcal{L}_2$ gain, ISS

In this section we describe how to analyze performance in a robust setting in terms of the $\mathcal{L}_2$ gains from w and v to a performance output

$$\hat{y} = f_{\hat{y}}(x), \tag{49}$$

where $f_{\hat{y}}$ is a polynomial. We assume the same dynamics (41) as in Section (5.4) but now all that is known about w and v is that they take values in a given (possibly state-dependent) set

$$\mathbf{W}(x,z) = \{(w,v) \in \mathbb{R}^{n_w} \times \mathbb{R}^{n_v} \mid \psi_w(x,z,w,v) \geq 0\}, \tag{50}$$

where each component of $\psi_w : \mathbb{R}^{n_x+n_z+n_w+n_v} \to \mathbb{R}^{n_{\psi_w}}$ is a polynomial in (x,z,w,v). Note that we do not a priori assume that the set $\mathbf{W}(x,z)$ is compact.

Defining

$$\begin{aligned}\mathbf{K}_w = \big\{(x,z,\theta,\lambda,w,v,x^+z^+,\theta^+,\lambda^+,w^+,v^+) \mid \\ \hat{h}(x,z,\theta,\lambda) = 0,\ \hat{g}(x,z,\theta,\lambda) \geq 0, \\ \hat{h}(x^+,z^+,\theta^+,\lambda^+) = 0,\ \hat{g}(x^+,z^+,\theta^+,\lambda^+) \geq 0, \\ \psi_w(x,z,w,v) \geq 0,\ \psi_w(x^+,z^+,w^+,v^+) \geq 0, \\ x^+ - f_x(x,\kappa(\theta),w) = 0,\ z^+ - f_z(z, f_y(x,v)) = 0\big\},\end{aligned} \tag{51}$$

and

$$\begin{aligned}\hat{\mathbf{K}}_w = \big\{(x,z,\theta,\lambda,w,v) \mid \hat{h}(x,z,\theta,\lambda) = 0,\ \hat{g}(x,z,\theta,\lambda) \geq 0, \\ \psi_w(x,z,w,v) \geq 0\},\end{aligned} \tag{52}$$

we can seek a function V such that

$$\begin{aligned}V(x^+,z^+,\theta^+,\lambda^+,w^+,v^+) - V(x,z,\theta,\lambda,w,v) \leq -\|f_{\hat{y}}(x)\|_2^2 + \alpha_w\|w\|_2^2 + \alpha_v\|v\|_2^2 \\ \forall\ (x,z,\theta,\lambda,w,v,x^+z^+,\theta^+,\lambda^+,w^+,v^+) \in \mathbf{K}_w,\end{aligned} \tag{53}$$

$$V(x,z,\theta,\lambda,w,v) \geq 0 \quad \forall\ (x,z,\theta,\lambda,w,v) \in \hat{\mathbf{K}}_w. \tag{54}$$

and

$$V(0,0,\theta,\lambda,w,v) = 0 \quad \forall\ (0,0,\theta,\lambda,w,v) \in \hat{\mathbf{K}}_w. \tag{55}$$

The following lemma and its immediate corollary links the satisfaction of (53), (54) and (55) to the $\mathcal{L}_2$ gain from w and v to $\hat{y}$.

Lemma 6. *If V satisfies (53), (54) for some $\alpha_w \geq 0$ and $\alpha_v \geq 0$, then*

$$\sum_{t=0}^{\infty} \|\hat{y}_t\|_2^2 \leq V(x_0, z_0, \theta_0, \lambda_0, w_0, v_0) + \alpha_w \sum_{t=0}^{\infty} \|w_t\|_2^2 + \alpha_v \sum_{t=0}^{\infty} \|v_t\|_2^2. \tag{56}$$

Proof. See the Appendix. □

Corollary 1. *If V satisfies (53), (54) and (55) for some $\alpha_w \geq 0$ and $\alpha_v \geq 0$, then the $\mathcal{L}_2$ gain from w to $\hat{y}$ respectively from v to $\hat{y}$ is bounded by α_w respectively α_v.*

Proof. Follows by setting $(x_0, z_0) = (0, 0)$ and using (55) which implies that

$$V(x_0, z_0, \theta_0, \lambda_0, w_0, v_0) = 0$$

in (56). □

Minimization of an upper bound on the $\mathcal{L}_2$ gain from w and v to $\hat{y}$ is then achieved by the following optimization problem:

$$\begin{array}{ll} \underset{V, \alpha_w, \alpha_v}{\text{minimize}} & \alpha_w + \gamma \alpha_v \\ \text{s.t.} & (53), (54), (55), \end{array} \tag{57}$$

where the parameter $\gamma \geq 0$ trades off the minimization of the $\mathcal{L}_2$ gains from w to $\hat{y}$ and from v to $\hat{y}$.

Remark 4. *If instead of (54) we require $V(x, z, \theta, \lambda, w, v) \geq \|x\|^2$ for all $(x, z, \theta, \lambda, w, v) \in \hat{\mathbf{K}}_w$, then this along with (53) implies that the system (41) is input-to-state stable (ISS) with respect to the input $(w, v) \in \mathbf{W}(x, z)$.*

Since the sets $\mathbf{K}_w$ and $\hat{\mathbf{K}}_w$ are defined by finitely many polynomial equalities and inequalities, we can find upper bounds on the $\mathcal{L}_2$ gains α_w and α_v by restricting V to be a polynomial of a prescribed degree and replacing the nonnegaivity constraints of (57) by sufficient SOS conditions according to Section 2.

6 Computational aspects

As discussed in Section 2, the constraint (2) translates to an SDP constraint. There is a natural tradeoff between computational complexity and the richness of the class of functions satisfying (2). This tradeoff is, for the most part, controlled by the size of the polynomial basis parametrizing the function V and by the vector of polynomials $\beta(x)$ in (3) associated to the SOS multipliers. In general, the larger $\beta(x)$ (in terms of the number of components), the larger the set of functions covered, but the higher the computational complexity. A typical choice for $\beta(x)$ is the vector of all multivariate monomials up to a given total degree. Another choice may be the set of all multivariate monomials up to a given total degree corresponding to a given subset of the variables $x = (x_1, \ldots, x_n)$; this is used in the numerical examples of this paper.

More specific monomial selection techniques require some insight into the problem structure. Fortunately, there exist automatic monomial reduction techniques (e.g., the Newton polytope [17]) which discard those monomials in $\beta(x)$ which cannot appear in the decomposition of $\sigma(x)$. It is also possible to directly reduce the size of the SDP arising from (3) using the facial reduction algorithm of [12]; the benefit of this approach is that it works at the SDP level and is therefore not limited to the situation where $\beta(x)$ is a subset of the monomial basis, allowing one to use numerically better conditioned bases (e.g., the Chebyshev basis).

The transformation from the abstract form (3) can be carried out automatically using freely available modeling tools such as Yalmip [8], SOSTOOLS [10] or SOSOPT [16] and solved using SDP solvers such as SeDuMi [14] or MOSEK.

7 Numerical examples

This section illustrates the approach on two numerical examples. The SOS problems were modeled using SOSOPT [16] and solved using MOSEK. For both examples we report the parsing time of SOSOPT, the time to carry out monomial reduction by SOSOPT and the solve time of MOSEK. The bottleneck of the approach is the monomial reduction phase (which, however, is very effective in the sense of reducing the size of the problem significantly). The authors expect that a more efficient implementation of the reduction phase and polynomial handling in general would allow the approach to scale much beyond what is presented here.

7.1 Bilinear system + PI with saturation – performance analysis

First we demonstrate the approach on a bilinear dynamical system

$$\begin{aligned} x^+ = f_x(x,u) &:= \begin{bmatrix} 0.9x_1 + u + 0.2ux_1 \\ 0.85x_2 + x_1 \end{bmatrix} \\ y = f_y(x) &:= x_2 \end{aligned}$$

controlled by a PI controller with input saturation given by

$$\begin{aligned} z^+ = f_z(z,y) &:= z - k_i y \\ s = f_s(z,y) &:= k_p(z - y) \end{aligned}$$

Figure 2: Bilinear system performance bound – Red: upper bound $V(x, 0)$ of degree 6. Blue: true closed-loop cost $J(x, 0)$.

with $k_p = 0.05$, $k_i = 0.02$. The control input is given by saturating u on the input constraint set $U = [-0.5, 0.5]$, i.e, $u = \mathrm{proj}_U(s)$. In addition the system is subject to the state constraints $\|x\|_\infty \leq 10$. In view of Section 4.1, this set up can be analyzed using the presented method. The goal is to estimate the performance of this closed-loop system with respect to the cost function (29) with $l(x, u) = \|x\|^2 + u^2$, $\alpha = 0.95$ and $L = (2 \cdot 10^2 + 0.5^2)/(1 - \alpha) = 4.05 \cdot 10^3$ chosen according to (31). We estimate the performance using the optimization problem (38), where we consider V as function of (x, z) only and therefore do not need the upper bounding function $\overline{V}$. Assume that we are interested only in closed-loop performance for initial conditions starting from $\mathbf{X}' = \{x \mid \|x\|_\infty \leq 1\}$ and $z = 0$ (i.e., zero integral component at the beginning of the closed-loop evolution), which is a strict subset of the set $\mathbf{X} := \{(x, z) \mid \|x\|_\infty \leq 10, z \in \mathbb{R}\}$. To this effect we minimize $\int_{\mathbf{X}'} V(x, 0)\, dx$ as the objective of (38), which corresponds to setting $\rho(x, z) = \mathbb{I}_{\mathbf{X}'}(x)\delta_0(z)$, where $\mathbb{I}_{\mathbf{X}'}$ is the indicator function of $\mathbf{X}'$ and δ_0 the dirac distribution centered at zero. We compare the upper bound obtained by solving (38) with the exact cost function evaluated on a dense grid of initial conditions in $\mathbf{X}'$ by forward simulation of the closed-loop system. The comparison is in Figure (2); we see a relatively good fit over the whole region of interest $\mathbf{X}'$. The constraints (32) and (33) of (38) were replaced with sufficient SOS conditions with SOS multipliers of degree four containing only monomials in (x, z) and polynomial multipliers of degree three containing monomials in (x, z, θ, λ).

7.2 Uncertain linear system – global asymptotic stability

Consider the Quanser active suspension model in continuous-time $\dot{x} = A_c x + B_c u$ with

$$A_c = \begin{bmatrix} 0 & 1 & 0 & -1 \\ -K_s/M_s & -B_s/M_s & 0 & B_s/M_s \\ 0 & 0 & 0 & 1 \\ K_s/M_{us} & B_s/M_{us} & -K_{us}/M_{us} & -(B_s + B_{us})/M_{us} \end{bmatrix},$$

$$B_c = \begin{bmatrix} 0 & 1/M_s & 0 & -1/M_{us} \end{bmatrix}^T ,$$

where $K_s = 1205$, $K_{us} = 2737$, $M_{us} = 1.5$, $B_s = 20$, $B_{us} = 20$ and the mass M_s is unknown and possibly time-varying in the interval $[2.85, 4]$. After discretization[3] with sampling period 0.01, this model can be written as $x^+ = (A_0 + A_1 w)x + (B_0 + B_1 w)x$, where $w := 1/M_s \in [1/4, 1/2.85]$.

We consider an MPC controller (with perfect state measurement) with cost function given by matrices $Q = I$ and $R = 20$ minimized over prediction horizon N subject to input constraint $|u| \leq 250$ and nominal dynamics $x^+ = A_0 x + B_0 u$. This problem is expressed in a dense form (i.e., the state is eliminated using the dynamics equation) to which we apply M steps of the projected gradient method (18) (see Section 4.4) initialized with the LQ solution and seek a quadratic ISS Lyapunov function V (see Remark 4) while minimizing the $\mathcal{L}_2$ gain α_w using the optimization problem (57) (with $\alpha_v = 0$). The problem (57) is feasible (for all combinations of M and N tested) when we take the SOS multipliers σ_1, σ_2 in equation (25a) of degree two in (x, θ) and the polynomial multipliers p_1, p_2 of degree one in (x, θ, λ). The list of monomials $\beta(x, \theta, \lambda, w)$ constituting the multiplier σ_0 in the form $\beta(x, \theta, \lambda, w)^T \mathcal{Q} \beta(x, \theta, \lambda, w)$, $\mathcal{Q} \succeq 0$, is determined automatically by SOSOPT and contains monomials linear in λ, x, θ and w, and products $x \cdot w$ and $\theta \cdot w$. In Eq. (25b) we set all multipliers to zero except for $\bar{\sigma}_0$, monomials of which are again determined automatically by SOSOPT. Determining the smallest list of monomials $\beta(x, \theta, \lambda, w)$ takes the most time of the whole procedure; this is documented by Table 1 reporting the time breakdown for different values of N and M. The optimal $\mathcal{L}_2$ gain α_w is equal to zero, showing closed-loop global robust asymptotic stability (i.e., convergence $\|x_k\| \to 0$ for any sequence $\{w_k \in [1/4, 1/2.85]\}_{k=0}^{\infty}$). Figure 3 shows a sample trajectory of $\|x_k\|$, $V(x_k)$ (the Lyapunov function is function of x only in this case) and u_k and w_k for $N = M = 4$.

Table 1: Global asymptotic stability of an uncertain system – timing breakdown as a function of the number of iterations of the projected gradient method M and the horizon length N used in the cost function. The parsing and monomial reduction was carried out by SOSOPT; the SDP solve by MOSEK.

	parsing	monomial reduction	SDP solve
$M, N = 1, 1$	0.99 s	1.12 s	0.45 s
$M, N = 2, 2$	1.56 s	5.9 s	0.5 s
$M, N = 3, 3$	4.85 s	65.5 s	1.95 s
$M, N = 4, 4$	45 s	755 s	13.9 s

[3] The matrices A_0, A_1, B_0, B_1 were found as a least-squares fit of the continuous-time dynamics discretized on a grid of values of $w \in [1/4, 1/2.85]$.

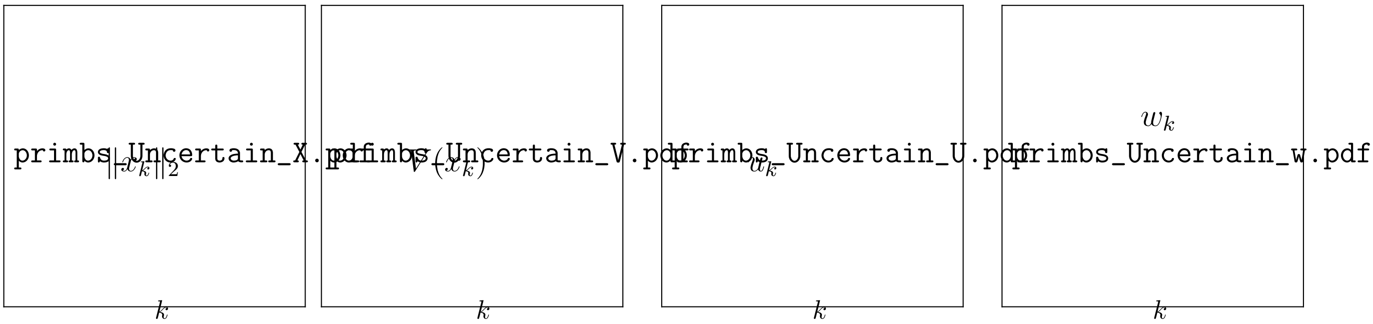

Figure 3: Global asymptotic stability of an uncertain system – one trajectory starting from the initial condition $x_0 = [1\ 1\ 1\ 1]^T$ of the norm of the state $\|x_k\|$, the Lyapunov function $V(x_k)$, the control input u_k and the disturbance w_k.

7.3 Local stability of a quadcopter

This example investigates stability of a linearized attitude and vertical velocity model of a quadcopter. The system has seven states (Roll, Pitch and Yaw angles and angular velocities, and velocity in the vertical direction) and four control inputs (the thrusts of the four rotors). The system is controlled by a one-step MPC controller (with perfect state information) which at time k approximately minimizes the cost $x_k^T Q x_k + u_k^T R u_k + x_{k+1}^T P x_{k+1}$, where $Q = I$, $R = 10I$ and P is the infinite-time LQ matrix associated to the cost matrices Q and R, using one step of the projected gradient method (18) subject to the input constraints $\|u\|_\infty \le 1$. This model is open-loop unstable and therefore we investigate closed-loop stability in the region $\mathbf{X} = [-1, 1]^7$ as described in Section 5.2. The SOS problem (26) is feasible when seeking a quadratic Lyapunov function using SOS multipliers σ_1, σ_2 in equation (25a) of degree two in x and the polynomial multipliers p_1, p_2 of degree one in (x, θ, λ). The smallest set of monomials constituting σ_0 is chosen automatically by SOSOPT. In (25b), we chose all multipliers zero except for $\bar{\sigma}_0$ whose monomials are determined automatically by SOSOPT. Computing the largest γ such that $\{x \mid V(x) \le \gamma\}$ is included in $\mathbf{X}$ yields $\gamma = 6.37$; this proves that all trajectories starting in $\{x \mid V(x) \le \gamma\}$ stay there and converge to the origin. One closed-loop trajectory of $\|x\|_2$, $V(x)$ and u are depicted in Figure 4; note that this trajectory does not start in $\{x \mid V(x) \le \gamma\}$ but still converges to the origin and the Lyapunov function decreases. The parsing time and monomial reduction carried out by SOSOPT took 2.7 s and 16.2 s, respectively; the MOSEK solve time was 0.55 s.

8 Conclusion

We presented a method for closed-loop analysis of polynomial dynamical systems controlled by an optimization based controller. Provided that all data is polynomial

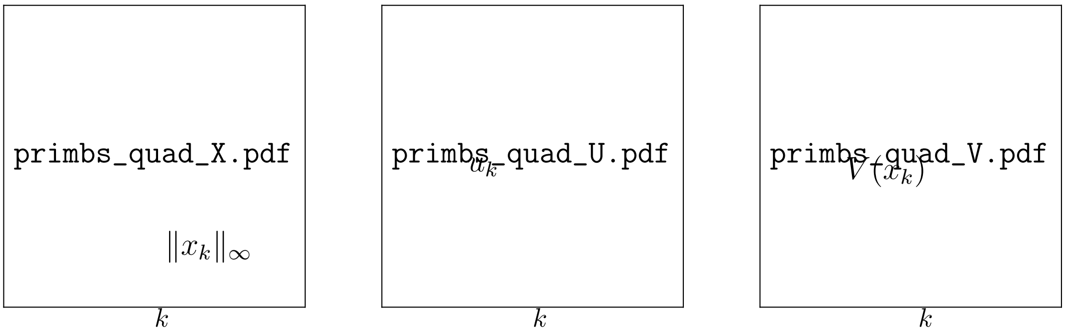

Figure 4: Local stability of a quadcopter – trajectories of the norm of the state $\|x_k\|$, the Lyapunov function $V(x_k)$, the control input u_k for initial condition $x_0 = [1\ 1\ 1\ 1\ 1\ 1\ 1]^\top$.

the analysis problem boils down to a semidefinite programming (SDP) problem which can be readily modeled and solved using freely available tools. The limiting factor is the parsing time of the SOS problems, which, however, should be possible to ameliorate by a tailored implementation. Besides the parsing time, the second limiting factor is the size of the resulting SDP problem, especially if tighter performance bounds are required; using first-order-like SDP solvers (e.g., SDPNAL [20]), and/or parellel solvers (e.g., SDPARA [19]) should enable the presented method to scale beyond the reach of interior point methods used in this paper.

Acknowledgments

The first author would like to thank Ye Pu for a discussion that initiated the work, and Zlatko Emedji and Mahdieh Sadat Sadabad for providing the uncertain system for the numerical example.

The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP/2007-2013) / Grant agreement number 607957 (TEMPO).

Appendix

Proof of Lemma 2

Let (x_t, z_t) be a solution to (21) for $t \in \{0, 1, \ldots\}$ and let $u_t \in \kappa(\mathbf{K}_s)$. Then there exist $\theta_t \in \mathbb{R}^{n_\theta}$ and $\lambda_t \in \mathbb{R}^{n_\lambda}$ such that

$$(x_t, z_t, \theta_t, \lambda_t, x_{t+1}, z_{t+1}, \theta_{t+1}, \lambda_{t+1}) \in \bar{\mathbf{K}} \tag{58}$$

and

$$(x_t, z_t, \theta_t, \lambda_t) \in \hat{\mathbf{K}} \tag{59}$$

for all $t \in \{0, 1, \ldots, \tau - 1\}$, where $\tau := \tau(x_0, z_0)$ is defined in (30) and

$$(x_\tau, z_\tau, \theta_\tau, \lambda_\tau) \in \hat{\mathbf{K}}_{\mathbf{c}}. \tag{60}$$

Using (32) and (58), we conclude that

$$V(x_t, z_t, \theta_t, \lambda_t) - \alpha V(x_{t+1}, z_{t+1}, \theta_{t+1}, \lambda_{t+1}) - l(x_t, u_t) \geq 0$$

for all $t \in \{0, 1, \ldots, \tau - 1\}$. This implies that

$$\alpha^\tau V(x_\tau, z_\tau, \theta_\tau, \lambda_\tau) + \sum_{t=0}^{\tau-1} \alpha^t l(x_t, u_t) \leq V(x_0, z_0, \theta_0, \lambda_0).$$

Using (33) and (60) we conclude that

$$\mathcal{C}(x_0, z_0) = \alpha^\tau L + \sum_{t=0}^{\tau-1} \alpha^t l(x_t, u_t) \leq V(x_0, z_0, \theta_0, \lambda_0).$$

Using (34) and (59) we have $V(x_0, z_0, \theta_0, \lambda_0) \leq \overline{V}(x_0, z_0)$ and hence $\overline{V}(x_0, z_0) \geq \mathcal{C}(x_0, z_0)$ as desired. ■

Proof of Lemma 4

The proof proceeds along similar lines as the deterministic version by decomposing the probability space according to the values of the stopping time τ. On the probability event $\{\tau = k\}$ we get, by iterating the inequality (43),

$$\alpha^k V(x_k, z_k) + \mathbf{E}\Big\{ \sum_{t=0}^{k-1} \alpha^t l(x_t, u_t) \mid \tau = k \Big\} \leq V(x_0, z_0).$$

Since $(x_k, z_k) \notin \mathbf{X}$ on $\{\tau = k\}$ we have $V(x_k, z_k) \geq L$ by (44) and hence

$$\alpha^k L + \mathbf{E}\{\sum_{t=0}^{k-1} \alpha^t l(x_t, u_t) \mid \tau = k\} \leq V(x_0, z_0)$$

on $\{\tau = k\}$. Summing over k gives the result.

Proof of Lemma 6

Let (x_t, z_t) be a solution to (41) with $(w_t, v_t) \in \mathbf{W}(x_t, z_t)$ for all $t \in \{0, 1, \ldots\}$. Then there exist $\theta_t \in \mathbb{R}^{n_\theta}$ and $\lambda_t \in \mathbb{R}^{n_\lambda}$ such that

$$(x_t, z_t, \theta_t, \lambda_t, w_t, v_t, x_{t+1} z_{t+1}, \theta_{t+1}, \lambda_{t+1}, w_{t+1}, v_{t+1}) \in \mathbf{K}_w$$

and

$$(x_t, z_t, \theta_t, \lambda_t, w_t, v_t,) \in \hat{\mathbf{K}}_w$$

for all $t \in \{0, 1, \ldots\}$. Hence, by (53) and (54),

$$\begin{aligned} V(x_{t+1}, z_{t+1}, \theta_{t+1}, \lambda_{t+1}, w_{t+1}, v_{t+1}) - V(x_t, z_t, \theta_t, \lambda_t, w_t, v_t) \\ \leq -\|\hat{y}_t\|_2^2 + \alpha_w \|w_t\|_2^2 + \alpha_v \|v_t\|_2^2 \end{aligned} \tag{61}$$

and

$$V(x_t, z_t, \theta_t, \lambda_t) \geq 0$$

for all $t \in \{0, 1, \ldots\}$. Iterating (61) we obtain

$$\begin{aligned} &\sum_{t=0}^{T-1} \|\hat{y}_t\|_2^2 \leq -V(x_T, z_T, \theta_T, \lambda_T, w_T, v_T) \\ &+ V(x_0, z_0, \theta_0, \lambda_0, w_0, v_0) + \sum_{t=0}^{T-1} \alpha_w \|w_t\|_2^2 + \alpha_v \|v_t\|_2^2 \\ &\leq V(x_0, z_0, \theta_0, \lambda_0, w_0, v_0) + \sum_{t=0}^{T-1} \alpha_w \|w_t\|_2^2 + \alpha_v \|v_t\|_2^2, \end{aligned}$$

where we have used the fact that $V(x_T, z_T, \theta_T, \lambda_T) \geq 0$ in the second inequality. Letting T tend to infinity gives the result. ■

References

[1] S. Boyd, L. E. Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory.* Society for Industrial and Applied Mathematics (SIAM).

[2] D. Gabay and B. Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & Mathematics with Applications*, 2(1):17–40, 1976.

[3] T. Goldstein, B. ODonoghue, and S. Setzer. Fast alternating direction optimization methods. *UCLA Computational and Applied Mathematics Report*, pages 12–35, 2012.

[4] J. H. Hours and C. N. Jones. A parametric multi-convex splitting technique with application to real-time nmpc. In *53st IEEE Conference on Decision and Control (CDC 2014)*, 2014.

[5] M. Korda and C. N. Jones. Certification of fixed computation time first-order optimization-based controllers for a class of nonlinear dynamical systems. In *American Control Conference*, 2013.

[6] J. B. Lasserre. *Moments, Positive Polynomials and Their Applications,*. Imperial College Press, first edition, 2009.

[7] J. B. Lasserre. On representations of the feasible set in convex optimization. *Optimization Letters*, 4(1):1–5, 2010.

[8] J. Löfberg. Yalmip : A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[9] Y. Nesterov. *Introductory lectures on convex optimization: A basic course.* Springer, 2004.

[10] A. Papachristodoulou, J. Anderson, S. P. G. Valmorbida, P. Seiler, and P. A. Parrilo. *SOSTOOLS: Sum of squares optimization toolbox for MATLAB*, 2013.

[11] P. A. Parrilo. Semidefinite programming relaxations for semialgebraic problems. *Mathematical programming*, 96(2):293–320, 2003.

[12] F. Permenter and P. Parrilo. Partial facial reduction: Simplified, equivalent semidefinite programs via approximations of the positive semidefinite cone. *Arxiv preprint arXiv:0905.3447*, 2014.

[13] D. W. Peterson. A review of constraint qualifications in finite-dimensional spaces. *SIAM Review*, 15(3):639–654, 1973.

[14] I. Pólik, T. Terlaky, and Y. Zinchenko. Sedumi: a package for conic optimization. In *IMA workshop on Optimization and Control*, 2007.

[15] J. A. Primbs. The analysis of optimization based controllers. *Automatica*, 37(6):933–938, 2001.

[16] P. Seiler. SOSOPT: A toolbox for polynomial optimization. *University of Minnesota*, 2010.

[17] B. Sturmfels. Polynomial equations and convex polytopes. American Mathematical Monthly. *American Mathematical Monthly*, 105(10):907–922, 1998.

[18] P. Tseng. Applications of a splitting algorithm to decomposition in convex programming and variational inequalities. *SIAM Journal on Control and Optimization*, 29(1):119–138, 1991.

[19] M. Yamashita, K. Fujisawa, and M. Kojima. SDPARA: Semidefinite programming algorithm parallel version. *Parallel Computing*, 29(8):1053–1067, 2003.

[20] X.-Y. Zhao, D. Sun, and K.-C. Toh. A newton-cg augmented lagrangian method for semidefinite programming. *SIAM Journal on Optimization*, 20(4):1737–1765, 2010.