Controls Applications of Sum of Squares Programming

Zachary Jarvis-Wloszek, Ryan Feeley, Weehong Tan, Kunpeng Sun and Andrew Packard

Department of Mechanical Engineering, University of California, Berkeley
{zachary, rfeeley, weehong, kpsun, pack}@jagger.me.berkeley.edu

Summary. We consider nonlinear systems with polynomial vector fields and pose two classes of system theoretic problems that may be solved by sum of squares programming. The first is disturbance analysis using three different norms to bound the reachable set. The second is the synthesis of a polynomial state feedback controller to enlarge the provable region of attraction. We also outline a variant of the state feedback synthesis for handling systems with input saturation. Both classes of problems are demonstrated using two-state nonlinear systems.

1 Introduction

Recent developments in sum of squares (SOS) programming [1, 2] have greatly extended the class of problems that can be solved with convex optimization. These results provide a general methodology to find formulations or relaxations, solvable by semidefinite programming, which address seemingly intractable nonconvex problems. Many of the problems that are amenable to SOS programming relate to polynomial optimization or algebraic geometry and reach back to the original work on global lower bounds for polynomials. This work is collected and expanded upon in [3].

First, we define the basic tools needed to state the main theorem, the Positivstellensatz, which leads to the development of our results. We use this methodology to pose two classes of system theoretic problems for nonlinear systems with polynomial vector fields. The first class of problems is disturbance analysis, which we will show three different ways of quantifying the effects of disturbances on polynomial systems:

1. bounding the reachable set subject to unit energy disturbance,
2. bounding the peak bounded disturbance that retains set invariance, and
3. bounding the induced $\mathcal{L}_2 \to \mathcal{L}_2$ gain.

The second class of problems is expanding a region of attraction with state feedback, and its variant for systems with input saturation. We will illustrate our methods of solving these problems by presenting two proof of concept numerical examples. The two classes of problems presented here is a selection of work done in [4] and [5].

2 Preliminaries

We often use the same letter to denote a signal (i.e. a function of time), as well as the possible values that the signal may take on at any time. We hope this abuse of notation will not confuse the reader.

2.1 Polynomial Definitions

Definition 1 (Monomials). *A **Monomial** m_α in n variables is a function defined as $m_\alpha(x) = x^\alpha := x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$ for $\alpha \in \mathbb{Z}_+^n$. The degree of a monomial is defined,* $\deg m_\alpha := \sum_{i=1}^n \alpha_i$.

Definition 2 (Polynomials). *A **Polynomial** f in n variables is a finite linear combination of monomials,*

$$f := \sum_\alpha c_\alpha m_\alpha = \sum_\alpha c_\alpha x^\alpha$$

with $c_\alpha \in \mathbb{R}$. Define $\mathcal{R}_n$ to be the set of all polynomials in n variables. The degree of f is defined as $\deg f := \max_\alpha \deg m_\alpha$ *(provided the associated c_α is non-zero).*

Additionally we define Σ_n to be the set of sum of squares (SOS) polynomials in n variables.

$$\Sigma_n := \left\{ p \in \mathcal{R}_n \;\middle|\; p = \sum_{i=1}^t f_i^2 \ ,\ f_i \in \mathcal{R}_n,\, i = 1, \ldots, t \right\} .$$

Obviously if $p \in \Sigma_n$, then $p(x) \geq 0 \ \forall x \in \mathbb{R}^n$.

It is interesting to note that there are polynomials that are positive semidefinite (PSD) that are not sum of squares. In general, there are only three combinations of number of variables and degree such that the set of SOS polynomials is equivalent to the set of positive semidefinite ones, namely, $n = 2$; $d = 2$; and $n = 3$ with $d = 4$. This result dates to Hilbert and is related to his 17th problem.

2.2 Positivstellensatz

In the section we define concepts to state a central theorem from real algebraic geometry, the Positivstellensatz, which we will hereafter refer to as the P-satz. This is a powerful theorem which generalizes many known results. For example, applying the P-satz, it is possible to derive the $\mathcal{S}$-procedure by carefully picking the free parameters, as will be shown in Sect. 2.4.

Definition 3. *Given* $\{g_1, \ldots, g_t\} \in \mathcal{R}_n$*, the* ***Multiplicative Monoid*** *generated by* g_j*'s is the set of all finite products of* g_j*'s, including 1 (i.e. the empty product). It is denoted as* $\mathcal{M}(g_1, \ldots, g_t)$*. For completeness define* $\mathcal{M}(\phi) := 1$.

An example: $\mathcal{M}(g_1, g_2) = \{g_1^{k_1} g_2^{k_2} \mid k_1, k_2 \in \mathbb{Z}_+\}$.

Definition 4. *Given* $\{f_1, \ldots, f_r\} \in \mathcal{R}_n$*, the* ***Cone*** *generated by* f_i*'s is*

$$\mathcal{P}(f_1, \ldots, f_r) := \left\{ s_0 + \sum_{i=1}^{l} s_i b_i \;\middle|\; l \in \mathbb{Z}_+, s_i \in \Sigma_n, b_i \in \mathcal{M}(f_1, \ldots, f_r) \right\} . \quad (1)$$

Note that if $s \in \Sigma_n$ and $f \in \mathcal{R}_n$, then $f^2 s \in \Sigma_n$ as well. This allows us to express a cone of $\{f_1, \ldots, f_r\}$ as a sum of 2^r terms. An example: $\mathcal{P}(f_1, f_2) = \{s_0 + s_1 f_1 + s_2 f_2 + s_3 f_1 f_2 \mid s_0, \ldots, s_3 \in \Sigma_n\}$.

Definition 5. *Given* $\{h_1, \ldots, h_u\} \in \mathcal{R}_n$*, the* ***Ideal*** *generated by* h_k*'s is*

$$\mathcal{I}(h_1, \ldots, h_u) := \left\{ \sum_{k=1}^{u} h_k p_k \;\middle|\; p_k \in \mathcal{R}_n \right\} .$$

With these definitions we can state the following theorem taken from [6, Theorem 4.2.2]

Theorem 1 (Positivstellensatz). *Given polynomials* $\{f_1, \ldots, f_r\}$*,* $\{g_1, \ldots, g_t\}$*, and* $\{h_1, \ldots, h_u\}$ *in* $\mathcal{R}_n$*, the following are equivalent:*

1. *The set*
$$\left\{ x \in \mathbb{R}^n \;\middle|\; \begin{array}{c} f_1(x) \geq 0, \ldots, f_r(x) \geq 0 \\ g_1(x) \neq 0, \ldots, g_t(x) \neq 0 \\ h_1(x) = 0, \ldots, h_u(x) = 0 \end{array} \right\}$$
is empty.
2. *There exist polynomials* $f \in \mathcal{P}(f_1, \ldots, f_r)$*,* $g \in \mathcal{M}(g_1, \ldots, g_t)$*,* $h \in \mathcal{I}(h_1, \ldots, h_u)$ *such that*
$$f + g^2 + h = 0 .$$

When there are only inequality constraints, and they describe a compact region, this theorem can be improved to reduce the number of free parameters [7], and with slightly stronger assumptions [8]. These results have been used to improve bounds on nonconvex polynomial optimization [2] and [9] highlighted a software package to do so.

2.3 SOS Programming

Sum of squares polynomials play an important role in the P-satz. Using a "Gram matrix" approach, Choi et al. [10] showed that $p \in \Sigma_n$ iff $\exists\ Q \succeq 0$ such that $p(x) = z^*(x) Q z(x)$, with $z(x)$ a vector of suitable monomials. Powers and Wörmann [11] proposed an algorithm to check if any $Q \succeq 0$ exists for a given $p \in \mathcal{R}_n$. Parrilo [1] showed that their algorithm is an LMI, and proved the following extension.

Theorem 2 (Parrilo). *Given a finite set* $\{p_i\}_{i=0}^m \in \mathcal{R}_n$*, the existence of* $\{a_i\}_{i=1}^m \in \mathbb{R}$ *such that*

$$p_0 + \sum_{i=1}^m a_i p_i \in \Sigma_n$$

is an LMI feasibility problem.

This theorem is useful since it allows one to answer questions like the following SOS programming example.

Example 1. Given $p_0, p_1 \in \mathcal{R}_n$, does there exist a $k \in \mathcal{R}_n$, of a given degree, such that

$$p_0 + kp_1 \in \Sigma_n \ . \tag{2}$$

To answer this question, write k as a linear combination of its monomials $\{m_j\}$, $k = \sum_{j=1}^s a_j m_j$. Rewrite (2) using this decomposition

$$p_0 + kp_1 = p_0 + \sum_{j=1}^s a_j (m_j p_1)$$

which since $(m_j p_1) \in \mathcal{R}_n$ is a feasibility problem that can be checked by Theorem 2.

A software package, SOSTOOLS, [12, 13], exists to aid in solving the LMIs that result from Theorem 2. This package as well as [9] use Sturm's SeDuMi semidefinite programming solver [14].

2.4 $\mathcal{S}$-Procedure

What does the $\mathcal{S}$-procedure look like in the P-satz formalism? Given symmetric $n \times n$ matrices $\{A_k\}_{k=0}^m$, the $\mathcal{S}$-procedure states: if there exist nonnegative scalars $\{\lambda_k\}_{k=1}^m$ such that $A_0 - \sum_{k=1}^m \lambda_k A_k \succeq 0$, then

$$\bigcap_{k=1}^m \left\{ x \in \mathbb{R}^n \mid x^T A_k x \geq 0 \right\} \subseteq \left\{ x \in \mathbb{R}^n \mid x^T A_0 x \geq 0 \right\} \ .$$

Written in P-satz form, the question becomes "is

$$\left\{ x \in \mathbb{R}^n \,\middle|\, \begin{array}{c} x^T A_1 x \geq 0, \ldots, x^T A_m x \geq 0, \\ -x^T A_0 x \geq 0,\, x^T A_0 x \neq 0 \end{array} \right\}$$

empty?" Certainly, if the λ_k exist, define $0 \preceq Q := A_0 - \sum_{k=1}^m \lambda_k A_k$. Further define SOS functions $s_0(x) := x^T Q x$, $s_{01} := \lambda_1, \ldots, s_{0m} := \lambda_m$. Note that

$$\begin{aligned} f &:= (-x^T A_0 x) s_0 + \textstyle\sum_{k=1}^m (-x^T A_0 x)(x^T A_k x) s_{0k} \\ &\in \mathcal{P}\left(x^T A_1 x, \ldots, x^T A_m x, -x^T A_0 x\right) \end{aligned}$$

and that $g := x^T A_0 x \in \mathcal{M}(x^T A_0 x)$. Substitution yields $f + g^2 = 0$ as desired. We will use this insight to make specific selections in the P-satz formulation of in Sects. 3 and 4. For the special case of $m = 1$, the converse of the $\mathcal{S}$-Procedure is also true [15, Sect. 2.6.3].

Using the tools of SOS programming and the P-satz, we can, after some simplifications, cast some control problems for systems with polynomial vector fields as tractable optimization problems. In the next two sections, we discuss two classes of problems that these techniques are applicable to.

3 Disturbance Analysis

In this section, we consider the local effects of external disturbances on polynomial systems. The following types of disturbance analysis are considered:

1. Reachable set bounds under unit energy disturbances
2. Set invariance under peak bounded disturbances
3. Bounding the induced $\mathcal{L}_2 \to \mathcal{L}_2$ gain

3.1 Reachable Set Bounds under Unit Energy Disturbances

Given a system of the form

$$\dot{x} = f(x) + g_w(x)w \tag{3}$$

with $x(t) \in \mathbb{R}^n$, $w(t) \in \mathbb{R}^{n_w}$, $f \in \mathcal{R}_n^n$, $f(0) = 0$, and $g_w \in \mathcal{R}_n^{n \times n_w}$. We want to compute a bound on the set of points $x(T)$ that are reachable from $x(0) = 0$ under (3), provided the disturbance satisfies $\int_0^T w(t)^* w(t)\, dt \leq 1$, $T \geq 0$.

A similar problem is considered in [16], where real quantifier elimination is used to calculate the exact reachable set for a larger class of dynamical systems. Our approach only considers convex relaxations of the exact problem, and as such requires less computation. A comparison of SOS programming and computational algebra is given in [17] for the case of polynomial minimization.

Following the Lyapunov-like argument in [15, Sect. 6.1.1], if we have a polynomial V such that

$$V(x) > 0 \text{ for all } x \in \mathbb{R}^n \backslash \{0\} \text{ with } V(0) = 0\ , \text{ and} \tag{4}$$

$$\frac{\partial V}{\partial x}(f(x) + g_w(x)w) \leq w^* w \quad \text{for all} \quad x \in \mathbb{R}^n,\ w \in \mathbb{R}^{n_w}\ , \tag{5}$$

then $\{x | V(x) \leq 1\}$ contains the set of points $x(T)$ that are reachable from $x(0) = 0$ for any w such that $\int_0^T w(t)^* w(t)\, dt \leq 1$, $T \geq 0$. We can see this by integrating the inequality in (5) from 0 to T, yielding

$$V(x(T)) - V(x(0)) = \int_0^T w(t)^* w(t)\, dt \leq 1\ .$$

Recalling $V(x(0)) = 0$, $x(T) \in \{x|V(x) \leq 1\}$. Furthermore, $x(\tau) \in \{x|V(x) \leq 1\}$ for all $\tau \in [0, T]$, allowing us to relax the inequality in (5) to

$$\frac{\partial V}{\partial x}(f(x) + g_w(x)w) \leq w^*w \quad \forall x \in \{x \,|\, V(x) \leq 1\},\ \forall w \in \mathbb{R}^{n_w} \,.$$

To bound the reachable set, we require a V satisfying these conditions. Additionally, to achieve a useful bound, the level set $\{x|V(x) \leq 1\}$ should be as small as possible. This is accomplished by requiring $\{x|V(x) \leq 1\}$ to be contained in a variable sized region $P_\beta := \{x \in \mathbb{R}^n | p(x) \leq \beta\}$, for some positive definite p, and minimizing β under the constraint that we can find a V satisfying (4) and (5). Restricting V to be a polynomial with no constant term, so that $V(0) = 0$, we formulate the problem in the following way, leading to application of the P-satz.

$$\min_{V \in \mathcal{R}_n} \beta$$

such that

$$\{x \in \mathbb{R}^n \mid V(x) \leq 0, l_1(x) \neq 0\} \text{ is empty}\,, \tag{6}$$

$$\{x \in \mathbb{R}^n \mid V(x) \leq 1, p(x) \geq \beta, p(x) \neq \beta\} \text{ is empty}\,, \tag{7}$$

$$\left\{ \begin{array}{c|c} x \in \mathbb{R}^n\,, & V(x) \leq 1, \\ & \frac{\partial V}{\partial x}(f(x) + g_w(x)w) \geq w^*w, \\ w \in \mathbb{R}^{n_w} & \frac{\partial V}{\partial x}(f(x) + g_w(x)w) \neq w^*w \end{array} \right\} \text{ is empty}\,. \tag{8}$$

where l_1 is some positive definite and SOS polynomial that replaces x in the non-polynomial constraint $x \neq 0$. The constraints (6) and (8) make V and $\dot{V}$ behave properly, while (7) allows that $\{x|V(x) \leq 1\} \subseteq P_\beta$.

Invoking the P-satz, constraints (6)–(8) are equivalent to the constraints in the following minimization.

$$\min \beta \quad \text{over} \quad \begin{array}{c} V \in \mathcal{R}_n\,, \quad s_1, \ldots, s_6 \in \Sigma_n \\ s_7, \ldots, s_{10} \in \Sigma_{n+n_w}\,, \quad k_1, k_2, k_3 \in \mathbb{Z}_+ \end{array}$$

such that

$$s_1 - V s_2 + l_1^{2k_1} = 0\,, \tag{9}$$

$$\begin{aligned} &s_3 + (1 - V)s_4 + (p - \beta)s_5\,, \\ &\quad + (1 - V)(p - \beta)s_6 + (p - \beta)^{2k_2} = 0\,, \end{aligned} \tag{10}$$

$$\begin{aligned} &s_7 + \left(\frac{\partial V}{\partial x}(f(x) + g_w(x)w) - w^*w\right) s_8 + (1 - V)s_9 \\ &\quad + (1 - V)\left(\frac{\partial V}{\partial x}(f(x) + g_w(x)w) - w^*w\right) s_{10}\,, \\ &\quad + \left(\frac{\partial V}{\partial x}(f(x) + g_w(x)w) - w^*w\right)^{2k_3} = 0\,. \end{aligned} \tag{11}$$

Conditions (9)–(11) cannot be directly checked using SOS programming methods. Therefore we specify convenient values for some of the s_i's and k_j's. We also restrict the degree of V and the remaining s_i's. Consequently (9)–(11) become only sufficient for (6)–(8).

Picking any of the $k_j = 0$, can prevent feasibility. Thus we set all of the k_j's equal to the next smallest value, 1. If we pick $s_2 = l_1$ and $s_1 = \hat{s}_1 l_1$, then (9) looks like the form used to show positive definiteness of a Lyapunov function in [18]. Additionally, if we pick $s_7 = s_9 = 0$, and realize that $\frac{\partial V}{\partial x}(f(x) + g_w(x)w) - w^*w$ is not the zero polynomial, we can write (11) in the form of a "generalized" $\mathcal{S}$-procedure. These choices leave the following problem:

$$\min \beta \quad \text{over } V \in \mathcal{R}_n, \quad s_4, s_5, s_6 \in \Sigma_n, \quad s_{10} \in \Sigma_{n+n_w}$$

such that

$$V - l_1 \in \Sigma_n \ , \tag{12}$$

$$-\Big((1-V)s_4 + (p-\beta)s_5 + (1-V)(p-\beta)s_6 + (p-\beta)^2\Big) \in \Sigma_n \ , \tag{13}$$

$$-\left((1-V)s_{10} + \left(\frac{\partial V}{\partial x}(f(x) + g_w(x)w) - w^*w\right)\right) \in \Sigma_{n+n_w} \ . \tag{14}$$

where (12) ensures the positive definiteness of V, (13) establishes $\{x|V(x) \leq 1\} \subseteq P_\beta$, and (14) constrains $\dot{V} \leq w^*w$ on $\{x|V(x) \leq 1\}$.

Note that some of the decision polynomials enter the constraints in a bilinear form, which SOS programming cannot handle directly. For example, in (13), there are bilinear terms such as Vs_4 and Vs_6. Our approach is to hold one set of decision polynomials fixed while optimizing the other set, then switching over. This results in an iterative algorithm whereby at any step, the constraints (12)–(14) can be checked using SOS programming.

Before presenting the algorithm, two issues deserve mention. First, to use SOS programming, we must specify the maximum degree of V and the SOS polynomials s_i. To ensure (12)–(14) might be satisfied, the degree of the polynomials must satisfy

$$\begin{gathered} \deg V = \deg l_1 \ , \\ \max\{\deg(Vs_4); \deg(Vps_6)\} \geq \max\{\deg(ps_5); 2\deg p\} \ , \\ \deg s_{10} \geq \max\{\deg f; \deg(g_w w)\} - 1 \ . \end{gathered} \tag{15}$$

These constraints are a consequence of the nature of polynomials; e.g. a SOS polynomial of degree 2 cannot be greater than a SOS polynomial of degree 4 for all x.

The second issue is that the algorithm does not reliably find a feasible point $\{V, s_4, s_5, s_6, s_{10}, \beta\}$. Rather it can only improve upon one, by driving β smaller. As written, the user must supply an initial V_0 that is a component of some feasible point, though the other components can be determined with SDPs. *Given* a V_0 satisfying (12), an SDP can determine the existence of s_4, s_5, and s_6 to satisfy (13). Likewise, a separate SDP can determine the existence of s_{10} satisfying (14). Note that a "poor" choice of initial V_0 may render (13) and/or (14) unsatisfiable for any choice of $\beta, s_4, s_5, s_6, s_{10}$, although for a different V_0, (13) and (14) may be satisfied. Heuristics (based on linearizations) to find suitable initial V_0's are possible. However, once a feasible point $\{V, s_4, s_5, s_6, s_{10}, \beta\}$ is found, the optimization will remain feasible and β will be at least monotonically non-increasing with every step of the algorithm. Since we do not have a lower bound on β, we do not have a formal stopping criteria. Heuristics, such as β between each iteration of the algorithm improving by less than a specified tolerance, is used as our stopping criterion.

Iterative Bounding Procedure

Setup: Specify the maximum degree that will be considered for both V and the s_i's, observing the constraints in (15). Set $l_1 = \epsilon \sum x_i^m$ for some small $\epsilon > 0$, and m is the maximum degree of V. Each step of the iteration, which is indexed by i, consists of three substeps, the first two subject to constraints (12)–(14). To begin the iteration, choose a V_0, initialize $V^{(i=0)} = V_0$ and the iteration index $i = 1$, and proceed to step 1.

1. **SOS Optimization:**
   Minimize β over s_4, s_5, s_6, and s_{10}, with $V = V^{(i-1)}$ fixed, to obtain $s_4^{(i)}$, $s_6^{(i)}$, and $s_{10}^{(i)}$.
2. **Lyapunov Function Synthesis:**
   Minimize β over s_5 and V, with $s_4 = s_4^{(i)}$, $s_6 = s_6^{(i)}$, and $s_{10} = s_{10}^{(i)}$ fixed, to obtain $V^{(i)}$ and $\beta^{(i)}$.
3. **Stopping Criterion:**
   If $\beta^{(i)} - \beta^{(i-1)}$ is less than a specified tolerance, conclude the iteration, otherwise increment i and return to substep 1.

In (13), β is multiplied by polynomials we are searching over. Therefore we minimize β in substeps 1 and 2 using a line search.

If we restrict ourselves to linear dynamics, $\dot{x} = Ax + B_w w$, and quadratic Lyapunov functions, $V(x) = x^* P x$, then (12) becomes $P \succ 0$, and with $s_{10} = 0$, (14) becomes

$$\begin{bmatrix} A^*P + PA & PB_w \\ B_w^* P & -I \end{bmatrix} \preceq 0 \; .$$

Thus (12) and (14) generalize the LMI in [15, Sect. 6.1.1].

3.2 Set Invariance under Peak Bounded Disturbances

Considering again a polynomial system subject to disturbances as in (3),

$$\dot{x} = f(x) + g_w(x)w \, .$$

We can now look at bounding the maximum peak disturbance value such that a given set remains invariant under these bounded disturbances and the action of the system's dynamics.

Let the peak of a signal w be bounded by

$$\|w\|_\infty := \sup_t |w(t)| \leq \sqrt{\gamma}$$

and define the invariant set as

$$\Omega_1 := \{x \in \mathbb{R}^n | V(x) \leq 1\}$$

for some fixed $V \in \mathcal{R}_n$, positive definite. We know that if $\frac{\partial V}{\partial x}(f(x)+g_w(x)w) \leq 0$ on the boundary of Ω_1 for all w meeting the peak bound, then the flow of the system from any point in Ω_1 cannot ever leave Ω_1, which makes it invariant. In set containment terms we can write this relationship as

$$\begin{aligned} &\{x \in \mathbb{R}^n, w \in \mathbb{R}^{n_w} | V(x) = 1\} \cap \{x \in \mathbb{R}^n, w \in \mathbb{R}^{n_w} | w^* w \leq \gamma\} \\ &\qquad \subseteq \left\{ x \in \mathbb{R}^n, w \in \mathbb{R}^{n_w} \,\middle|\, \frac{\partial V}{\partial x}(f(x) + g_w(x)w) \leq 0 \right\} \end{aligned} \tag{16}$$

which can be rewritten in set emptiness form as

$$\left\{ \begin{array}{l} x \in \mathbb{R}^n, \\ w \in \mathbb{R}^{n_w} \end{array} \,\middle|\, \begin{array}{l} V(x) - 1 = 0, \gamma - w^* w \geq 0, \\ \frac{\partial V}{\partial x}(f(x) + g_w(x)w) \geq 0, \\ \frac{\partial V}{\partial x}(f(x) + g_w(x)w) \neq 0 \end{array} \right\} = \phi$$

Employing the P-satz, this becomes

$$\begin{aligned} s_0 &+ s_1(\gamma - w^* w) + s_2 \tfrac{\partial V}{\partial x}(f(x) + g_w(x)w) \\ &+ s_3(\gamma - w^* w) \tfrac{\partial V}{\partial x}(f(x) + g_w(x)w) \\ &+ \left(\tfrac{\partial V}{\partial x}(f(x) + g_w(x)w)\right)^{2k} + q(V-1) = 0 \end{aligned}$$

with $k \in \mathbb{Z}_+$, $q \in \mathcal{R}_{n+n_w}$ and $s_0, s_1, s_2, s_3 \in \Sigma_{n+n_w}$.

Using our standard approach of $k = 1$, we can write the following SOS constraint that guarantees invariance under bounded w,

$$\begin{aligned} &- s_1(\gamma - w^* w) - s_2 \tfrac{\partial V}{\partial x}(f(x) + g_w(x)w) \\ &\quad - s_3(\gamma - w^* w) \tfrac{\partial V}{\partial x}(f(x) + g_w(x)w) \\ &\quad - \left(\tfrac{\partial V}{\partial x}(f(x) + g_w(x)w)\right)^2 - q(V-1) \in \Sigma_{n+n_w} \, . \end{aligned} \tag{17}$$

Notice that this SOS condition has terms that are not linear in the monomials of V, and thus there is no way to use our convex optimization approach to adjust V while checking this condition. Since (17) in linear in γ we can search for the maximum peak disturbance for which the set is invariant, by searching over q and the s_i's to maximize γ subject to (17). We will need to have the following degree relationship hold to make (17) possibly feasible

$$\max\left\{\deg(s_1)+2,\ \deg(s_2\tfrac{\partial V}{\partial x}(f(x)+g_w(x)w)),\ \deg(qV)\right\}$$
$$\geq \max\left\{\deg(s_3\tfrac{\partial V}{\partial x}(f(x)+g_w(x)w)))+2,\ 2\deg(\tfrac{\partial V}{\partial x}(f(x)+g_w(x)w)))\right\}\ .$$

If we set $x(0)=0$, then the invariant set Ω_1 bounds the system's reachable set under disturbances with peak less than γ. This bound is similar, but less stringent, than the bound for linear systems given in [15].

The constraint in (17) can result in searching for polynomials with many coefficients. We can reduce the degree of this constraint by setting $q = \left(\frac{\partial V}{\partial x}(f(x)+g_w(x)w)\right)^2 \hat{q}$ and $s_i = \left(\frac{\partial V}{\partial x}(f(x)+g_w(x)w)\right)^2 \hat{s}_i$ for $i=1,2,3$. This allows us to factor out a $\left(\frac{\partial V}{\partial x}(f(x)+g_w(x)w)\right)^2$ term to get the following sufficient condition:

$$\begin{aligned}&-\hat{s}_1(\gamma - w^*w) - \hat{s}_2\tfrac{\partial V}{\partial x}(f(x)+g_w(x)w)\\ &\quad -\hat{s}_3(\gamma - w^*w)\tfrac{\partial V}{\partial x}(f(x)+g_w(x)w) - 1 - \hat{q}(V-1) \in \Sigma_{n+n_w}\ .\end{aligned} \tag{18}$$

For this simplified constraint (18), the polynomials must satisify this degree relationship:

$$\begin{aligned}\max\left\{\deg(\hat{s}_1)+2,\ \deg(\hat{s}_2\tfrac{\partial V}{\partial x}(f(x)+g_w(x)w)),\ \deg(\hat{q}V)\right\}&\\ \geq \deg\left(\hat{s}_3\tfrac{\partial V}{\partial x}(f(x)+g_w(x)w))\right)+2\ .&\end{aligned} \tag{19}$$

Effect of $\|w\|_\infty$ on $\|x\|_\infty$

Using the bounded peak disturbances techniques above to find a bound for the largest disturbance peak value for which Ω_1 is invariant, we can then bound the peak size of the system's state to get a relationship that is similar to the induced $\mathcal{L}_\infty \to \mathcal{L}_\infty$ norm from disturbance to state for this invariant set.

For a given V, we solve the optimization to find the largest γ such that (17) is feasible. Then we can bound the size of the state by optimizing to find the smallest α such that

$$\Omega_1 = \{x \in \mathbb{R}^n \mid V(x) \leq 1\} \subseteq \{x \in \mathbb{R}^n \mid x^*x \leq \alpha\}$$

This containment constraint is easily solved with a generalized $\mathcal{S}$-procedure following from Sect. 2.4. From this point we know that the following implication holds

$$\forall x(0) \in \Omega_1\ , \text{ and } \ \|w\|_\infty \leq \sqrt{\gamma} \quad \Rightarrow \quad \|x\|_\infty \leq \sqrt{\alpha}\ ,$$

which provides our induced norm-like bound.

3.3 Bounding the Induced $\mathcal{L}_2 \to \mathcal{L}_2$ Gain

Consider the disturbance driven system with outputs,

$$\begin{aligned}\dot{x} &= f(x) + g_w(x)w \\ y &= h(x)\end{aligned}$$

with $x(t) \in \mathbb{R}^n$, $w(t) \in \mathbb{R}^{n_w}$, $y(t) \in \mathbb{R}^p$, $f \in \mathcal{R}_n^n$, $f(0) = 0$, $g_w \in \mathcal{R}_n^{n \times n_w}$, and $h \in \mathcal{R}_n^p$ with $h(0) = 0$.

For a region, $\Omega_1 = \{x \in \mathbb{R}^n | V(x) \le 1\}$ as in Sect. 3.2, that is invariant under disturbances with $\|w\|_\infty \le \sqrt{\gamma}$, we can bound the induced $\mathcal{L}_2 \to \mathcal{L}_2$ gain from w to y on this invariant set by finding a positive definite $H \in \mathcal{R}_n$ and $\beta \ge 0$ such that the following set containment holds

$$\begin{aligned}&\{x \in \mathbb{R}^n, w \in \mathbb{R}^{n_w} | w^* w \le \gamma\} \cap \{x \in \mathbb{R}^n, w \in \mathbb{R}^{n_w} | V(x) \le 1\} \\ &\subseteq \left\{x \in \mathbb{R}^n, w \in \mathbb{R}^{n_w} \,\middle|\, \tfrac{\partial H}{\partial x}(f(x) + g_w(x)w) + h(x)^* h(x) - \beta w^* w \le 0 \right\}\end{aligned} \tag{20}$$

If we can find a β, H pair to make (20) hold, then we can follow the steps from Sect. 3.1 to show that

$$x(0) = 0 \quad \text{and} \quad \|w\|_\infty \le \sqrt{\gamma} \;\Rightarrow\; \frac{\|y\|_2}{\|w\|_2} \le \sqrt{\beta}\,.$$

We can search for the tightest bound on the induced norm by employing a generalized $\mathcal{S}$-procedure to satisfy (20) and solving the following optimization

$$\begin{aligned}&\min_{H \in \mathcal{R}_n} \beta \qquad \text{s.t.} \\ &H - l \in \Sigma_n\,, \\ &-\left(\tfrac{\partial H}{\partial x}(f(x) + g_w(x)w) + h(x)^* h(x) - \beta w^* w\right) \\ &\qquad\qquad -s_1(\gamma - w^* w) - s_2(1 - V) \in \Sigma_{n+n_w}\end{aligned} \tag{21}$$

with $s_1, s_2 \in \Sigma_{n+n_w}$ and $l \in \Sigma_n$, positive definite.

In an effort to make the optimization (21) feasible we will pick the degrees of s_1 and s_2 so that

$$\begin{aligned}\deg(s_1) + 2 &\ge \deg\left(\tfrac{\partial H}{\partial x}(f(x) + g_w(x)w) + h^* h\right) \quad \text{and} \\ \deg(s_2 V) &\ge \deg\left(\tfrac{\partial H}{\partial x}(f(x) + g_w(x)w) + h^* h\right)\;.\end{aligned}$$

3.4 Disturbance Analysis Example

Consider the following nonlinear system

$$\begin{aligned}\dot{x}_1 &= -x_1 + x_2 - x_1 x_2^2 \\ \dot{x}_2 &= -x_2 - x_1^2 x_2 + w \\ y &= [x_1\ x_2]^T\end{aligned} \tag{22}$$

with $x(t) \in \mathbb{R}^2$ and $w(t) \in \mathbb{R}$. Given $p(x) = 8x_1^2 - 8x_1x_2 + 4x_2^2$, we would like to determine the smallest level set $P_\beta := \{x \in \mathbb{R}^2 \,|\, p(x) \le \beta\}$ that contains all possible system trajectories for $t \le T$ starting from $x(0) = 0$ with $\int_0^T w^2\,dt \le R$, where R is a given constant. Employing the algorithm in Sect. 3.1, we fix $s_5 = 1$ and $s_6 = 0$ to eliminate the need for a line search in each substep. We set the maximum degree of V, s_4 and s_{10} all to be of degree 4 and initialized the algorithm with $V_0(x) = x_1^2 + x_2^2$. Figure 1 shows the algorithm's progress in reducing β versus iteration number as well as the trade off between R and β. The insert shows the monotonically decreasing behavior of our algorithm for $R = 1$, and after 10 iterations, β is reduced to 1.08, which is a large improvement over the first iteration bound of $\beta = 10.79$. For increasing values of disturbance energy R, the size of the reachable set increases, which is expected.

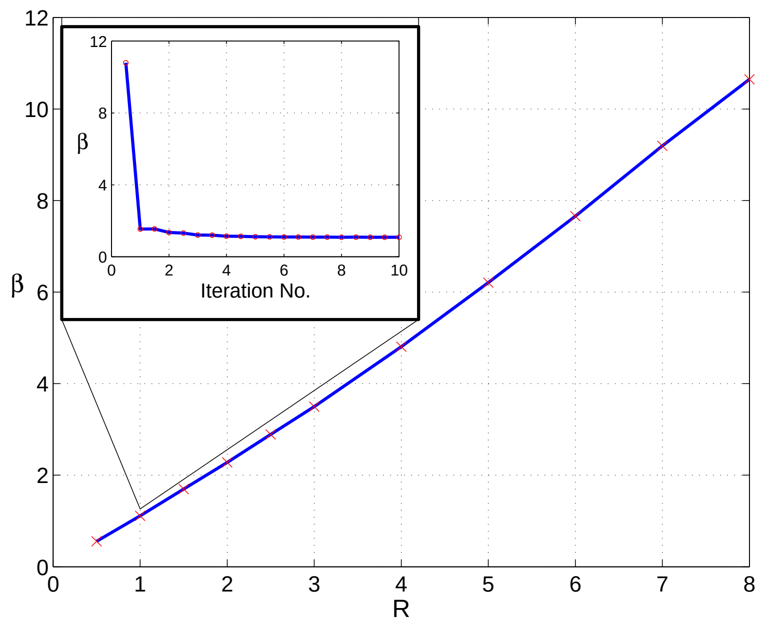

Fig. 1. Insert: Algorithm's progress for R=1, Main: Trade off between R and β

Using the Lyapunov function V found in the reachable set analysis for $R = 1$, we can bound the peak disturbance such that the set $\Omega_1 = \{x \in \mathbb{R}^2 \mid V(x) \le 1\}$ remains invariant. Using the optimization in (18), we get $\|w\|_\infty \le \sqrt{\gamma} = 0.642$ by choosing the degree of $\hat{s}_1, \hat{s}_2, \hat{s}_3$ and $\hat{p}$ to be 6, 2, 0 and 4 respectively. If we start from $x(0) \in \Omega_1$ and have $\|w\|_\infty \le 0.642$, then $\|x\|_\infty \le \sqrt{\alpha} = 0.784$. We can also bound the induced $\mathcal{L}_2 \to \mathcal{L}_2$ disturbance to state gain for this system. The maximum degree of H, s_1, and s_2 are chosen to be 2, 0 and 2 respectively. Using (21), we get $\frac{\|x\|_2}{\|w\|_2} \le 1.41$ if we start from $x(0) = 0$, and as long as $\|w\|_\infty \le 0.642$.

4 Expanding a Region of Attraction with State Feedback

Given a system of the form

$$\dot{x} = f(x) + g(x)u \tag{23}$$

with $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}$, and f, g n-vectors of elements of $\mathcal{R}_n$ such that $f(0) = 0$, we would like to synthesize a state feedback controller $u = K(x)$ with $K \in \mathcal{R}_n$ that enlarges the set of points that we can show are attracted to the fixed point at the origin.

We define a variable sized region as $P_\beta := \{x \in \mathbb{R}^n | p(x) \leq \beta\}$, for some given positive definite p. We then expand the provable region of attraction by maximizing β while requiring that all of the points in P_β converge to the origin under the controller K. Using a Lyapunov argument, every point in P_β will converge asymptotically to the origin if there exists $K, V \in \mathcal{R}_n$ such that the following hold:

$$V(x) > 0 \text{ for all } x \in \mathbb{R}^n \backslash \{0\} \text{ and } V(0) = 0\ , \tag{24}$$

$$\{x \in \mathbb{R}^n \mid p(x) \leq \beta\} \subseteq \{x \in \mathbb{R}^n \mid V(x) \leq 1\}\ , \tag{25}$$

$$\begin{aligned}&\{x \in \mathbb{R}^n \mid V(x) \leq 1\} \backslash \{0\} \subseteq \\ &\qquad \left\{ x \in \mathbb{R}^n \;\middle|\; \frac{\partial V}{\partial x}(f(x) + g(x)K(x)) < 0 \right\}\ .\end{aligned} \tag{26}$$

These conditions show that V is positive definite, P_β is contained in a level set of V, and $\frac{dV}{dt}$ is strictly negative on all the points contained in the level set aside from $x = 0$.

The condition that $V(0) = 0$ is satisfied by setting the constant term to zero. Enlarging the region of attraction subject to the preceding requirements can be cast into the following form which is amenable to the P-satz.

$$\max_{K, V \in \mathcal{R}_n} \beta$$

such that

$$\{x \in \mathbb{R}^n \mid V(x) \leq 0,\ l_1(x) \neq 0\} \text{ is empty}\ , \tag{27}$$

$$\{x \in \mathbb{R}^n \mid p(x) \leq \beta,\ V(x) \geq 1,\ V(x) \neq 1\} \text{ is empty}\ , \tag{28}$$

$$\left\{ x \in \mathbb{R}^n \;\middle|\; \begin{array}{c} V(x) \leq 1,\ l_2(x) \neq 0, \\ \frac{\partial V}{\partial x}\Big(f(x) + g(x)K(x)\Big) \geq 0 \end{array} \right\} \text{ is empty}\ . \tag{29}$$

where l_1, l_2 are fixed positive definite and SOS polynomials which replace the non-polynomial constraints $x \neq 0$ in (24) and (26).

Applying the P-satz, the region maximization problem with constraints (27)–(29) is equivalent to

$$\max \beta \quad \text{over} \quad \begin{array}{c} K, V \in \mathcal{R}_n \quad k_1, k_2, k_3 \in \mathbb{Z}_+ \\ s_1, \ldots, s_{10} \in \Sigma_n \end{array}$$

such that

$$s_1 - V s_2 + l_1^{2k_1} = 0 \; , \tag{30}$$

$$s_3 + (\beta - p)s_4 + (V-1)s_5 + (\beta - p)(V-1)s_6 + (V-1)^{2k_2} = 0 \; , \tag{31}$$

$$s_7 + (1-V)s_8 + (\frac{\partial V}{\partial x}(f + gK))s_9 + (1-V)(\frac{\partial V}{\partial x}(f+gK))s_{10} + l_2^{2k_3} = 0 \; . \tag{32}$$

We cannot check (30)–(32) using SOS programming methods, so we will have to pick values for some of the s_i's and k_j's. We set $k_1 = k_2 = k_3 = 1$ and pick $s_2 = l_1$ and $s_1 = \hat{s}_1 l_1$ to simplify (30). Equation (31) has a $(V-1)^{2k_2}$ term which we can not directly optimize over using SOS programming, so we cast this constraint as an $\mathcal{S}$-procedure (see Sect. 2.4). This is done by setting $s_3 = s_4 = 0$, $k_2 = 1$, and factoring out a $(V-1)$ term. To simplify (32) we set $s_{10} = 0$ and factor out l_2, leaving the sufficient conditions below,

$$\max \beta \quad \text{over } K, V \in \mathcal{R}_n \quad s_6, s_8, s_9 \in \Sigma_n$$

such that

$$V - l_1 \in \Sigma_n \; , \tag{33}$$

$$-\Big((\beta - p)s_6 + (V-1)\Big) \in \Sigma_n \; , \tag{34}$$

$$-\left((1-V)s_8 + \frac{\partial V}{\partial x}(f+gK)s_9 + l_2\right) \in \Sigma_n \; . \tag{35}$$

Again, the decision polynomials do not enter the constraints linearly, so we employ an iterative algorithm to solve this maximization. A slight modification to (35) is needed because for a given Lyapunov candidate function V, searching over K does not affect β at all. An intermediate variable, α, is introduced to (35) so that we maximize the level set of $\{x \,|\, V(x) \leq \alpha\}$ that is contractively invariant under K and use α to scale V and l_2. We will elaborate more in the control design algorithm.

To initialize the algorithm, set V_0 to be a control Lyapunov function (CLF) of the linearized system. Since V_0 is a CLF, (33) is automatically satisfied and (35) is easily satisfied by scaling V_0. Constraint (34) is also satisfied for sufficiently small β. As such, if we can find a CLF for the linearized system, we would have a feasible starting point for our algorithm. Otherwise, the algorithm might fail on the first iteration.

For reasons highlighted in Sect. 3.1, the maximum degree of V, K, l_1, l_2, and the s_i's must satisfy the following constraints:

$$
\begin{aligned}
\deg V &= \deg l_1 \ , \\
\deg (p s_6) &\geq \deg V \ , \\
\deg s_8 &\geq \max\{\deg(f s_9); \deg(gK s_9)\} - 1 \ , \\
\deg (V s_8) &= \deg l_2 \ .
\end{aligned} \tag{36}
$$

Control Design Algorithm
Setup: Specify the maximum degree that will be considered for both V and the s_i's. Set $l_1 = \epsilon \sum x_i^m$ for some small $\epsilon > 0$, and m is the maximum degree of V. Each step of the iteration, indexed by i, consists of three substeps, two of which also involve iterations. These inner iterations will be indexed by j. To begin the iteration, choose a V_0 that is a CLF of the linearized system, and initialize $V^{(i=0)} = V_0$ and $s_9^{(i=0)} = 1$. Also, set $l_2^{(i=0)} = \epsilon \sum x_i^q$, where q is the maximum degree of $(V s_8)$. Now set the outer iteration index $i = 1$ and proceed to step 1.

1. **Controller Synthesis:**
   Set $V = V^{(i-1)}$, $s_9^{(j=0)} = s_9^{(i-1)}$, and the inner iteration index $j = 1$. In substeps 1a and 1b, solve the following optimization problem:

$$
\begin{aligned}
&\max \alpha \quad \text{over } K \in \mathcal{R}_n, \quad s_8, s_9 \in \Sigma_n \quad \text{such that} \\
&-\left((\alpha - V)s_8 + \frac{\partial V}{\partial x}(f + gK)s_9 + l_2\right) \in \Sigma_n \ .
\end{aligned} \tag{37}
$$

   (a) Maximize α over s_8, K, with $s_9 = s_9^{(j-1)}$ fixed, to obtain $K^{(j)}$.
   (b) Maximize α over s_8, s_9, with $K = K^{(j)}$ fixed, to obtain $s_9^{(j)}$ and $\alpha^{(j)}$.
   (c) If $\alpha^{(j)} - \alpha^{(j-1)}$ is less than a specified tolerance, set $s_8^{(i)} = s_8^{(j)}$, $s_9^{(i)} = s_9^{(j)}$, $l_2^{(i)} = l_2^{(i-1)}/\alpha^{(j)}$, and $\alpha^{(i)} = \alpha^{(j)}$ and continue to step 2. Otherwise increment j and return to 1a.
2. **Lyapunov Function Synthesis:**
   Set $V^{(j=0)} = V^{(i-1)}/\alpha^{(i)}$ and the inner iteration index $j = 1$. Hold $s_8 = s_8^{(i)}$, $s_9 = s_9^{(i)}$, and $l_2 = l_2^{(i)}$ fixed.
   (a) Maximize β over s_6, with $V = V^{(j-1)}$ fixed, subject to (34) to obtain $s_6^{(j)}$. i.e.

$$
\begin{aligned}
&\max \beta \qquad \text{over } s_6 \in \Sigma_n \quad \text{such that} \\
&-\Big((\beta - p)s_6 + (V - 1)\Big) \in \Sigma_n \ .
\end{aligned}
$$

   (b) Maximize β over V, with $s_6 = s_6^{(j)}$ fixed, subject to (33)–(35) to obtain $V^{(j)}$ and $\beta^{(j)}$.

   (c) If $\beta^{(j)} - \beta^{(j-1)}$ is less than a specified tolerance, set $V^{(i)} = V^{(j)}$ and $\beta^{(i)} = \beta^{(j)}$ and continue to step 3. Otherwise increment j and return to 2a.
3. **Stopping Criterion:** If $\beta^{(i)} - \beta^{(i-1)}$ is less than a specified tolerance conclude the iterations, otherwise return to step 1.

As in Sect. 3.1, we use a line search to maximize α and β in the steps above.

4.1 Expanding the Region of Attraction for Systems with Input Saturation

Given a system of the form

$$\dot{x} = f(x) + g(x)\,\mathrm{sat}(u) \tag{38}$$

where

$$\mathrm{sat}(u) := \begin{cases} u \text{ if } |u| \leq 1 \\ 1 \text{ if } u > 1 \\ -1 \text{ if } u < -1 \end{cases}$$

with $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}$, and f, g n-vectors of elements of $\mathcal{R}_n$ such that $f(0) = 0$, we would like to synthesize a state feedback controller $u = K(x)$ with $K \in \mathcal{R}_n$ to enlarge the set of points which are attracted to the origin.

Again, we define the region to expand as $P_\beta := \{x \in \mathbb{R}^n | p(x) \leq \beta\}$, for some given positive definite p. We want to design state feedback controller $K(x)$ to maximize β such that the P_β is a domain of attraction and $|u| \leq 1$. This is accomplished by appending two conditions to (24)–(26):

$$\{x \in \mathbb{R}^n \mid V(x) \leq 1\} \subseteq \{x \in \mathbb{R}^n \mid K(x) \leq 1\} \;, \tag{39}$$

$$\{x \in \mathbb{R}^n \mid V(x) \leq 1\} \subseteq \{x \in \mathbb{R}^n \mid K(x) \geq -1\} \;. \tag{40}$$

These two equations ensure that $|u| = |K(x)| \leq 1$ for all x in the contractively invariant set $\{x \in \mathbb{R}^n | V(x) \leq 1\}$, so the control action will not hit saturation.

Following the procedure in Sect. 4, we will obtain constraints (33)–(35). Additionally due to the saturation, we have

$$\Big((1 - K) - (1 - V)s_{10}\Big) \in \Sigma_n \;, \tag{41}$$

$$\Big((1 + K) - (1 - V)s_{11}\Big) \in \Sigma_n \;. \tag{42}$$

The control design algorithm for this problem is similar to that proposed in Sect. 4, with the inclusions of the two additional constraints (41) and (42).

4.2 State Feedback Example

Consider the following nonlinear system:

$$\begin{aligned}\dot{x}_1 &= u \\ \dot{x}_2 &= -x_1 + \tfrac{1}{6}x_1^3 - u\end{aligned} \tag{43}$$

with $x(t) \in \mathbb{R}^2$ and $u(t) \in \mathbb{R}$. We are interested in enlarging the domain of attraction described by the level set $P_\beta := \{x \in \mathbb{R}^2 \,|\, p(x) \leq \beta\}$, where $p(x) = \frac{1}{6}x_1^2 + \frac{1}{6}x_1x_2 + \frac{1}{12}x_2^2$, through state feedback. Using the algorithm in Sect. 4, we start with randomized $V_0(x)$ that are CLFs of the linearized system. We set the maximum degrees of V, K, s_6, s_8 and s_9 to 2, 1, 2, 2, and 0 respectively.

Figure 2 shows the progress of β with iteration number for 10 random V_0. Out of these 10 random V_0, the largest β achieved is 54.65. Figure 3 shows the resulting domain of attraction for this case. The corresponding controller is $K = -145.94x_1 + 12.2517x_2$ and $V = 0.001(2.3856x_1^2 + 2.108x_1x_2 + 1.17x_2^2)$. Surprisingly, for higher orders of V and K, we have obtained smaller regions of attraction. This is likely due to the nonconvexity of the overall control design algorithm. Although each substep is optimal (i.e., convex), our iterative approach of breaking the algorithm into substeps is not.

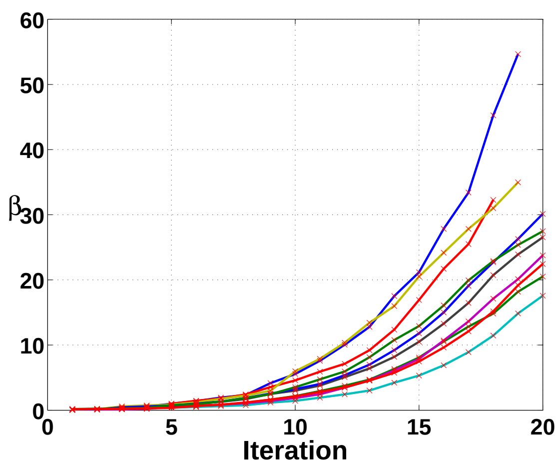

Fig. 2. β vs. iteration no. for various V_0

We can also analyze the disturbance rejection properties of this controller when the disturbances enter the system additively in the control channel, i.e. $g_w(x) = g(x)$. Using the Lyapunov function V found in the state feedback design, we can bound the peak disturbance such that the set $\Omega_1 = \{x \in \mathbb{R}^2 \mid V(x) \leq 1\}$ remains invariant. Using the optimization in (18), we get $\|w\|_\infty \leq \sqrt{\gamma} = 31.62$ by choosing the degree of $\hat{s}_1, \hat{s}_2, \hat{s}_3$ and $\hat{p}$ to be 4, 2, 0 and 4 respectively. If we start with $x(0) \in \Omega_1$ and have $\|w\|_\infty \leq 31.62$, then

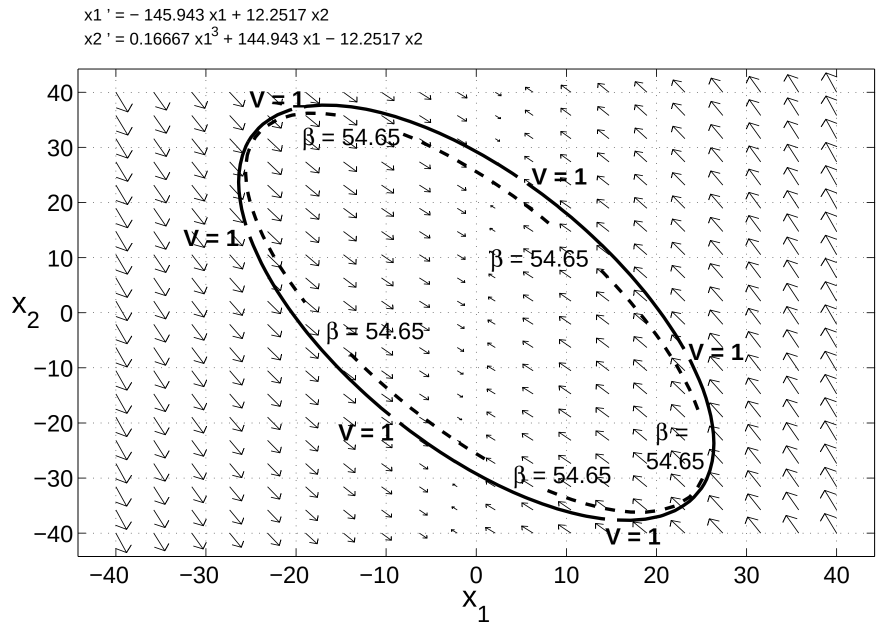

Fig. 3. Closed loop system's region of attraction

$\|x\|_\infty \leq \sqrt{\alpha} = 42.22$. We can also bound the induced $\mathcal{L}_2 \to \mathcal{L}_2$ disturbance to state gain for this system by setting $h(x) = [x_1\ x_2]^T$. H, s_1, and s_2 are all chosen to be of degree 2. Applying (21), if we start from $x(0) = 0$, and as long as $\|w\|_\infty \leq 31.62$, we get $\frac{\|x\|_2}{\|w\|_2} \leq 0.99$.

5 Conclusions

Our expansion of existing SOS programming results to two classes of system theoretic questions about nonlinear systems with polynomial vector fields appears promising. The authors believe that there is a multitude of classes of system theoretic questions that can be answered by application of SOS programming. Work in this area is still in its infancy, and the present classes of problems considered is documented in [4].

For the two cases where the decision polynomials do not enter linearly, we resorted to using iterative algorithms. As limited as the two iterative algorithms are, the underlying technique provides opportunities to extend standard LMI analysis of linear systems to more general polynomial vector fields. A drawback of the approach is that implementation of each algorithm requires a feasible starting point. This may be produced by trial and error, or using established nonlinear design techniques.

6 Acknowledgements

The authors would like to thank the following for providing support for this project: DARPA's Software Enabled Control Program under USAF contract #F33615-99-C-1497, the NSF under contract #CTS-0113985, and DSO National Laboratories-Singapore.

References

1. P. Parrilo, "Semidefinite programming relaxations for semialgebraic problems," *Mathematical Programming*, , no. 96:2, pp. 293–320, 2003.
2. J. Lasserre, "Global optimization with polynomials and the problem of moments," *SIAM Journal of Optimization*, vol. 11(3), pp. 796–817, 2000.
3. N. Shor, *Nondiferentiable Optimization and Polynomial Problems*, Kluwer Academic Pub, 1998.
4. Z. Jarvis-Wloszek, *Lyapunov Based Analysis and Controller Synthesis for Polynomial Systems using Sum-of-Squares Optimizaion*, Ph.D. Dissertation, University of California, Berkeley, Dec 2003, available at `http://jagger.me.berkeley.edu/~zachary/`.
5. Z. Jarvis-Wloszek, R. Feeley, W. Tan, K. Sun, and A. Packard, "Some controls applications of sum of squares programming," in *Proceedings of the Conference on Decision and Control*, 2003, pp. 4676–4681.
6. J. Bochnak, M. Coste, and M-F. Roy, *Géométrie algébrique réelle*, Springer, Berlin, 1986.
7. K. Schmüdgen, "The k-moment problem for compact semialgebraic sets," *Mathematische Annalen*, vol. 289, pp. 203–206, 1991.
8. M. Putinar, "Positive polynomials on compact semialgebraic sets," *Indiana University Mathematical Journal*, vol. 42, pp. 969–984, 1993.
9. D. Henrion and J. Lasserre, "Gloptipoly: Global optimization over polynomials with matlab and sedumi," in *Proc of the Conference on Decision and Control*, 2002, pp. 747–752.
10. M. Choi, T. Lam, and B. Reznick, "Sums of squares of real polynomials," in *Proc of Symposia in Pure Mathematics*, 1995, vol. 58(2), pp. 103–126.
11. V. Powers and T. Wörmann, "An algorithm for sums of squares of real polynomials," *Journal of pure and applied algebra*, vol. 127, pp. 99–104, 1998.
12. S. Prajna, A. Papachristodoulou, and P. Parrilo, "Introducing sostools: A general purpose sum of squares programming solver," in *Proc of the Conf on Decision and Control*, 2002, pp. 741–746.
13. S. Prajna, A. Papachristodoulou, and P. A. Parrilo, *SOSTOOLS: Sum of squares optimization toolbox for MATLAB*, 2002.
14. J. Sturm, "Using SeDuMi 1.02, a Matlab toolbox for optimizaton over symmetric cones," *Optimization Methods and Software*, vol. 11–12, pp. 625–653, 1999, Available at `http ://fewcal.cub.nl/sturm/software/sedumi.html`.
15. S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, SIAM, Philadelphia, 1994.
16. H. Anai and V. Weispfenning, "Reach set computations using real quantifier elimination," in *Hybrid Systems: Comp and Ctrl*, M. Di Benedetto and A. Sangiovanni-Vincentelli, Eds. 2001, number 2034 in Lecture Notes in Comp Sci, pp. 63–76, Springer.

17. P. Parrilo and B. Sturmfels, "Minimizing polynomial functions," in *Workshop on Algorithmic and Quantitative Aspects of Real Algebraic Geometry in Mathematics and Computer Science, held at DIMACS, Rutgers University, March 12–16*, 2001, available at `http://control.ee.ethz.ch/~parrilo`.
18. A. Papachristodoulou and S. Prajna, "On the construction of lyapunov functions using the sum of squares decomposition," in *Proc of the Conf on Decision and Control*, 2002, pp. 3482–3487.

Index