# Contents

1 Polynomial Background ............................................. 2  
1.1 Sum-of-squares polynomials .................................. 3  
  1.1.1 Properties of the set of SOS polynomials .......... 4  
  1.1.2 Computational aspects of SOS polynomials ....... 4  
1.2 The Positivstellensatz ........................................ 11  
  1.2.1 Examples .............................................. 12  
  1.2.2 Theorems related to the Positivstellensatz ....... 15  

2 Global System Theoretic Applications of SOS Polynomials 18  
2.1 Stability Background ........................................ 18  
  2.1.1 Related Polynomial Conditions .................. 21  
2.2 Convex Stability Tests .................................... 22  
  2.2.1 Stability Examples .................................. 24  
2.3 State Feedback Controller Design ......................... 25  
  2.3.1 Iterative State Feedback Design Algorithms ....... 26  
  2.3.2 State Feedback Design Example ................. 29  
2.4 Output Feedback Controller Design ......................... 30  
  2.4.1 Iterative Output Feedback Design Algorithm .... 32  
  2.4.2 Output Feedback Design Example ................. 34  

A Semidefinite Programming ........................................ 36  

Bibliography ....................................................... 38
Chapter 1

Polynomial Background

The properties of real polynomials and especially one very important subset of them form the basis for the results of later chapters; this chapter provides the necessary background for the later results as well as pointers to references for more in-depth treatments of the well developed fields that these topics touch upon.

First, let $\mathbb{Z}_+$ denote the set of nonnegative integers, $\{0, 1, \ldots\}$. With this notation we can make the formal definitions that will be used in almost every result.

**Definition 1 (Monomial)** Every $\alpha \in \mathbb{Z}_+^n$ defines a function $m_\alpha : \mathbb{R}^n \to \mathbb{R}$, called a monomial. Given a specific $\alpha \in \mathbb{Z}_+^n$, the monomial $m_\alpha$ maps $x \in \mathbb{R}^n$ into $m_\alpha(x) = x^\alpha := x_1^{\alpha_1}x_2^{\alpha_2} \cdots x_n^{\alpha_n}$. The degree of a monomial is defined as $\deg m_\alpha := \sum_{i=1}^n \alpha_i$.

**Definition 2 (Polynomial)** A polynomial $p$ is defined as a linear combination of a finite set of monomials $\{m_\alpha\}_{j=1}^k$. Given a set of scalar reals, $\{c_j\}_{j=1}^k \in \mathbb{R}$, a polynomial $p$ is defined as:

$$p := \sum_{j=1}^k c_j m_\alpha_j$$

or in terms of its action on $x \in \mathbb{R}^n$

$$p(x) = \sum_{j=1}^k c_j m_\alpha_j(x) = \sum_{j=1}^k c_j x^{\alpha_j}$$

Using the definition of degree for a monomial, the degree of $p$ is defined as $\deg p := \max_j (\deg m_\alpha_j)$.
The set of polynomials with real coefficients and common independent variables, say, \( x_1, \ldots, x_n \), is often denoted as \( \mathbb{R}[x_1, \ldots, x_n] \) to emphasize that these polynomials form a ring. To eliminate reference to a particular set of independent variables, we will denote the set of all polynomials in \( n \) variables with real coefficients as \( \mathcal{R}_n \), with the assumption that if \( p \in \mathcal{R}_n \) and \( f \in \mathcal{R}_n \) then \( p \) and \( f \) are functions of the same independent variables.

Additionally, define a subset of \( \mathcal{R}_n \), \( \mathcal{R}_{n,d} := \{ p \in \mathcal{R}_n \mid \deg p \leq d \} \); this is just the set of all polynomials in \( n \) variables that have maximum degree \( d \). If all the monomials of polynomial \( p \) are of the same degree, say \( d \), then \( p \) is called homogeneous and it obeys the relation \( p(\lambda x) = \lambda^d p(x) \) for any scalar \( \lambda \).

Another subset of \( \mathcal{R}_n \) is the set of positive semidefinite (PSD) polynomials, which are nonnegative on all of \( \mathbb{R}^n \). This set is defined as \( \mathcal{P}_n := \{ p \in \mathcal{R}_n \mid p(x) \geq 0, \forall x \in \mathbb{R}^n \} \). Also define \( \mathcal{P}_{n,d} := \mathcal{P}_n \cap \mathcal{R}_{n,d} \).

Following standard notation for the real numbers, we will define any of these sets raised to a integer power, \( m \), to denote an \( m \)-vector whose elements are drawn from the indicated set; as an example, \( \mathcal{R}_n^m \) denotes an \( m \)-vector of polynomials in \( n \) variables.

### 1.1 Sum-of-squares polynomials

A very important subset of the polynomials are the Sum-of-Squares (SOS) polynomials. Let \( \Sigma_n \) be the set of all SOS polynomials in \( n \) variables, which is defined as

\[
\Sigma_n := \{ s \in \mathcal{R}_n \mid \exists M < \infty, \exists \{ p_i \}_{i=1}^M \subset \mathcal{R}_n \text{ such that } s = \sum_{i=1}^M p_i^2 \}
\]

The SOS polynomials take their name from the fact that they can be represented as sums of squares of other polynomials. Additionally, define \( \Sigma_{n,d} = \Sigma_n \cap \mathcal{R}_{n,d} \).
1.1.1 Properties of the set of SOS polynomials

Since every $s \in \Sigma_n$ is a sum of squared polynomials, it is clear that $s(x) \geq 0$, $\forall x \in \mathbb{R}^n$, which implies that $\Sigma_n \subseteq P_n$. An interesting question is whether the set of SOS polynomials is equal to or strictly contained in the set of positive semidefinite polynomials.

Hilbert showed that, when restricted to homogeneous polynomials, there are only three cases of $n, d$ such that $\Sigma_{n,d} = P_{n,d}$. These results can be translated to general polynomials, see §3.2 in [12], to prove that $\Sigma_{n,d} = P_{n,d}$ only for

- Polynomials in one variable, $n = 1$.
- Quadratic polynomials, $d = 2$.
- Quartics in two variables, $n = 2$, $d = 4$.

Thus, in general $\Sigma_{n,d} \subset P_{n,d}$. Hilbert’s method to construct a polynomial in $P_n \setminus \Sigma_n$ is very complicated and he did not use it to demonstrate any examples. In [18], Reznick gives an overview of the technique used, its relation to Hilbert’s 17th problem, as well as, a series of examples derived by more modern methods. One of the first examples exhibited dates from 1965 and is the Motzkin polynomial, $M(x, y, z)$ given below, from [10].

$$M(x, y, z) = x^4 y^2 + x^2 y^4 + z^6 - 3x^2 y^2 z^2$$

This polynomial can be shown to be positive semidefinite using the arithmetic-geometric inequality $\frac{a+b+c}{3} \geq (abc)^{\frac{1}{3}}$ with $(a, b, c) = (x^4 y^2, x^2 y^4, z^6)$. By methods to be described later, §1.1.2, it can be shown to not be SOS.

1.1.2 Computational aspects of SOS polynomials

Working with polynomials in $P_n$ can be difficult since there is no full parameterization of the set, nor, in general, are there efficient tests to check if a given polynomial is in the set. However, given the number of variables, $n$, and degree of polynomials, $d$, we can form a full parameterization of $\Sigma_{n,d}$, which directly leads to an efficient
semidefinite programming test to check if a polynomial is SOS (see Appendix A for a brief overview of semidefinite programming).

A full parameterization of fixed degree SOS polynomials

First we note that SOS polynomials must always be of even degree, so we will consider the parameterization of the set $\Sigma_{n, 2d}$ for some $n, d \in \mathbb{Z}_+$. The following lemma provides the starting point for the parameterization.

**Lemma 1** If $s \in \Sigma_{n, 2d}$, then there exist $p_i \in \mathbb{R}^{n,d}$, $i = 1, \ldots, M$, for some finite $M$ such that

$$s = \sum_{i=1}^{M} p_i^2$$

This lemma is a restricted version of Theorem 1 in [17], which gives tighter restrictions on the $p_i$’s when $s$ is known.

Using Lemma 1, we can pose a full parameterization, often referred to as the “Gram matrix” approach, [6]. First, define $z_{n,d}$ to be the vector of all monomials in $n$ variables of degree less than or equal to $d$ ordered in the following manner. Given $\alpha, \beta \in \mathbb{Z}_+^n$, $x^\alpha$ precedes $x^\beta$ if $\deg x^\alpha < \deg \beta$ or if $\deg x^\alpha = \deg \beta$ and the first entry of $\alpha - \beta$ that is strictly negative is preceded by a strictly positive entry. As an example, with $n = 2, d = 2$

$$z_{2,2}(x) := \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ x_1^2 \\ x_1x_2 \\ x_2^2 \end{bmatrix}$$

For a general pair of $n$ and $d$, $z_{n,d}(x)$ will be a $\binom{n+d}{d}$-vector.

With the definition of $z_{n,d}(x)$ it is possible to characterize a polynomial, $p \in \mathcal{R}_{n, 2d}$, as

$$p(x) = z_{n,d}^*(x)Qz_{n,d}(x)$$
where $Q$ is the “Gram” matrix. The idea of representing polynomials as quadratic forms of vectors of monomials predates [6] by many years. The earliest quadratic representation, dating from 1968, [3], started a framework which was used to find homogeneous polynomial Lyapunov functions in [24]. The following result, which first appeared as Proposition 2.3 in [6], generalizes the earlier works and establishes when a polynomial is SOS.

**Theorem 1**  Fix $p \in \mathcal{R}_{n,2d}$. $p \in \Sigma_{n,2d}$ if and only if there exists a $Q \succeq 0$ such that $p(x) = z_{n,d}(x)^*Qz_{n,d}(x)$.

**Proof:**

$\Rightarrow$ If $p \in \Sigma_{n,2d}$ then via Lemma 1 we know that there exist $p_i \in \mathcal{R}_{n,d}$, $i = 1, \ldots, M$ such that $p = \sum_{i=1}^{M} p_i^2$. Writing each of these polynomials as $p_i(x) = q_i^*z_{n,d}(x)$ with $q_i$ a real vector of appropriate dimension, we have

\[
p(x) = \sum_{i=1}^{M} \left( q_i^*z_{n,d}(x) \right)^2 \\
= \sum_{i=1}^{M} z_{n,d}(x)q_iq_i^*z_{n,d}(x) \\
= z_{n,d}(x) \left( \sum_{i=1}^{M} q_iq_i^* \right) z_{n,d}(x) \\
= z_{n,d}(x)Qz_{n,d}(x)
\]

From its construction it is clear that $Q \succeq 0$.

$\Leftarrow$ Since $Q \succeq 0$, we can factor $Q = \sum_{i=1}^{r} q_iq_i^*$ where $r$ is the rank of $Q$. Then reversing the argument above

\[
p(x) = z_{n,d}(x) \left( \sum_{i=1}^{r} q_iq_i^* \right) z_{n,d}(x) \\
= \sum_{i=1}^{r} z_{n,d}(x)q_iq_i^*z_{n,d}(x) \\
= \sum_{i=1}^{r} \left( q_i^*z_{n,d}(x) \right)^2 \\
\overset{(a)}{=} \sum_{i=1}^{r} p_i^2(x)
\]
where (a) comes from defining \( p_i(x) := q_i^* z_{n,d}(x) \).

\[ \square \]

**Corollary 1** *The number of terms for an SOS decomposition can be chosen to be the number of elements in \( z_{n,d} \) or fewer.*

This theorem gives necessary and sufficient conditions for a polynomial to be SOS, however for \( p \in \Sigma_{n,2d} \) there are, in general, many symmetric \( Q \) such that \( p(x) = z_{n,d}^*(x) Q z_{n,d}(x) \) and some are not positive semidefinite as the following example shows.

**Example 1** *Take \( p \in \mathcal{R}_{2,4} \) to be such that \( p(x) = x_1^4 + x_2^3 x_2 + x_2^4. \) Both \( Q_1 \) and \( Q_2 \) below are such that \( z_{2,2}^*(x) Q z_{2,2}(x) = p(x). \)

\[
\begin{bmatrix}
1 \\
x_1 \\
x_2 \\
x_1^2 \\
x_1 x_2 \\
x_2^2
\end{bmatrix}^* \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 \\
\end{bmatrix} = \begin{bmatrix}
1 \\
x_1 \\
x_2 \\
x_1^2 \\
x_1 x_2 \\
x_2^2
\end{bmatrix}^* \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
\end{bmatrix} = \begin{bmatrix}
1 \\
x_1 \\
x_2 \\
x_1^2 \\
x_1 x_2 \\
x_2^2
\end{bmatrix} = p(x)
\]

Note that \( Q_1 \succeq 0 \) while \( Q_2 \preceq 0 \). \( Q_1 \)'s positive semidefiniteness shows that \( p \in \Sigma_{2,4} \), while \( Q_2 \preceq 0 \) shows nothing.

**An LMI test for SOS**

Theorem 1 gives the complete parametrization of the SOS polynomials for a given number of variables and fixed degree, however it does not give a method to check if a given polynomial is SOS.

In an effort to understand the set of matrices \( Q \) that make \( z_{n,d}^*(x) Q z_{n,d}(x) = p(x) \) for some \( p \in \mathcal{R}_{n,2d} \), pick the standard basis for symmetric matrices, \( \{ E_i \} \), of the
appropriate size, \((n^d + d) \times (n^d + d)\). Working out \(z_{n,d}^*(x) (\sum q_i E_i) z_{n,d}(x)\) and equating coefficients with \(p(x)\) shows that set of matrices that make the equality hold are an affine subspace of the symmetric matrices as was shown in [13].

Given \(p \in \mathcal{R}_{n,2d}\), let \(Q_0\) be any symmetric matrix such that

\[
z_{n,d}^*(x)Q_0z_{n,d}(x) = p(x)
\]

and let \(\{Q_i\}_{i=1}^{n_q}\) be the set of symmetric matrices such that

\[
z_{n,d}^*(x)Q_i z_{n,d}(x) = 0
\]

With this setup, we can define the affine subspace of symmetric matrices related to \(p\) as

\[
Q_p := \{Q | z_{n,d}^*(x)Qz_{n,d} = p(x)\} = \left\{ Q_0 + \sum_{i=1}^{n_q} \lambda_i Q_i | \lambda_i \in \mathbb{R}, i = 1, \ldots, n_q \right\}
\]

The following example illustrates the general procedure for finding the set of \(Q_i\)'s that define the subspace.

**Example 2** For \(p \in \mathcal{R}_{2,4}\) find some symmetric \(Q_0\) such that \(z_{2,2}^*(x)Q_0z_{2,2}(x) = p(x)\).

Picking the standard basis for symmetric matrices, we write \(z_{2,2}^*(x)Qz_{2,2}(x) = 0\) as

\[
\begin{bmatrix}
1 \\
x_1 \\
x_2 \\
x_1^2 \\
x_2^2
\end{bmatrix}^* \begin{bmatrix}
q_1 & q_2 & q_3 & q_4 & q_5 & q_6 \\
q_2 & q_7 & q_8 & q_9 & q_{10} & q_{11} \\
q_3 & q_8 & q_{12} & q_{13} & q_{14} & q_{15} \\
q_4 & q_9 & q_{13} & q_{16} & q_{17} & q_{18} \\
q_5 & q_{10} & q_{14} & q_{17} & q_{19} & q_{20} \\
q_6 & q_{11} & q_{15} & q_{18} & q_{20} & q_{21}
\end{bmatrix} \begin{bmatrix}
1 \\
x_1 \\
x_2 \\
x_1^2 \\
x_2^2
\end{bmatrix} = 0
\]

Equating terms we have...
and since each coefficient must be identically zero, we can find the subspace of matrices such that $z_{2,2}^2(x)Qz_{2,2}(x) = 0$ to be

$$\{ Q | z_{2,2}^2(x)Qz_{2,2}(x) = 0 \} = \{ \sum_{i=1}^6 \lambda_i Q_i | \lambda_i \in \mathbb{R} \}$$

which shows how to find $Q_p$ for a polynomial with fixed number of variables and degree.

In [13], Powers and Wörmann got as far as finding the affine subspace, which allowed them to give the equivalence $p \in \Sigma_{n,2d}$ iff $\exists Q \in Q_p$ such that $Q \succeq 0$. However, they did not recognize that checking if there existed $\lambda_i$’s to make $Q_0 + \sum \lambda_i Q_i \succeq 0$ was convex and just an LMI feasibility problem, instead they proposed a less efficient search method using quantifier elimination. In [12], Parrilo realized that the existence of a $Q \succeq 0$, can be solved as an LMI and gave the following theorem.
Theorem 2 ([12], Theorem 3.3) Given $p \in \mathcal{R}_{n,2d}$, find the relevant affine subspace $\mathcal{Q}_p = \{Q_0 + \sum_i \lambda_i Q_i | \lambda_i \in \mathbb{R}\}$. $p \in \Sigma_{n,2d}$ iff the following LMI is feasible

$$\exists \lambda_i$$

$$\text{s.t. } Q_0 + \sum \lambda_i Q_i \succeq 0$$

Proof:

From Theorem 1, we know that $p \in \Sigma_{n,2d}$ iff there exists a $Q \succeq 0$ such that $z_{2n,d}^*(x)Qz_{n,d}(x) = p(x)$, so we only need search over $\mathcal{Q}_p$, which is exactly the LMI given.

$k$

Parrilo also introduced the following important extension that can be proved in a similar manner to Theorem 2.

Theorem 3 ([12], §3.2) 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 very useful since it allows to answer questions like the following example.

Example 3 Given $p_0, p_1 \in \mathcal{R}_n$, does there exist $k \in \mathcal{R}_n$ such that

$$p_0 + kp_1 \in \Sigma_n$$

(1.1)

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 (1.1) using this decomposition

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

which, since $(m_j p_1) \in \mathcal{R}_n$, can be checked by Theorem 3.
A software package, SOSTOOLS [14, 15], was written to aid in solving the LMIs that result from Theorem 3. This package sets up the LMIs from the polynomial problems, does some smart preprocessing to reduce problem size and uses Sturmand’s SeDuMi semidefinite programming solver, [22], to solve the LMIs.

Additional computational gains can be had by exploiting polynomial symmetries, [7], and using the Newton polytope algorithm presented in [6] to reduce the number of monomials in the Gram matrix formulation, which makes the resulting LMIs smaller with fewer free parameters. These computational improvements are set to appear in the next release of SOSTOOLS.

1.2 The Positivstellensatz

Having introduced SOS polynomials it is now possible to make the algebraic definitions that are necessary to present one of the seminal theorems of real algebraic geometry, which generalizes many known results including the S-Procedure, (1.2.1).

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

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

**Definition 4** Given \( \{f_1, \ldots, f_s\} \in \mathbb{R}_n \), the *Cone* generated by \( f_i \)'s is

\[
\mathcal{P}(f_1, \ldots, f_s) := \left\{ s_0 + \sum s_ib_i | s_i \in \Sigma_n, b_i \in \mathcal{M}(f_1, \ldots, f_s) \right\}
\]

For completeness note that \( \mathcal{P}(\phi) := \Sigma_n \).

Remembering that if \( f \in \mathbb{R}_n, s \in \Sigma_n \), then \( sf^2 \in \Sigma_n \), allows us to write any cone as a sum of \( 2^s \) terms. Note that this reduction in the number of free SOS polynomials need not be beneficial.

An example: \( \mathcal{P}(f_1, f_2) = \{s_0 + s_1f_1 + s_2f_2 + s_3f_1f_2 | s_0, \ldots, s_3 \in \Sigma_n\} \)
Definition 5  Given \( \{h_1, \ldots, h_u\} \in \mathcal{R}_n \), the **Ideal** generated by \( h_k \)’s is

\[
\mathcal{I}(h_1, \ldots, h_u) := \left\{ \sum h_k p_k \mid p_k \in \mathcal{R}_n \right\}
\]

For completeness note that \( \mathcal{I}(\phi) := 0 \).

With these definitions we can state the following theorem which is a version of the original theorem in [20] restricted to \( \mathbb{R}^n \).

Theorem 4 (Positivstellensatz [2, Theorem 4.2.2] ) Given sets of polynomials \( \{f_1, \ldots , f_s\}, \ {g_1, \ldots , g_t\} \), and \( \{h_1, \ldots , h_u\} \) in \( \mathcal{R}_n \), the following are equivalent:

1. The set

\[
\{x \in \mathbb{R}^n \mid f_i(x) \geq 0, \ g_j(x) \neq 0, \ h_k(x) = 0, \ i = 1, \ldots , s, \ j = 1, \ldots , t, \ k = 1, \ldots , u\}
\]

is empty,

2. There exist polynomials \( f \in \mathcal{P}(f_1, \ldots, f_s) \), \( g \in \mathcal{M}(g_1, \ldots, g_t) \), \( h \in \mathcal{I}(h_1, \ldots, h_u) \) such that

\[
f + g^2 + h = 0
\]

1.2.1 Examples

To reinforce the usefulness of the Positivstellensatz (P-satz), consider the range of the following examples that become convex and thus tractable when the P-satz is combined with the results of Theorem 3.

Positivstellensatz Certificates

The LMI based tests for SOS polynomials from Theorem 3 can be used to prove that the set emptiness condition from the P-satz holds, by finding specific \( f, g \), and \( h \) such that \( f + g^2 + h = 0 \). These \( f, g \), and \( h \) are known as P-satz certificates since they certify that the equality holds. The following theorem states precisely how semidefinite programming can be used to search for certificates.
Theorem 5 (Theorem 4.8, [12]) Given polynomials \(\{f_1, \ldots, f_s\}, \{g_1, \ldots, g_t\},\) and \(\{h_1, \ldots, h_u\}\) in \(\mathbb{R}^n\), if the set
\[
\{x \in \mathbb{R}^n | f_i(x) \geq 0, g_j(x) \neq 0, h_k(x) = 0, i = 1, \ldots, s, j = 1, \ldots, t, k = 1, \ldots, u\}
\]
is empty then the search for bounded degree Positivstellensatz refutations can be done using semidefinite programming. If the degree bound is chosen large enough the semidefinite programs will be feasible and give the refutation certificates.

The proof of this theorem involves writing out all of the terms of \(P(f_1, \ldots, f_s)\), \(\mathcal{M}(g_1, \ldots, g_t)\), and \(\mathcal{I}(h_1, \ldots, h_u)\) to form the equality constraint, \(f + g^2 + h = 0\). For a fixed degree \(d\), set the term \(g \in \mathcal{M}(g_1, \ldots, g_t)\) such that it is of degree greater than or equal to \(d/2\), then pick each of the free polynomials in \(f\) and \(h\) such that they have degree at least \(d\). Now run the LMI with the equality constraint as well as the SOS constraints on the free polynomials in \(f\). If you search over all \(g\) for each \(d\), then you eventually find the Positivstellensatz certificates.

An LMI test for \(\mathcal{P}_n\)

Using the P-satz, we can now test to see if a polynomial \(p \in \mathbb{R}^n\) is in \(\mathcal{P}_n\). If \(p \in \mathcal{P}_n\), then \(\forall x \in \mathbb{R}^n, p(x) \geq 0\). Equivalently \(\{x \in \mathbb{R}^n | p(x) < 0\}\) is empty, or in the P-satz format
\[
\{x \in \mathbb{R}^n | -p(x) \geq 0, p(x) \neq 0\}\text{ is empty}
\]
This condition holds iff \(\exists f \in \mathcal{P}(-p)\) and \(g \in \mathcal{M}(p)\) such that \(f + g^2 = 0\). Using the definition of the cone and the monoid, \(p \in \mathcal{P}_n\) iff \(\exists s_0, s_1 \in \Sigma_n\) and \(k \in \mathbb{Z}_+\) such that
\[
s_0 - ps_1 + p^{2k} = 0
\]
If we fix \(k\) and the degree of \(s_1\) to be \(d\), we can rewrite the conditions above as \(p \in \mathcal{P}_n\) iff \(\exists s_1\) such that
\[
s_1 \in \Sigma_{n,d}
\]
\[
ps_1 - p^{2k} \in \Sigma_{n,d}
\] (1.2)
with \( d = \max(2k \deg p, d + \deg p) \). For fixed \( k \) and \( d \) we know that, via Theorem 3, checking the conditions in (1.2) is just an LMI, so for fixed \( k \) and \( d \) we have an LMI sufficient condition for a polynomial to be PSD.

**The \( S \)-Procedure**

What does the familiar \( S \)-procedure look like in the Positivstellensatz formalism? Given symmetric \( n \times n \) matrices \( \{A_i\}_{i=0}^m \), the \( S \)-procedure states: if there exist nonnegative scalars \( \{\lambda_i\}_{i=1}^m \) such that \( A_0 - \sum_{i=1}^m \lambda_i A_i \succeq 0 \), then

\[
\bigcap_{i=1}^m \{x \in \mathbb{R}^n | x^* A_i x \geq 0\} \subset \{x \in \mathbb{R}^n | x^* A_0 x \geq 0\}
\]

Rephrased as a set emptiness question, we would like to know if

\[
W := \{x \in \mathbb{R}^n | x^* A_1 x \geq 0, \ldots, x^* A_m x \geq 0, x^* A_0 x \geq 0, x^* A_0 x \neq 0\}
\]

is empty?

If the \( \lambda_i \) exist, define \( Q := A_0 - \sum_{i=1}^m \lambda_i A_i \). By assumption \( Q \succeq 0 \) and thus \( x^* Q x \in \Sigma_n \). Define \( g(x) := x^* A_0 x \in \mathcal{M}(x^* A_0 x) \) as well as

\[
f(x) := (x^* Q x)(-x^* A_0 x) + \sum_{i=1}^m \lambda_i (-x^* A_0 x)(x^* A_i x)
\]

By their non-negativity each \( \lambda_i \in \Sigma_n \) which shows that function \( f(x) \) is in the cone \( \mathcal{P}(x^* A_1 x, \ldots, x^* A_m x, -x^* A_0 x) \). An easy rearrangement gives \( f + g^2 = 0 \), which illustrates that \( f \) and \( g \) are Positivstellensatz certificates that prove that \( W \) is empty.

**A generalized \( S \)-Procedure**

The \( S \)-procedure given above can be generalized to deal with non-quadratic functions and non-scalar weights in the following way. Given \( \{p_i\}_{i=0}^m \in \mathcal{R}_n \), if there exist \( \{s_i\}_{i=1}^m \in \Sigma_n \) such that

\[
p_0 - \sum_{i=1}^m s_i p_i = q
\]
with \( q \in \Sigma_n \), which for fixed degree \( s_i \)'s can be checked with Theorem 3, then
\[
\bigcap_{i=1}^{m} \{ x \in \mathbb{R}^n | p_i(x) \geq 0 \} \subset \{ x \in \mathbb{R}^n | p_0(x) \geq 0 \}
\]
The related set emptiness question asks if
\[
W := \{ x \in \mathbb{R}^n | p_1(x) \geq 0, \ldots, p_m(x) \geq 0, -p_0(x) \geq 0, p_0(x) \neq 0 \}
\]
is empty. Similar to the standard \( S \)-procedure approach, define \( g := p_0 \in \mathcal{M}(p_0) \) as well as
\[
 f := -qp_0 - \sum_{i=1}^{n} s_ip_0p_i
\]
Since \( q \) as well as the \( s_i \)'s are SOS, \( f \in \mathcal{P}(p_1, \ldots, p_m, -p_0) \). Verifying \( f + g^2 = 0 \),
\[
f + g^2 = -qp_0 - \sum_{i=1}^{n} s_ip_0p_i + p_0^2
\]
\[
= - \left( p_0 - \sum_{i=1}^{m} s_ip_i \right) p_0 - \sum_{i=1}^{n} s_ip_0p_i + p_0^2
\]
\[
= 0
\]
illustrating that \( f \) and \( g \) provide certificates that the set \( W \) is empty.

1.2.2 Theorems related to the Positivstellensatz

The multidimensional moment problem, which considers when a sequence of numbers are the moments of some nonnegative Borel measure on \( \mathbb{R}^n \), has a long relation with SOS polynomials, [1]. Many interesting results about SOS polynomials have been generated from this approach and two theorems that are especially interesting to polynomial optimization are presented below. The setup is as follows; let \( \{ f_i \}_{i=1}^{m} \in \mathcal{R}_n \) and define \( K := \{ x | f_1(x) \geq 0, \ldots, f_m(x) \geq 0 \} \).

**Theorem 6 (Corollary 3, [19])** If \( K \) is compact and \( p \in \mathcal{R}_n \) is such that \( p(x) > 0 \) for all \( x \in K \), then \( p \in \mathcal{P}(f_1, \ldots, f_m) \).
This tells us that if a polynomial is positive on $K$ then it is in the cone generated by the polynomials that describe $K$. With one additional assumption this result can be strengthened further.

**Theorem 7 (Lemma 4.1, [16])** Let $K$ be compact. $p \in \mathcal{R}_n$ such that $p(x) > 0$ for all $x \in K$ belongs to the set

$$\{s_0 + f_1 s_1 + \cdots + f_m s_m | s_0, \cdots, s_m \in \Sigma_n\}$$

if and only if there is a polynomial $g$ in the set with the property that $g^{-1}[0, \infty)$ is compact in $\mathbb{R}^n$.

These theorems can be used to define certificate searches with fewer terms than the P-satz would require, however, these smaller searches can require polynomials of much higher degree. In [21], Stengle provides a simple example that requires unbounded degree polynomial certificates using Theorem 7, but has degree four certificates if the P-satz is used, as shown in [5].

**Applications to Polynomial Optimization**

Consider the following optimization problem

$$\begin{align*}
\min_{x \in \mathbb{R}^n} f_0(x) \\
\text{s.t. } f_1(x) &\geq 0 \\
&\vdots \\
&f_m(x) \geq 0
\end{align*}$$

with $\{f_i\}_{i=0}^m \in \mathcal{R}_n$ and no assumed convexity. Let the optimum value be $f^* > -\infty$ with $f_0(x^*) = f^*$.

Lasserre, [9], noticed that if the feasible region is compact we can always satisfy the requirements of Theorem 7 by adding an additional constraint on norm of $x$, $f_{m+1}(x) := a - \|x\|^2 \geq 0$, since $f^{-1}_{m+1}[0, \infty)$ is clearly compact. Additionally, if $\gamma$ is a lower bound on $f^*$, then $f_0(x) > \gamma$ at any feasible point, which implies that
$f_0(x) - \gamma > 0$ for $\{x | f_1(x) \geq 0, \cdots, f_{m+1}(x) \geq 0\}$. Using the theorem, we can rewrite the optimization (1.3) as

$$\max \gamma$$
$$\text{s.t. } f_0 - \gamma = s_0 + f_1 s_1 + \cdots + f_{m+1} s_{m+1}$$
$$s_0, \cdots, s_{m+1} \in \Sigma_n$$

(1.4)

which Theorem 3 shows to be an LMI, as long as the degree of the SOS polynomials is fixed, however, the degree for which the equality constraint in (1.4) holds is unknown. By increasing the maximum degree of the $s_i$’s, this approach allows for a series of convex relaxations to the nonconvex problem (1.3), which are shown to monotonically converge to $f^*$ in [9]. A software package to carry out this algorithm is described in [8].
Chapter 2

Global System Theoretic
Applications of SOS Polynomials

If we consider the system

\[ \dot{x} = f(x) \]  

(2.1)

for \( x \in \mathbb{R}^n \) with \( f \in \mathbb{R}_n^n \) as well as \( f(0) = 0 \), we can pose many global system theoretic questions about its behavior as searches for SOS polynomials. However, first we need a few definitions that will allow us to make the Lyapunov based stability arguments that will be at the heart of the polynomial searches.

Define the flow of the system (2.1) starting from a point \( x_0 \in \mathbb{R}^n \) and evolving forward for \( t \) time units to be \( \phi_t(x_0) \). Additionally for a differentiable scalar function \( V \), defined on the same state space as (2.1), define its derivative with respect to time, \( \dot{V} \), as the dot product between its gradient, \( \nabla V \), and \( f \).

2.1 Stability Background

Using a polynomial Lyapunov function we can prove global asymptotic stability as well as global exponential stability of the system (2.1) by checking semialgebraic conditions on polynomials. However, we will first state and prove the conditions that we will later exploit to design Lyapunov functions.
Definition 6 (\( \mathcal{K}_\infty \) Functions) A function \( \sigma : \mathbb{R} \to \mathbb{R} \) is called a \( \mathcal{K}_\infty \) function if it is continuous, strictly increasing, and has the properties \( \sigma(0) = 0 \) and \( \sigma(\xi) \to \infty \) as \( \xi \to \infty \).

Definition 7 (Positive Definite Functions) A function \( \rho : \mathbb{R}^n \to \mathbb{R} \) is called positive definite if it is continuous, has the property \( \rho(0) = 0 \), and there exists some \( \mathcal{K}_\infty \) function \( \sigma \) such that

\[
\sigma(\|x\|) \leq \rho(x)
\]

for all \( x \in \mathbb{R}^n \).

With these definitions the standard Lyapunov theorems for global asymptotic stability as well as global exponential stability can be stated and proved.

Theorem 8 (Lyapunov) The system (2.1) is globally asymptotically stable about its equilibrium point if there exists a positive definite function \( V : \mathbb{R}^n \to \mathbb{R}_+ \) such that \( -\dot{V} \) is also positive definite.

Proof:

By definition there exist \( \mathcal{K}_\infty \) functions \( \alpha, \beta \) such that \( \alpha(\|x\|) \leq V(x), \forall x \in \mathbb{R}^n \) and \( \beta(\|x\|) \leq -\dot{V}(x), \forall x \in \mathbb{R}^n \). We will first prove stability followed by asymptotic convergence to the fixed point; both results will be shown by contradiction.

Given any \( \epsilon > 0 \) pick \( \delta_\epsilon \) such that

\[
\sup_{\|x\|<\delta_\epsilon} V(x) < \alpha(\epsilon)
\]

which, by the continuity of \( V \), always exists. Pick any point \( x_0 \) such that \( \|x_0\| < \delta_\epsilon \). At this point

\[
\alpha(\|x_0\|) \leq V(x_0) < \alpha(\epsilon)
\]

which implies that \( \|x_0\| < \epsilon \). Assume there exists a \( T > 0 \) such that \( \|\phi_T(x_0)\| \geq \epsilon \), which would imply that

\[
V(x_0) < \alpha(\epsilon) \leq V(\phi_T(x_0))
\]
contradicting the assumption that $-\dot{V}$ is positive definite.

To show asymptotic convergence to the fixed point, $x = 0$, for any given $x_0 \in \mathbb{R}^n$ and $\epsilon > 0$, we need to find a $T > 0$ such that $\|\phi_t(x_0)\| < \epsilon$ for all $t \geq T$. If $x_0 = 0$ then there is nothing to prove, so we can assume that $x_0 \neq 0$. Assume that there exists an $x_0 \neq 0$ and $\epsilon > 0$ such that $\|\phi_t(x_0)\| \geq \epsilon$ for all $t > 0$. By integration we know that

$$V(\phi_t(x_0)) = V(x_0) + \int_0^t \dot{V}(\phi_\tau(x_0)) \, d\tau$$

Using the positive definiteness of $V$ as well as the fact that $\|\phi_t(x_0)\| \geq \epsilon$ we can bound the expression above from below,

$$\alpha(\epsilon) \leq \alpha(\|\phi_t(x_0)\|) \leq V(\phi_t(x_0)) = V(x_0) + \int_0^t \dot{V}(\phi_\tau(x_0)) \, d\tau$$

Additionally using the positive definiteness of $-\dot{V}$ and the fact that $\|\phi_t(x_0)\| \geq \epsilon$ we can bound $V(\phi_t(x_0))$ from above,

$$V(\phi_t(x_0)) = V(x_0) + \int_0^t \dot{V}(\phi_\tau(x_0)) \, d\tau \leq V(x_0) - \int_0^t \beta(\|\phi_\tau(x_0)\|) \, d\tau \leq V(x_0) - t\beta(\epsilon)$$

End-to-end we now have

$$\alpha(\epsilon) \leq V(x_0) - t\beta(\epsilon)$$

for all $t > 0$. However, if $t \geq V(x_0)/\beta(\epsilon)$ this implies that $\alpha(\epsilon) \leq 0$, which contradicts $\epsilon > 0$.

\[ \square \]

Building from Theorem 8, we can now state the following theorem for global exponential stability.

**Theorem 9** If there exists a function $V$, such that $V(x) \geq \alpha \|x\|_d^d \forall x \in \mathbb{R}^n$, where $\alpha > 0$ and $d$ is an integer greater than one, as well as a $\gamma > 0$ such that

$$\dot{V}(x) \leq -\gamma V(x)$$

for all $x \in \mathbb{R}^n$, then the system (2.1) is globally exponentially stable about its fixed point, $x = 0$, with an exponential convergence rate of $\gamma/d$. 


Proof:

By definition $\alpha \|x\|_d^d \leq V(x)$, $\forall x \in \mathbb{R}^n$. Clearly the function $\alpha(\cdot)^d$ is in class $\mathcal{K}_\infty$, and $\gamma \alpha \|x\|_d^d \leq \gamma V(x) \leq -\dot{V}(x)$, for all $x \in \mathbb{R}^n$, which shows that $-\dot{V}$ is positive definite and therefore the system (2.1) is globally asymptotically stable about $x = 0$.

For $x \neq 0$, $V(x) > 0$ which allows us to write one of the assumptions as

$$\frac{\dot{V}(x)}{V(x)} \leq -\gamma$$

or

$$\frac{d}{dt} \log(V(x)) \leq -\gamma$$

which can be integrated over $[0, t]$ starting from $x_0$ to give

$$\log(V(\phi_t(x_0))) \leq \log(V(x_0)) - \gamma t$$

which gives the exponential bound

$$V(\phi_t(x_0)) \leq V(x_0) e^{-\gamma t}$$

that proves that $V(\phi_t(x_0))$ decays exponentially with rate $\gamma$. Since $\alpha \|\phi_t(x_0)\|_d^d \leq V(\phi_t(x_0))$ for all $t > 0$, we can make the following bound

$$\|\phi_t(x_0)\|_d \leq \left(\frac{V(x_0)}{\alpha}\right)^{\frac{1}{d}} e^{-\frac{\gamma}{d} t}$$

which shows that $\|\phi_t(x_0)\|$ decays exponentially with a rate of $\gamma/d$.

\[\square\]

2.1.1 Related Polynomial Conditions

These Lyapunov based stability tests rely on assumptions of positive definiteness and global non-negativity of an unknown function $V$, its time derivative $\dot{V}$ and their linear combinations ($\dot{V} + \gamma V$). If $V$ is restricted to be polynomial, then we can check the assumptions of Theorems 8 and 9 via the P-satz.
Given a known function \( l \in \mathcal{R}_n \) that is positive definite, we known that if \((V - l) \in \mathcal{P}_n\) then \(V\) is positive definite as well. Thinking back to §1.2.1, we know that \((V - l) \in \mathcal{P}_n\) iff the set

\[
\{ x \in \mathbb{R}^n | -(V(x) - l(x)) \geq 0, V(x) - l(x) \neq 0 \}
\]

is empty, which is equivalent to the existence of \(s_0, s_1 \in \Sigma_n\) and \(k \in \mathbb{Z}_+\) such that

\[
s_0 - (V - l)s_1 + (V - l)^{2k} = 0.
\]

Since \(V\) is unknown and we wish to search over many \(V\)’s using Theorem 3, we must pick values for \(k\), \(s_0\) and \(s_1\) such that the condition is linear in the monomials that make up \(V\). We can pick \(s_0 = 0\) and \(k = 1\) to form the following sufficient condition for \(V\)’s positive definiteness

\[
V - l \in \Sigma_n
\]

which for a fixed positive definite \(l\) and fixed degree \(V\) is a semidefinite programming feasibility problem. Using the P-satz in the same way results in a sufficient condition for \(-\gamma V + \dot{V} \in \mathcal{P}_n\) which is just

\[
-(\gamma V + \dot{V}) \in \Sigma_n.
\]

Remembering that \(f\) is fixed and \(\nabla V\) is a linear combination of the monomials of \(V\) establishes that for fixed \(\gamma\) the condition above can also be checked as a semidefinite programming feasibility problem.

**Remark 1** From these examples we can generalize the following result. Given a polynomial \(p\), if we desire that \(p \in \mathcal{P}\) then a sufficient condition that is linear in the monomials of \(p\) is just \(p \in \Sigma\).

### 2.2 Convex Stability Tests

Using the polynomial conditions from §2.1.1 to test the assumptions to Theorems 8 and 9, we can formulate the following convex stability tests.
Lemma 2 Given the system (2.1) and fixed positive definite functions $l_1, l_2 \in \mathcal{R}_n$, the system is globally asymptotically stable if there exists $V \in \mathcal{R}_n$ with $V(0) = 0$ such that

$$V - l_1 \in \Sigma_n$$

$$- (\nabla V^* f + l_2) \in \Sigma_n$$

which when $V$ has fixed degree is an LMI.

Proof:

From Theorem 3, it is clear that the conditions that $V - l_1$ and $- (\nabla V^* f + l_2)$ are SOS polynomials can be checked as LMIs. If a $V$ is found that meets these conditions, the positive definiteness of $l_1$ and $l_2$ insure that both $V$ and $-\dot{V}$ are positive definite, which meets the assumptions of Theorem 8 making the system globally asymptotically stable.

\[\square\]

Note that if the dynamics were chosen to be linear, $f(x) = Ax$ and the Lyapunov function to be quadratic $V(x) = x^*Px$ then the lemma’s SOS conditions can be simplified to the LMI conditions $P \succ 0$ and $A^*P + PA < 0$.

A set of sufficient conditions which are very similar to Lemma 2 appear in [11], where they are used to prove stability for systems like (2.1) with additional state and control equality and inequality constraints.

Looking to Theorem 9, we can formulate a similar set of polynomial conditions, however due to a term that is bilinear in $\gamma$ and $V$, we can not check the assumptions with a single LMI.

Lemma 3 Given the system (2.1) and the fixed positive definite function $l(x) = \|x\|_d^d$ with $d$ an integer greater than one, the system is globally exponentially stable if there exists $\gamma > 0$ and $V \in \mathcal{R}_n$ with $V(0) = 0$ such that

$$V - l \in \Sigma_n$$

$$- (\gamma V + \nabla V^* f) \in \Sigma_n$$

which when $V$ has fixed degree can be checked by performing a linesearch on $\gamma$ and solving the resulting LMI at each point. Additionally, the rate of convergence is $\gamma/d$. 

Proof:

The proof follows along the same lines as the proof for Lemma 2 by establishing that the assumptions for Theorem 9 are met.

In general the value for $d$ in this and other related lemmas will be picked to be the highest even degree in $V$.

\[ \square \]

Again, if the dynamics are linear and the Lyapunov function is quadratic the SOS conditions of Lemma 3 collapse into simple LMIs, which are remain bilinear in $\gamma$, $P \succ 0$ and $A^*P + PA \preceq -\gamma P$.

Note that for both LMI based lemmas, if the system (2.1) has equilibrium points away from $x = 0$, then neither of these conditions can ever hold and moreover the system can not be globally asymptotically stable.

### 2.2.1 Stability Examples

Consider the following system

\[
\begin{bmatrix}
\dot{x}_1 \\
\dot{x}_2 \\
\end{bmatrix} = \begin{bmatrix}
-x_2 - x_1^3 \\
x_1 - x_2^3 \\
\end{bmatrix}
\]

where it is clear that $f(0) = 0$. The linearization about the origin has eigenvalues of $\pm j$, so it is not even possible to verify that the nonlinear system (2.2) is stable.

If we look to construct a Lyapunov function to demonstrate stability, a simple quadratic Lyapunov function $V(x) = \|x\|_2^2$ will work. This $V$ is clearly positive definite, and if we compute its time derivative we find

\[
\dot{V} = \nabla V^* f \\
= 2x_1(-x_2 - x_1^3) + 2x_2(x_1 - x_2^3) \\
= -2(x_1^4 + x_2^4)
\]

which clearly makes $-\dot{V}$ positive definite. The definiteness of $V$ and $-\dot{V}$ satisfy the assumptions of Theorem 8, so it is clear that the system (2.2) is asymptotically stable.
Table 2.1: Results of applying Lemma 3 to system (2.2)

<table>
<thead>
<tr>
<th>( d )</th>
<th>( \gamma_{\text{max}} )</th>
<th>( \gamma/d )</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0.0052</td>
<td>0.0013</td>
</tr>
<tr>
<td>6</td>
<td>0.0431</td>
<td>0.0072</td>
</tr>
<tr>
<td>8</td>
<td>0.1094</td>
<td>0.0137</td>
</tr>
<tr>
<td>10</td>
<td>0.1472</td>
<td>0.0147</td>
</tr>
</tbody>
</table>

However, it is unclear how to construct a Lyapunov function that will demonstrate exponential stability, so we will use Lemma 3 to set up a linesearch on a sum of squares optimization problem.

If we follow the approach given in Lemma 3 for a set of \( d \) values we can construct the table of maximal \( \gamma \) values given in Table 2.2.1. The table shows that for \( d = 4 \) we can demonstrate global exponential stability, since \( \gamma_{\text{max}} > 0 \), and that the state decays with rate \( \gamma/d = 0.0013 \). As the degree of the Lyapunov function is increased, the maximum feasible value for \( \gamma \) increases, with \( \gamma/d \) increasing as well. For \( d > 10 \), the numerical errors in solving the large resulting LMIs cause the linesearch to become erratic and return lower values for \( \gamma_{\text{max}} \).

### 2.3 State Feedback Controller Design

As shown in the previous section, sufficient conditions for the the assumptions of Theorems 8 and 9 can be checked as LMIs or linesearches on LMIs. The existence of efficient tests for stability makes us wonder if similar techniques can be used, not just for analysis, but also for controller synthesis. Consider now the system

\[
\dot{x} = f(x) + g(x)u \tag{2.3}
\]

for \( x \in \mathbb{R}^n \) with \( f \in \mathcal{R}_n \), \( f(0) = 0 \) and \( u \in \mathbb{R}^m \) with \( g \in \mathcal{R}_m^{n \times m} \). If we allow \( u \) to be generated by a state feedback controller \( K \in \mathcal{R}_n^m \) with \( K(0) = 0 \), we get the following closed loop system

\[
\dot{x} = f(x) + g(x)K(x) \tag{2.4}
\]
where $K$ is still unknown.

Now we can look for conditions on $K$ such that we can find a Lyapunov equation that meets the assumptions for Theorems 8 and 9. The analog for Lemmas 2 and 3 for the system (2.4) are as follows.

**Lemma 4** Given the system (2.4) and fixed positive definite functions $l_1, l_2 \in \mathcal{R}_n$, the system is globally asymptotically stable if there exists $V \in \mathcal{R}_n$ with $V(0) = 0$ and $K \in \mathcal{R}_n^m$ with $K(0) = 0$ such that

\[
V - l_1 \in \Sigma_n \\
-(\nabla V^*(f + gK) + l_2) \in \Sigma_n
\]

**Lemma 5** Given the system (2.4) and the fixed positive definite function $l(x) = \|x\|_d^d$ with $d$ an integer greater than one, the system is globally exponentially stable if there exists $\gamma > 0$, $K \in \mathcal{R}_n^m$ with $K(0) = 0$ and $V \in \mathcal{R}_n$ with $V(0) = 0$ such that

\[
V - l \in \Sigma_n \\
-(\gamma V + \nabla V^*(f + gK)) \in \Sigma_n
\]

Additionally, the rate of convergence is $\gamma/d$.

The proofs for these lemmas follow exactly along the lines of the proofs of Lemmas 2 and 3. However, since both lemmas have conditions that are bilinear in the monomials of $V$ and $K$, neither the SOS conditions of Lemma 4 nor those of 5 can be checked directly with semidefinite programming, unless the dynamics and controller are linear and the Lyapunov function is quadratic. In the linear/quadratic case the standard “feedback trick” can be used to change variables and it yields LMI conditions, see §7.2.1 in [4].

### 2.3.1 Iterative State Feedback Design Algorithms

Since, in general, Lemmas 4 and 5 are not amenable to the semidefinite programming based approach of Theorem 3, we will need to employ an iterative approach that solves the lemmas’ SOS conditions in $V$ and $K$ by holding one of these polynomials fixed while adjusting the other.
Consider the conditions for Lemma 4; the second SOS condition is bilinear in $V$ and $K$ as noted above, which indicates that if either is fixed it is an LMI feasibility problem to find the other. The problem with this approach is that if $V$ is fixed and the resulting convex search for $K$ fails, then there is no way to redesign $V$ from the search on $K$. The same pitfall occurs if $K$ is held fixed and the search is for $V$. This approach gives a single shot at finding a controller for a given Lyapunov function or Lyapunov function for a fixed controller, but it cannot be extended to search for both.

The conditions for Lemma 5 have better feasibility properties that allow us to propose an iterative design procedure for controller and Lyapunov function. The first step is to find the linearization of the system (2.4) about the equilibrium point $f(0) = 0$,

$$\dot{x} = A_f x + B_g u$$  \hspace{1cm} (2.5)

where $A_f$ and $B_g$ are the linearizations of $f$ and $g$ respectively.

It is possible to use the “feedback trick” to pose the problem of finding a linear static state feedback controller, $u = K_{\text{lin}}(x) = K x$, for system (2.5) with a quadratic Lyapunov function that demonstrates global exponential stability, $V_{\text{lin}}(x) = x^* P x$, as the following linesearch with LMI constraints

$$\max \gamma$$

s.t.

$$Q > 0$$

$$-\gamma Q \geq QA_f^* + A_f Q + L^* B_g^* + B_g L$$  \hspace{1cm} (2.6)

where $P = Q^{-1}$ and $K = LP$. If a positive value of $\gamma$ is obtainable, then $K_{\text{lin}}$ and $V_{\text{lin}}$ show that the linearized system (2.5) is exponentially stable.

We can now use the solutions to the linearized version of Lemma 5 given as the linesearch in (2.6) to start our iterative search for solutions to the SOS conditions for the full nonlinear problem.

**Algorithm 1 (State Feedback Controller Design)** A search for $K$ and $V$ to satisfy the SOS conditions of Lemma 5.
1. Initialize the problem with $K = K_{\text{lin}}$ and $V = V_{\text{lin}}$.

2. Solve the semidefinite program

$$
\begin{align*}
\max \quad & \gamma \\
\text{s.t.} \quad & -(\gamma V + \nabla V^*(f + gK)) \in \Sigma_n
\end{align*}
$$

(2.7)

If the maximal $\gamma$, $\gamma_{\text{max}}$, is strictly positive, then the system (2.4) is globally exponentially stable with controller $K_{\text{lin}}$, if $\gamma_{\text{max}} \leq 0$ go to step 3.

3. Fix the degree of the controller polynomial, and hold the Lyapunov function $V$ fixed. Solve the semidefinite programming problem where $K \in \mathbb{R}_n^m$ with $K(0) = 0$

$$
\begin{align*}
\max_K \quad & \gamma \\
\text{s.t.} \quad & -(\gamma V + \nabla V^*(f + gK)) \in \Sigma_n
\end{align*}
$$

(2.8)

If $\gamma_{\text{max}} > 0$, then the system (2.4) is globally exponentially stable with controller $K$, if $\gamma_{\text{max}} \leq 0$ go to step 4.

4. Fix the degree of the Lyapunov function and denote it $d$, while holding the controller polynomial $K$ fixed. Set $l(x) = \|x\|_d^d$ and solve the following linesearch on $\gamma$ where $V \in \mathbb{R}_n$ with $V(0) = 0$

$$
\begin{align*}
\max_V \quad & \gamma \\
\text{s.t.} \quad & V - l \in \Sigma_n \\
& -(\gamma V + \nabla V^*(f + gK)) \in \Sigma_n
\end{align*}
$$

(2.9)

If $\gamma_{\text{max}} > 0$, then the system (2.4) is globally exponentially stable with controller $K$, if $\gamma_{\text{max}} \leq 0$ repeat step 3.

Note that if any $\gamma$ can be found that makes (2.7) feasible, then all of the optimization problems will remain feasible through the iteration, however, this does not mean
that a $\gamma_{\text{max}} > 0$ will necessarily be found. Additionally, there are many variants on this algorithm that would work to the same ends; since the controller iteration (2.8) need not depend on $K_{\text{lin}}$ we could find only $V_{\text{lin}}$ and start directly from step 3.

### 2.3.2 State Feedback Design Example

We will design a state feedback controller following Algorithm 1 for the following spring-mass system given below where $x_1, x_3, x_5$ represent the displacement of $m_1, m_2, m_3$ respectively, the $k$’s identify the springs and $u$ is a forcing input.

![Spring-mass system diagram]

For simplicity we will consider only unit masses, $m_1 = m_2 = m_3 = 1$. If we let $k_1$ be a stiffening spring that generates the location dependent force

$$F_{k_1}(x) = x_1 + \frac{1}{10} x_1^3$$

and let the forces generated by $k_2, k_3$ be linear in their displacement

$$F_{k_2} = x_3 - x_1$$

$$F_{k_3} = x_5 - x_3$$

we can write the system’s dynamics as

$$\begin{bmatrix}
\dot{x}_1 \\
\dot{x}_2 \\
\dot{x}_3 \\
\dot{x}_4 \\
\dot{x}_5 \\
\dot{x}_6
\end{bmatrix} = \begin{bmatrix}
x_2 \\
-(x_1 + \frac{1}{10} x_1^3) + (x_3 - x_1) \\
x_4 \\
-(x_3 - x_1) + (x_5 - x_3) \\
x_6 \\
-(x_5 - x_3)
\end{bmatrix} f(x) + \begin{bmatrix}
0 \\
1 \\
0 \\
0 \\
0 \\
0
\end{bmatrix} u$$

$$\begin{bmatrix}
0 \\
0 \\
0 \\
0 \\
0 \\
0
\end{bmatrix} g(x)$$

(2.10)
The linearization of the system (2.10) has purely imaginary eigenvalues, so no conclusions about its stability can be made.

We want to design a state feedback controller, \( u = K(x) \), to globally exponentially stabilize the system

\[
\dot{x} = f(x) + g(x)u
\]

where \( f, g \) are defined in (2.10). Following Algorithm 1, we first solve the linesearch (2.6) to find \( K_{\text{lin}} \) and \( V_{\text{lin}} \). With this Lyapunov function and controller the semidefinite program (2.7) is infeasible, which by convention is denoted by \( \gamma_{\text{max}} = -\infty \).

For step 3 of the algorithm we need to fix the degree of the controller polynomial, and since the system has a matched nonlinearity, meaning that the control enters where it can directly counter the nonlinearity, we fix \( \deg(K) = 3 \). When we solve the semidefinite program (2.8) we find \( \gamma_{\text{max}} = 1.14 \), which proves that the closed loop system is globally exponentially stable; additionally, the controller found does not simply cancel the nonlinearity.

The controller is a degree three polynomial in 6 variables, so it has 83 terms. Since no part of the optimization tries to reduce the number of nonzero terms, all of them are used. If all the terms whose coefficients are less than \( 10^{-6} \) in absolute value are set to zero the number of used terms drops to 62, and the reduced controller continues to make the system globally exponentially stable with the same value of \( \gamma_{\text{max}} \).

Additionally, if, even though we have found a value of \( \gamma > 0 \), we continue the steps of Algorithm 1 in a effort to make the exponential convergence even faster we can improve on the value of \( \gamma_{\text{max}} = 1.14 \) found above. The increase in \( \gamma_{\text{max}} \) for 3 sets of controller and Lyapunov function iterations are shown in Figure 2.1.

### 2.4 Output Feedback Controller Design

We can expand the results for state feedback controllers by allowing the controller to be a dynamic system. Again we will be searching for ways to validate the assumptions of Theorem 9 using an iterative procedure to check SOS conditions. As in the state feedback case, we will not consider the globally asymptotic stability case, since
Define the system to be controlled as

\[
\begin{align*}
\dot{x} &= f(x) + g(x)u \\
y &= h(x)
\end{align*}
\] (2.11)

for \(x \in \mathbb{R}^n\) with \(f \in \mathcal{R}_n^m\), \(f(0) = 0\), \(u \in \mathbb{R}^m\), \(g \in \mathcal{R}_{n \times m}^n\) and \(y \in \mathbb{R}^p\) with \(h \in \mathcal{R}_p^m\). If we allow \(u\) to be generated by an unknown \(n_\xi\)-state dynamic output feedback controller of the form

\[
\begin{align*}
\dot{\xi} &= A(\xi) + B(\xi)y \\
u &= C(\xi) + D(\xi)y
\end{align*}
\] (2.12)

for \(\xi \in \mathbb{R}^{n_\xi}\) with \(A \in \mathcal{R}_{n_\xi}^{n_\xi}\), \(A(0) = 0\), \(B \in \mathcal{R}_{n_\xi \times p}^{n_\xi}\), \(C \in \mathcal{R}_{n_\xi}^m\) and \(D \in \mathcal{R}_{n_\xi \times p}^m\). With this controller structure, the closed loop system becomes

\[
\begin{align*}
\dot{x} &= f(x) + g(x)(C(\xi) + D(\xi)h(x)) \\
\dot{\xi} &= A(\xi) + B(\xi)h(x).
\end{align*}
\] (2.13)

To test for global exponential stability we can pose the following analog of Lemma 3 for the combined system (2.13).
Lemma 6 Given the system (2.11), a controller of the form (2.12) with \( n_\xi \) some fixed positive integer, and \( l([x;\xi]) = \|x;\xi\|_d^d \) with \( d \) some integer greater than one, then the system (2.13) is globally exponentially stable if there exists \( \gamma > 0 \), \( A \in \mathbb{R}^{n_\xi} \), \( A(0) = 0 \), \( B \in \mathbb{R}^{n_\xi \times p} \), \( C \in \mathbb{R}^{m \times n_\xi} \), \( D \in \mathbb{R}^{m \times p} \) and \( V \in \mathbb{R}^{n+n_\xi} \) with \( V(0) = 0 \) such that

\[
V - l \in \Sigma_{n+n_\xi} \subseteq \Sigma_{n+n_\xi} \\
- \left( \gamma V + \nabla V^* \begin{bmatrix} f + gC + gDh \\ A + Bh \end{bmatrix} \right) \in \Sigma_{n+n_\xi}
\]

Additionally, the rate of convergence is \( \gamma/d \).

As before this lemma can be proved along the lines of Lemma 2 and again it is highly bilinear in the elements of the controller system and the Lyapunov function. However, unlike the earlier lemmas, when the systems are linear and the Lyapunov function is quadratic the SOS conditions in Lemma 6 still can not be checked with a single semidefinite program.

2.4.1 Iterative Output Feedback Design Algorithm

In their stated forms, the SOS conditions of Lemma 6 do not fit into the set up for Theorem 3 since they are bilinear in the controller system and the Lyapunov function. To get around this problem we will use an iterative approach as in Algorithm 1.

Since there is no way to check the SOS conditions of Lemma 6 in the case of linear dynamics and quadratic Lyapunov function we will not be able to start the algorithm from the solution to the linearized problem as we could in Algorithm 1.

Algorithm 2 (Output Feedback Controller Design) A search for \( A, B, C, D \) and \( V \) to satisfy the SOS conditions of Lemma 6.

1. Fix the controller’s state dimension: \( n_\xi \).

2. Find a linear controller \( (A, B, C, D) \) that stabilizes the linearization of the system (2.11) with a quadratic Lyapunov function.
3. Solve the semidefinite program

\[ \max_{\gamma} \gamma \]
\[ \text{s.t.} \quad - \left( \gamma V + \nabla V^* \begin{bmatrix} f + gC + gDh \\ A + Bh \end{bmatrix} \right) \in \Sigma_{n+n_\xi} \]  

(2.14)

If the maximal \( \gamma, \gamma_{\text{max}} \), is strictly positive then the system (2.11) is exponentially stabilized by the linear controller \((A, B, C, D)\), if \( \gamma_{\text{max}} \leq 0 \) go to step 4.

4. Fix the degrees of the controller polynomials, and hold the Lyapunov function fixed. Solve the semidefinite programming problem where \( A \in \mathcal{R}^{n_\xi}_{n_\xi} \) with \( A(0) = 0 \), \( B \in \mathcal{R}^{n_\xi \times p}_{n_\xi} \), \( C \in \mathcal{R}^{m}_{n_\xi} \), and \( D \in \mathcal{R}^{m \times p}_{n_\xi} \)

\[ \max_{A,B,C,D} \gamma \]
\[ \text{s.t.} \quad - \left( \gamma V + \nabla V^* \begin{bmatrix} f + gC + gDh \\ A + Bh \end{bmatrix} \right) \in \Sigma_{n+n_\xi} \]  

(2.15)

If \( \gamma_{\text{max}} > 0 \) then the system (2.11) is exponentially stabilized by the controller \((A, B, C, D)\), if \( \gamma_{\text{max}} \leq 0 \) go to step 5.

5. Fix the degree of the Lyapunov function and denote it \( d \) while holding the controller fixed. Set \( l([x; \xi]) = \|[x; \xi]\|^2_d \) and solve the following linesearch on \( \gamma \) where \( V \in \mathcal{R}_{n+n_\xi} \) with \( V(0) = 0 \)

\[ \max_{V} \gamma \]
\[ \text{s.t.} \quad V - l \in \Sigma_{n+n_\xi} \]  

(2.16)

If \( \gamma_{\text{max}} > 0 \) then the system (2.11) is exponentially stabilized by the controller \((A, B, C, D)\), if \( \gamma_{\text{max}} \leq 0 \) repeat step 4.
As with the state feedback controller design algorithm, there are many variants of this algorithm that could be used that would use the same two iterative steps: controller design and Lyapunov function design.

### 2.4.2 Output Feedback Design Example

Consider again the spring mass system used in §2.3.2, whose dynamics are given by (2.10). If we make the system’s output

\[ y = h(x) = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \]

the problem fits the form of (2.11), so we can design an output feedback controller for the system using Algorithm 2.

First, we need to fix the state dimension of the controller. Since the size of the semidefinite programs that are used to check if a polynomial is SOS grow rapidly with the number of variables, it is expedient to pick a low state dimension for the controller. In this light, we fix \( n_\xi = 2 \).

Looking at the root locus plots of the linearized system we choose a linear controller consisting of the sum of a lead filter on the position measurement and a single pole on the velocity measurement.

\[
\begin{bmatrix}
\dot{\xi}_1 \\
\dot{\xi}_2 
\end{bmatrix} = \begin{bmatrix}
-4 & 0 \\
0 & -4 
\end{bmatrix} \begin{bmatrix}
\xi_1 \\
\xi_2 
\end{bmatrix} + \begin{bmatrix}
8 & 0 \\
0 & 4 
\end{bmatrix} \begin{bmatrix}
y_1 \\
y_2 
\end{bmatrix}
\]

\[
u = \begin{bmatrix}
8 & -4 \\
0 & -16 
\end{bmatrix} \begin{bmatrix}
\xi_1 \\
\xi_2 
\end{bmatrix} + \begin{bmatrix}
8 \\
-4 
\end{bmatrix} \begin{bmatrix}
y_1 \\
y_2 
\end{bmatrix}
\]

This linear controller stabilizes the system’s linearization which allows us to find a quadratic Lyapunov function for the linearized system.

Following §2.3.2 we pick all of the controller polynomials to be degree three and since they are only in two variables they only have 10 terms. From this point we can begin the controller and Lyapunov function iteration, however, (2.14) is infeasible so at the end of step 3 we have \( \gamma_{\max} = -\infty \).
In the controller iteration for step 4, we find $\gamma_{\text{max}} = -1.199$. If we fix the Lyapunov function to be quadratic (36 terms), then in step 5 we achieve $\gamma_{\text{max}} = 0.048$. Since $\gamma_{\text{max}} > 0$ we know that the controller, given below with terms of magnitude less than $10^{-6}$ removed, does indeed globally exponentially stabilize the system.

$$A(\xi) = \begin{bmatrix} -7.72\xi_1^3 - 0.59\xi_1^2\xi_2 - 5.09\xi_1\xi_2^2 - 4.44\xi_2^3 - 12.24\xi_1 - 2.15\xi_2 \\ -14.92\xi_1^3 - 1.33\xi_1^2\xi_2 - 9.75\xi_1\xi_2^2 - 8.76\xi_2^3 - 16.06\xi_1 - 8.75\xi_2 \end{bmatrix}$$

$$B(\xi) = \begin{bmatrix} 2.26\xi_1^2 - 0.83\xi_1\xi_2 + 2.40\xi_2^2 + 8.55 & -3.65\xi_1^2 + 2.26\xi_1\xi_2 - 3.79\xi_2^2 - 5.64 \\ 4.28\xi_1^2 - 1.61\xi_1\xi_2 + 4.47\xi_2^2 + 5.73 & -7.36\xi_1^2 + 4.33\xi_1\xi_2 - 7.45\xi_2^2 - 8.77 \end{bmatrix}$$

$$C(\xi) = \begin{bmatrix} 0.44\xi_1^3 - 0.17\xi_1^2\xi_2 + 0.34\xi_1\xi_2^2 + 0.08\xi_2^3 + 8.6815\xi_1 - 4.1818\xi_2 \end{bmatrix}$$

$$D(\xi) = \begin{bmatrix} -0.17\xi_1^2 + 0.07\xi_1\xi_2 - 0.27\xi_2^2 - 11.85 & -0.10\xi_1^2 - 0.17\xi_1\xi_2 + 0.06\xi_2^2 - 1.2557 \end{bmatrix}$$

We could continue iterating to find larger values for $\gamma$, however, step 5 checks two SOS conditions that result in a very large semidefinite program. The first term is a homogeneous quadratic polynomial in 6 variables (36 terms), while the second term is a degree five polynomial without linear and constant terms in 8 variables (1278 terms). The size of the resulting semidefinite program causes it to very quickly become numerically unstable as the iteration progresses.
### Appendix A

#### Semidefinite Programming

Semidefinite programming (SDP) considers the following optimization problem

\[
\min_{x \in \mathbb{R}^n} \ c^*x \\
\text{s.t. } F(x) := F_0 + x_1F_1 + \cdots + x_nF_n \succeq 0
\]

with \(c \in \mathbb{R}^n\), and \(F_i = F_i^* \in \mathbb{R}^{p \times p}\) for \(i = 0, 1, \ldots, n\). The most important theoretical property of SDP is its convexity, which allows it to be solved numerically with great efficiency [22]. SDP has many other useful theoretical properties that are covered in detail in the survey [23].

Often SDP is used to solve the feasibility problem: does there exist \(x \in \mathbb{R}^n\) such that \(F(x) \succeq 0\)? The generalized inequality \(F(x) \succeq 0\) is linear, or strictly speaking affine, in \(x\), so the feasibility problem is often referred to as a linear matrix inequality (LMI). One property of LMIs is that a set of them can be turned into single larger block diagonal LMI.

**Example 4** Consider the LMIs \(F(x) \succeq 0\) and \(G(x) \succeq 0\) they can be written as

\[
\begin{bmatrix}
G(x) \\
F(x)
\end{bmatrix} = \begin{bmatrix}
G_0 & \\
F_0
\end{bmatrix} + x_1 \begin{bmatrix}
G_1 & \\
F_1
\end{bmatrix} + \cdots + x_n \begin{bmatrix}
G_n & \\
F_n
\end{bmatrix} \succeq 0
\]

To illustrate a less obvious use of LMIs consider the following Lyapunov stability argument.
Example 5 Consider the linear system $\dot{x} = Ax$ with $x \in \mathbb{R}^n$ and the Lyapunov function $V(x) = x^*Px$, with $P$ unknown. If a $P$ can be picked such that $V(x)$ is positive definite and $-\dot{V}(x)$ is positive semidefinite, then the system is stable. Noting that $\dot{V}(x) = x^*(A^*P + PA)x$, the problem can be posed as: does there exist a $P$ such that

$$
P > 0$$

$$-(A^*P + PA) \succeq 0$$

If $P$ is represented on the standard basis as $\sum_{i=1}^{m} p_i E_i$, with $m := (n + 1)n/2$, then the LMIs above become

$$
\begin{bmatrix}
\sum_{i=1}^{m} p_i E_i - \epsilon I \\
-(A^* \sum_{i=1}^{m} p_i E_i + \sum_{i=1}^{m} p_i E_i A)
\end{bmatrix} \succeq 0
$$

for any $\epsilon > 0$. The question of what value to use for $\epsilon$ can also be incorporated to give the following SDP in the variables $\epsilon, p_1, \ldots, p_m$

$$\max \epsilon$$

s.t. $p_1 \begin{bmatrix} E_1 \\ -(A^*E_1 + E_1 A) \end{bmatrix} + \ldots + p_m \begin{bmatrix} E_m \\ -(A^*E_m + E_m A) \end{bmatrix} + \epsilon \begin{bmatrix} -I \\ 0 \end{bmatrix} \succeq 0$

which, if the maximum value for $\epsilon$ is strictly positive, gives a matrix $P$ such that the Lyapunov function $V$ demonstrates the stability of $A$. 

Bibliography


