Algebraic Relaxations and Hardness Results in Polynomial Optimization and Lyapunov Analysis

by

Amir Ali Ahmadi

B.S., Electrical Engineering, University of Maryland, 2006
B.S., Mathematics, University of Maryland, 2006
S.M., Electrical Engineering and Computer Science, MIT, 2008

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Electrical Engineering and Computer Science
at the Massachusetts Institute of Technology

September 2011

Signature of Author: ______________________________

Department of Electrical Engineering and Computer Science
September 2, 2011

Certified by: ______________________________

Pablo A. Parrilo
Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by: ______________________________

Leslie A. Kolodziejski
Professor of Electrical Engineering
Chair, Committee for Graduate Students

Algebraic Relaxations and Hardness Results in Polynomial Optimization and Lyapunov Analysis

by

Amir Ali Ahmadi

Submitted to the Department of Electrical Engineering and Computer Science on September 2, 2011 in partial fulfillment of the requirements for the degree of Doctor of Philosophy

Abstract

The contributions of the first half of this thesis are on the computational and algebraic aspects of convexity in polynomial optimization. We show that unless P=NP, there exists no polynomial time (or even pseudo-polynomial time) algorithm that can decide whether a multivariate polynomial of degree four (or higher even degree) is globally convex. This solves a problem that has been open since 1992 when N. Z. Shor asked for the complexity of deciding convexity for quartic polynomials. We also prove that deciding strict convexity, strong convexity, quasiconvexity, and pseudoconvexity of polynomials of even degree four or higher is strongly NP-hard. By contrast, we show that quasiconvexity and pseudoconvexity of odd degree polynomials can be decided in polynomial time.

We then turn our attention to sos-convexity—an algebraic sum of squares (sos) based sufficient condition for polynomial convexity that can be efficiently checked with semidefinite programming. We show that three natural formulations for sos-convexity derived from relaxations on the definition of convexity, its first order characterization, and its second order characterization are equivalent. We present the first example of a convex polynomial that is not sos-convex. Our main result then is to prove that the cones of convex and sos-convex polynomials (resp. forms) in n variables and of degree d coincide if and only if $n = 1$ or $d = 2$ or $(n, d) = (2, 4)$ (resp. $n = 2$ or $d = 2$ or $(n, d) = (3, 4)$). Although for disparate reasons, the remarkable outcome is that convex polynomials (resp. forms) are sos-convex exactly in cases where nonnegative polynomials (resp. forms) are sums of squares, as characterized by Hilbert in 1888.

The contributions of the second half of this thesis are on the development and analysis of computational techniques for certifying stability of uncertain and nonlinear dynamical systems. We show that deciding asymptotic stability of homogeneous cubic polynomial vector fields is strongly NP-hard. We settle some of the converse questions on existence of polynomial and sum of squares Lyapunov functions. We present a globally asymptotically stable polynomial vector field with no polynomial Lyapunov function. We show via an explicit counterexample that if the degree of the polynomial Lyapunov function is fixed, then sos programming can fail to find a valid Lyapunov function even though one exists. By contrast, we show that if the degree is allowed to increase, then existence of a polynomial Lyapunov function for a planar or a homogeneous polynomial vector field implies existence of a polynomial Lyapunov function that can be found with sos programming. We extend this result to develop a converse sos Lyapunov theorem for robust stability of switched linear systems.

In our final chapter, we introduce the framework of path-complete graph Lyapunov functions for approximation of the joint spectral radius. The approach is based on the analysis of the underlying switched system via inequalities imposed between multiple Lyapunov functions associated to a labeled directed graph. Inspired by concepts in automata theory and symbolic dynamics, we define a class of graphs called path-complete graphs, and show that any such graph gives rise to a method for proving stability of switched systems. The semidefinite programs arising from this technique include as special case many of the existing methods such as common quadratic, common sum of squares, and maximum/minimum-of-quadratics Lyapunov functions. We prove approximation guarantees for analysis via several families of path-complete graphs and a constructive converse Lyapunov theorem for maximum/minimum-of-quadratics Lyapunov functions.

Thesis Supervisor: Pablo A. Parrilo
Title: Professor of Electrical Engineering and Computer Science

To my parents, Maryam and Hamid Reza

به یاد روزهای خوب

Contents

Acknowledgements

The opportunities that have been available to me as a graduate student at MIT have been endless, but without a doubt, the greatest opportunity of all has been the chance to work with my advisor Pablo Parrilo. What inspires me most about Pablo—aside from his well-known traits like superior intelligence, humility, and professional attitude—is his never-ending passion for understanding things deeply. The joy that Pablo takes in interpreting a mathematical result from all different angles, placing it in the right level of abstraction, and simplifying it to the point that it cannot be made simpler is a virtue that I hope to take with me throughout my career. As Feynman once said, "The truth always turns out to be simpler than you thought." On several occasions in this thesis, Pablo's insights, or at times simply his questions, have made me realize this fact, and for that I am very grateful.

I would also like to thank Pablo for creating the perfect environment for me to pursue my research ideas. Not once did he ask me to work on a problem that I didn't choose to work on myself, not once did he inquire about a research result before I felt ready to present my progress, and not once did I ever have to worry about funding for my research or for going to various conferences. Pablo's approach was to meet with me regularly during my Master's studies, but to be more hands-off throughout my Ph.D. years. This worked out perfectly. I recall as a child, my father taught me how to ride a bicycle by running alongside my bike and holding on to its back, but then gradually letting go of his hands (without me realizing) so I could ride on my own. I feel that Pablo has very much done the same thing in the past few years in placing me on the path to becoming a great researcher. I will be grateful to him for as long as I continue on this ride.

I couldn't have asked for better thesis committee members than Vincent Blondel and John Tsitsiklis. Among their many other accomplishments, Vincent and John are two of the pioneering figures in complexity theory in control and optimization, a subject that as parts of this thesis reflect has become of much interest to me over the last couple of years. From discussions about complexity and the joint spectral radius to conversations about my business idea for the MIT $100K Entrepreneurship Competition, both Vincent and John have always been generous with their time and advice. In the case of Vincent, I was very fortunate that my last year at LIDS coincided with the year that he joined us as a visiting faculty member from Belgium. Some of the most creative talks I have ever attended have been given by Vincent. (A memorable example is in his LIDS seminar talk on *privacy in social networks*, where he fooled me into believing that his wife, Gilbert Strang, and Mark Zuckerberg sent him "private" messages which we saw popping up on the screen as he was speaking!) I am thankful to Vincent for sincerely caring about my thesis work, either in the form of a call from Belgium providing encouragement the day prior to my defense, or by deriving valuable results related to this thesis even after the completion of my defense.

My acquaintance with John goes back to my first year at MIT. I remember walking out of the first lecture of his probability course telling myself that I should attend every class that is taught by this professor. Surely, I have done that. John has an amazing ability to make every concept look simple and intuitive. I am indebted to him for everything he has taught me both in and out of the classroom. I would also like to thank him for his invaluable contributions to a joint work that led to Chapter 2 of this thesis.

My gratitude extends to my other coauthors: Gerg Blekherman, Raphaël Jungers, Miroslav Krstic, Alex Olshevsly, and Mardavij Roozbehani. I have learned a great deal from interactions with all of them. I am thankful to Greg for settling two (out of two!) mathematical bets that I made with Markus Schweighofer at a meeting in 2009. Had he not done that, it is not clear how long it would have taken me to get over my obsession with those problems and move on to the problems addressed in this thesis. (I lost one of the two bets when Greg proved that polynomials with a certain special property exist [26]; interestingly, no explicit examples are known to this day, see Section 3.6.) Raphaël, Alex, and Mardavij have all been great friends throughout my years at LIDS. I particularly want to express my gratitude to Mardavij for always being there when I needed someone to talk to, and to Raphaël for his nice gesture in giving me a copy of his book [85] as a gift in our first meeting, which he kindly signed with the note: "Let's collaborate!" I am grateful to Miroslav for hosting me (together with Tara Javidi) at the Cymer Center for Control Systems and Dynamics at UCSD, which led to immediate and fruitful collaborations between us.

This thesis has benefited from my interactions with several distinguished researchers outside of MIT. Among them I would like to mention Amitabh Basu, Stephen Boyd, Etienne de Klerk, Jesús De Loera, Bill Helton, Monique Laurent, Jiawang Nie, Bruce Reznick, Claus Scheiderer, Eduardo Sontag, Bernd Sturmfels, and André Tits. I had the good fortune to TA the Convex Optimization class that

Stephen co-taught with Pablo while visiting MIT from Stanford. I am grateful to Amitabh and Jesús for hosting me at the Mathematics Dept. of UC Davis, and to Eduardo for having me at the SontagFest in DIMACS. These have all been memorable and unique experiences for me. I wish to also thank Nima Moshtagh for the internship opportunity at Scientific Systems.

I am deeply grateful to the faculty at LIDS, especially Sasha Megretski, Asu Ozdaglar, Munther Dahleh, and Emilio Frazzoli for making LIDS a great learning environment for me. I have also learned a great deal from classes offered outside of LIDS, for example from the beautiful lectures of Michael Sipser on Theory of Computation, or from the exciting course of Constantinos Daskalakis on Algorithmic Game Theory. Special thanks are also due to members of the LIDS staff. I am grateful to Lisa Gaumond for her friendly smile and for making sure that I always got reimbursed on time, to Jennifer Donovan for making organizational tasks easy, and to Brian Jones for going out of his way to fix my computer twice when I was on the verge of losing all my files. Janet Fischer from the EECS graduate office and my academic advisor, Vladimir Stojanovic, are two other people that have always helped me stay on track.

This is also a good opportunity to thank my mentors at the University of Maryland, where I spent four amazing years as an undergraduate student. Thomas Antonsen, Thomas Murphy, Edward Ott, Reza Salem, André Tits, and (coach!) James Yorke helped set the foundation on which I am able to do research today.

Some of the results of this thesis would not have been possible without the use of software packages such as YALMIP [98], SOSTOOLS [132], and SeDuMi [157]. I am deeply grateful to the people who wrote these pieces of software and made it freely available. Special thanks go to Johan Löfberg for his patience in answering my questions about YALMIP.

My best memories at MIT are from the moments shared with great friends, such as, Amir Khandani, Emmanuel Abbe, Marco Pavone, Parikshit Shah, Noah Stein, Georgios Kotsalis, Borjan Gagoski, Yola Katsargyri, Mitra Osqui, Ermin Wei, Hoda Eydgahi, Ali ParandehGheibi, Sertac Karaman, Michael Rinehart, Mark Tobenkin, John Enright, Mesrob Ohannessian, Rose Faghih, Ali Faghih, Sidhant Misra, Aliaa Atwi, Venkat Chandrasekaran, Ozan Candogan, James Saunderson, Dan Iancu, Christian Ebenbauer, Paul Njoroge, Alireza Tahbaz-Salehi, Kostas Bimpikis, Ilan Lobel, Stavros Valavani, Spyros Zoumpoulis, and Kimon Drakopoulos. I particularly want to thank my longtime officemates, Pari and Noah. Admittedly, I didn't spend enough time in the office, but whenever I did, I had a great time in their company. Together with Amir and Emmanuel, I had a lot of fun exploring the night life of Boston during my first two years. Some of the memories there are unforgettable. I am thankful to Marco for making everyday life at MIT more enjoyable with his sense of humor. His presence was greatly missed during my last year when NASA decided to threaten the security of all humans on Earth and aliens in space by making the disastrous mistake of hiring him as a Research Technologist. I also owe gratitude to Marco's lovely wife-to-be, Manuela, for her sincere friendship. Unfortunately, just like NASA, Manuela is a victim of Marco's deception.

Aside from friends at MIT, my childhood friends who live in Iran and Washington DC have made a real effort to keep our friendships close by paying frequent visits to Boston, and for that I am forever grateful. I would also like to thank the coaches of the MIT tennis team, Dave Hagymas, Spritely Roche, and Charlie Maher, for always welcoming me on practice sessions, which allowed me to continue to pursue my childhood passion at MIT and maintain a balanced graduate life.

The work in this thesis was partially supported by the NSF Focused Research Group Grant on Semidefinite Optimization and Convex Algebraic Geometry DMS-0757207, and by AFOSR MURI subaward 07688-1.

My heart is full of gratitude for my parents, Maryam and Hamid Reza, my sister, Shirin, and my brother-in-law, Karim, who have always filled my life with unconditional love and support. I am so sorry and embarrassed for all the times I have been too "busy" to return the love, and so grateful to you for never expecting me to do so. "Even after all this time the sun never says to the earth, 'you owe me.' Look what happens with a love like that. It lights the whole sky."

I finally want to thank my girlfriend and best friend, Margarita, who has the heart of an angel and who has changed my life in so many ways since the moment we met. As it turns out, my coming to Boston was not to get a Ph.D. degree, but to meet you. "Who could be so lucky? Who comes to a lake for water and sees the reflection of moon."

Chapter 1

Introduction

With the advent of modern computers in the last century and the rapid increase in our computing power ever since, more and more areas of science and engineering are being viewed from a computational and algorithmic perspective—the field of optimization and control is no exception. Indeed, what we often regard nowadays as a satisfactory solution to a problem in this field—may it be the optimal allocation of resources in a power network or the planning of paths of minimum fuel consumption for a group of satellites—is an *efficient algorithm* that when fed with an instance of the problem as input, returns in a reasonable amount of time an output that is guaranteed to be optimal or near optimal.

Fundamental concepts from theory of computation, such as the notions of a Turing machine, decidability, polynomial time solvability, and the theory of NP-completeness, have allowed us to make precise what it means to have an (efficient) algorithm for a problem and much more remarkably to even be able to prove that for certain problems such algorithms do not exist. The idea of establishing "hardness results" to provide rigorous explanations for why progress on some problems tends to be relatively unsuccessful is commonly used today across many disciplines and rightly so. Indeed, when a problem is resisting all attempts for an (efficient) algorithm, little is more valuable to an unsatisfied algorithm designer than the ability to back up the statement "I cannot do it" with the claim that "it cannot be done".

Over the years, the line between what can or cannot be efficiently computed has shown to be a thin one. There are many examples in optimization and control where complexity results reveal that two problems that on the surface appear quite similar have very different structural properties. Consider for example the problem of deciding given a symmetric matrix Q, whether x^TQx is nonnegative for all $x \in \mathbb{R}^n$, and contrast this to the closely related problem of deciding whether x^TQx is nonnegative for all x's in $\mathbb{R}^n$ that are elementwise nonnegative. The first problem, which is at the core of semidefinite programming, can be answered in polynomial time (in fact in $O(n^3)$), whereas the second problem, which forms the basis of copositive programming, is NP-hard and can easily encode many hard combinatorial problems [109]. Similar scenarios arise in control theory. An interesting example is the contrast between the problems of deciding stability of interval polynomials and interval matrices. If we are given a single univariate polynomial of degree n or a single $n \times n$ matrix, then standard classical results enable us to decide in polynomial time whether the polynomial or the matrix is (strictly) stable, i.e, has all of its roots (resp. eigenvalues) in the open left half complex plane. Suppose now that we are given lower and upper bounds on the coefficients of the polynomial or on the entries of the matrix and we are asked to decide whether all polynomials or matrices in this interval family are stable. Can the answer still be given in polynomial time? For the case of interval polynomials, Kharitonov famously demonstrated [87] that it can: stability of an interval polynomial can be decided by checking whether *four* polynomials obtained from the family via some simple rules are stable. One may naturally speculate whether such a wonderful result can also be established for interval matrices, but alas, NP-hardness results [110] reveal that unless P=NP, this cannot happen.

Aside from ending the quest for exact efficient algorithms, an NP-hardness result also serves as an insightful bridge between different areas of mathematics. Indeed, when we give a reduction from an NP-hard problem to a new problem of possibly different nature, it becomes apparent that the computational difficulties associated with the first problem are intrinsic also to the new problem. Conversely, any algorithm that has been previously developed for the new problem can now readily be applied also to the first problem. This concept is usually particularly interesting when one problem is in the domain of discrete mathematics and the other in the continuous domain, as will be the case for problems considered in this thesis. For example, we will give a reduction from the canonical NP-complete problem of 3SAT to the problem of deciding stability of a certain class of differential equations. As a byproduct of the reduction, it will follow that a certificate of unsatisfiability of instances of 3SAT can always be given in form of a Lyapunov function.

In general, hardness results in optimization come with a clear practical implication: as an algorithm designer, we either have to give up optimality and be content with finding suboptimal solutions, or we have to work with a subclass of problems that have more tractable attributes. In view of this, it becomes exceedingly relevant to identify structural properties of optimization problems that allow for tractability of finding optimal solutions.

One such structural property, which by and large is the most fundamental one that we know of, is *convexity*. As a geometric property, convexity comes with many attractive consequences. For instance, every local minimum of a convex problem is also a global minimum. Or for example, if a point does not belong to a convex set, this nonmembership can be certified through a separating hyperplane. Due in part to such special attributes, convex problems generally allow for efficient algorithms for solving them. Among other approaches, a powerful theory of interior-point polynomial time methods for convex optimization was developed in [111]. At least when the underlying convex cone has an efficiently computable so-called "barrier function", these algorithms are efficient both in theory and in practice.

Extensive and greatly successful research in the applications of convex optimization over the last couple of decades has shown that surprisingly many problems of practical importance can be cast as convex optimization problems. Moreover, we have a fair number of rules based on the calculus of convex functions that allow us to design—whenever we have the freedom to do so—problems that are by construction convex. Nevertheless, in order to be able to exploit the potential of convexity in optimization in full, a very basic question is to understand whether we are even able to *recognize* the presence of convexity in optimization problems. In other words, can we have an efficient algorithm that tests whether a given optimization problem is convex?

We will show in this thesis—answering a longstanding question of N.Z. Shor—that unfortunately even for the simplest classes of optimization problems where the objective function and the defining functions of the feasible set are given by polynomials of modest degree, the question of determining convexity is NP-hard. We also show that the same intractability result holds for essentially any well-known variant of convexity (generalized convexity). These results suggest that as significant as convexity may be in optimization, we may not be able to in general guarantee its presence before we can enjoy its consequences.

Of course, NP-hardness of a problem does not stop us from studying it, but on the contrary stresses the need for finding good approximation algorithms that can deal with a large number of instances efficiently. Towards this goal, we will devote part of this thesis to a study of convexity from an *algebraic* viewpoint. We will argue that in many cases, a notion known as *sos-convexity*, which is an efficiently checkable algebraic counterpart of convexity, can be a viable substitute for convexity of polynomials. Aside from its computational implications, sos-convexity has recently received much attention in the area of convex algebraic geometry [26],[55],[75],[89],[90],[91], mainly due to its role in connecting the geometric and algebraic aspects of convexity. In particular, the name "sos-convexity" comes from the work of Helton and Nie on semidefinite representability of convex sets [75].

The basic idea behind sos-convexity is nothing more than a simple extension of the concept of representation of nonnegative polynomials as sums of squares. To demonstrate this idea on a concrete example, suppose we are given the polynomial

$$\begin{aligned} p(x) \;=\; & x_1^4 - 6x_1^3x_2 + 2x_1^3x_3 + 6x_1^2x_3^2 + 9x_1^2x_2^2 - 6x_1^2x_2x_3 - 14x_1x_2x_3^2 + 4x_1x_3^3 \\ & +5x_3^4 - 7x_2^2x_3^2 + 16x_2^4, \end{aligned} \tag{1.1}$$

and we are asked to decide whether it is nonnegative, i.e, whether $p(x) \geq 0$ for all $x := (x_1, x_2, x_3)$ in $\mathbb{R}^3$. This may seem like a daunting task (and indeed it is as deciding nonnegativity of quartic polynomials is also NP-hard), but suppose that we could "somehow" come up with a decomposition of the polynomial a sum of squares (sos):

$$p(x) = (x_1^2 - 3x_1x_2 + x_1x_3 + 2x_3^2)^2 + (x_1x_3 - x_2x_3)^2 + (4x_2^2 - x_3^2)^2. \tag{1.2}$$

Then, we have at our hands an *explicit certificate* of nonnegativity of $p(x)$, which can be easily checked (simply by multiplying the terms out).

It turns out (see e.g. [118], [119]) that because of several interesting connections between real algebra and convex optimization discovered in recent years and quite well-known by now, the question of existence of an sos decomposition can be cast as a semidefinite program, which can be solved efficiently e.g. by interior point methods. As we will see more formally later, the notion of sos-convexity is based on an appropriately defined sum of squares decomposition of the Hessian matrix of a polynomial and hence it can also be checked efficiently with semidefinite programming. Just like sum of squares decomposition is a sufficient condition for polynomial nonnegativity, sos-convexity is a sufficient condition for polynomial convexity.

An important question that remains here is the obvious one: when do nonnegative polynomials admit a decomposition as a sum of squares? The answer to this question comes from a classical result of Hilbert. In his seminal 1888 paper [77], Hilbert gave a complete characterization of the degrees and dimensions in which all nonnegative polynomials can be written as sums of squares. In particular, he proved that there exist nonnegative polynomials with no sum of squares decomposition, although explicit examples of such polynomials appeared only 80 years later. One of the main contributions of this thesis is to establish the counterpart of Hilbert's results for the notions of convexity and sos-convexity. In particular, we will give the first example of a convex polynomial that is not sos-convex, and by the end of the first half of this thesis, a complete characterization of the degrees and dimensions in which convexity and sos-convexity are equivalent. Some interesting and unexpected connections to Hilbert's results will also emerge in the process.

In the second half of this thesis, we will turn to the study of stability in dynamical systems. Here too, we will take a computational viewpoint with our goal will being the development and analysis of efficient algorithms for proving stability of certain classes of nonlinear and hybrid systems.

Almost universally, the study of stability in systems theory leads to Lyapunov's second method or one of its many variants. An outgrowth of Lyapunov's 1892 doctoral dissertation [99], Lyapunov's second method tells us, roughly speaking, that if we succeed in finding a *Lyapunov function*—an energy-like function of the state that decreases along trajectories—then we have proven that the dynamical system in question is stable. In the mid 1900s, a series of *converse Lyapunov theorems* were developed which established that any stable system indeed has a Lyapunov function (see [72, Chap. 6] for an overview). Although this is encouraging, except for the simplest classes of systems such as linear systems, converse Lyapunov theorems do not provide much practical insight into how one may go about finding a Lyapunov function.

In the last few decades however, advances in the theory and practice of convex optimization and in particular semidefinite programming (SDP) have rejuvenated Lyapunov theory. The approach has been to parameterize a class of Lyapunov functions with restricted complexity (e.g., quadratics, pointwise maximum of quadratics, polynomials, etc.) and then pose the search for a Lyapunov function as a convex feasibility problem. A widely popular example of this framework which we will revisit later in this thesis is the method of sum of squares Lyapunov functions [118],[121]. Expanding on the concept of sum of squares decomposition of polynomials described above, this technique allows one to formulate semidefinite programs that search for polynomial Lyapunov functions for polynomial dynamical systems. Sum of squares Lyapunov functions, along with many other SDP based techniques, have also been applied to systems that undergo switching; see e.g. [136],[131],[122]. The analysis of these types of systems will also be a subject of interest in this thesis.

An algorithmic approach to Lyapunov theory naturally calls for new converse theorems. Indeed, classical converse Lyapunov theorems only guarantee existence of Lyapunov functions within very broad classes of functions (e.g. the class of continuously differentiable functions) that are a priori not amenable to computation. So there is the need to know whether Lyapunov functions belonging to certain more restricted classes of functions that can be computationally searched over also exist. For example, do stable polynomial systems admit Lyapunov functions that are polynomial? What about polynomial functions that can be found with sum of squares techniques? Similar questions arise in the case of switched systems. For example, do stable linear switched systems admit sum of squares Lyapunov functions? How about Lyapunov functions that are the pointwise maximum of quadratics? If so, how many quadratic functions are needed? We will answer several questions of this type in this thesis.

This thesis will also introduce a new class of techniques for Lyapunov analysis of switched systems. The novel component here is a general framework for formulating Lyapunov inequalities between *multiple* Lyapunov functions that together guarantee stability of a switched system under arbitrary switching. The relation between these inequalities has interesting links to concepts from automata theory. Furthermore, the technique is amenable to semidefinite programming.

Although the main ideas behind our approach directly apply to broader classes of switched systems, our results will be presented in the more specific context of switched linear systems. This is mainly due to our interest in the notion of the *joint spectral radius* of a set of matrices which has intimate connections to stability of switched linear systems. The joint spectral radius is an extensively studied quantity that characterizes the maximum growth rate obtained by taking arbitrary products from a set of matrices. Computation of the joint spectral radius, although notoriously hard [35],[161], has a wide range of applications including continuity of wavelet functions, computation of capacity of codes, convergence of consensus algorithms, and combinatorics, just to name a few. Our techniques provide several hierarchies of polynomial time algorithms that approximate the JSR with guaranteed accuracy.

A more concrete account of the contributions of this thesis will be given in the following section. We remark that although the first half of the thesis is mostly concerned with convexity in polynomial optimization and the second half with Lyapunov analysis, a common theme throughout the thesis is the use of algorithms that involve algebraic methods in optimization and semidefinite programming.

1.1 Outline and contributions of the thesis

The remainder of this thesis is divided into two parts each containing two chapters. The first part includes our complexity results on deciding convexity in polynomial optimization (Chapter 2) and our study of the relationship between convexity and sos-convexity (Chapter 3). The second part includes new results on Lyapunov analysis of polynomial differential equations (Chapter 4) and a novel framework for proving stability of switched systems (Chapter 5). A summary of our contributions in each chapter is as follows.

Chapter 2. The main result of this chapter is to prove that unless P=NP, there cannot be a polynomial time algorithm (or even a pseudo-polynomial time algorithm) that can decide whether a quartic polynomial is globally convex. This answers a question of N.Z. Shor that appeared as one of seven open problems in complexity theory for numerical optimization in 1992 [117]. We also show that deciding strict convexity, strong convexity, quasiconvexity, and pseudoconvexity of polynomials of even degree four or higher is strongly NP-hard. By contrast, we show that quasiconvexity and pseudoconvexity of odd degree polynomials can be decided in polynomial time.

Chapter 3. Our first contribution in this chapter is to prove that three natural sum of squares (sos) based sufficient conditions for convexity of polynomials via the definition of convexity, its first order characterization, and its second order characterization are equivalent. These three equivalent algebraic conditions, which we will refer to as sos-convexity, can be checked by solving a single semidefinite program. We present the first known example of a convex polynomial that is not sos-convex. We explain how this polynomial was found with tools from sos programming and duality theory of semidefinite optimization. As a byproduct of this numerical procedure, we obtain a simple method for searching over a restricted family of nonnegative polynomials that are not sums of squares that can be of independent interest.

If we denote the set of convex and sos-convex polynomials in n variables of degree d with $\tilde{C}_{n,d}$ and $\tilde{\Sigma C}_{n,d}$ respectively, then our main contribution in this chapter is to prove that $\tilde{C}_{n,d} = \tilde{\Sigma C}_{n,d}$ if and only if $n = 1$ or $d = 2$ or $(n, d) = (2, 4)$. We also present a complete characterization for forms (homogeneous polynomials) except for the case $(n, d) = (3, 4)$ which will appear elsewhere [2]. Our result states that the set $C_{n,d}$ of convex forms in n variables of degree d equals the set $\Sigma C_{n,d}$ of sos-convex forms if and only if $n = 2$ or $d = 2$ or $(n, d) = (3, 4)$. To prove these results, we present in particular explicit examples of polynomials in $\tilde{C}_{2,6} \setminus \tilde{\Sigma C}_{2,6}$ and $\tilde{C}_{3,4} \setminus \tilde{\Sigma C}_{3,4}$ and forms in $C_{3,6} \setminus \Sigma C_{3,6}$ and $C_{4,4} \setminus \Sigma C_{4,4}$, and a general procedure for constructing forms in $C_{n,d+2} \setminus \Sigma C_{n,d+2}$ from nonnegative but not sos forms in n variables and degree d.

Although for disparate reasons, the remarkable outcome is that convex polynomials (resp. forms) are sos-convex exactly in cases where nonnegative polynomials (resp. forms) are sums of squares, as characterized by Hilbert.

Chapter 4. This chapter is devoted to converse results on (non)-existence of polynomial and sum of squares polynomial Lyapunov functions for systems described by polynomial differential equations. We present a simple, explicit example of a two-dimensional polynomial vector field of degree two that is globally asymptotically stable but does not admit a polynomial Lyapunov function of any degree. We then study whether existence of a polynomial Lyapunov function implies existence of one that can be found with sum of squares techniques. We show via an explicit counterexample that if the degree of the polynomial Lyapunov function is fixed, then sos programming can fail to find a valid Lyapunov function even though one exists. On the other hand, if the degree is allowed to increase, we prove that existence of a polynomial Lyapunov function for a planar vector field (under an additional mild assumption) or for a homogeneous vector field implies existence of a polynomial Lyapunov function that is sos and that the negative of its derivative is also sos. This result is extended to prove that asymptotic stability of switched linear systems can always be proven with sum of squares Lyapunov functions. Finally, we show that for the latter class of systems (both in discrete and continuous time), if the negative of the derivative of a Lyapunov function is a sum of squares, then the Lyapunov function itself is automatically a sum of squares.

This chapter also includes some complexity results. We prove that deciding asymptotic stability of homogeneous cubic polynomial vector fields is strongly NP-hard. We discuss some byproducts of the reduction that establishes this result, including a Lyapunov-inspired technique for proving positivity of forms.

Chapter 5. In this chapter, we introduce the framework of path-complete graph Lyapunov functions for approximation of the joint spectral radius. The approach is based on the analysis of the underlying switched system via inequalities imposed between multiple Lyapunov functions associated to a labeled directed graph. The nodes of this graph represent Lyapunov functions, and its directed edges that are labeled with matrices represent Lyapunov inequalities. Inspired by concepts in automata theory and symbolic dynamics, we define a class of graphs called path-complete graphs, and show that any such graph gives rise to a method for proving stability of the switched system. This enables us to derive several asymptotically tight hierarchies of semidefinite programming relaxations that unify and generalize many existing techniques such as common quadratic, common sum of squares, and maximum/minimum-of-quadratics Lyapunov functions.

We compare the quality of approximation obtained by certain families of path-complete graphs including all path-complete graphs with two nodes on an alphabet of two matrices. We argue that the De Bruijn graph of order one on m symbols, with quadratic Lyapunov functions assigned to its nodes, provides good estimates of the JSR of m matrices at a modest computational cost. We prove that the bound obtained via this method is invariant under transposition of the matrices and always within a multiplicative factor of $1/\sqrt[4]{n}$ of the true JSR (independent of the number of matrices).

Approximation guarantees for analysis via other families of path-complete graphs will also be provided. In particular, we show that the De Bruijn graph of order k, with quadratic Lyapunov functions as nodes, can approximate the JSR with arbitrary accuracy as k increases. This also proves that common Lyapunov functions that are the pointwise maximum (or minimum) of quadratics always exist. Moreover, the result gives a bound on the number of quadratic functions needed to achieve a desired level of accuracy in approximation of the JSR, and also demonstrates that these quadratic functions can be found with semidefinite programming.

A list of open problems for future research is presented at the end of each chapter.

■ 1.1.1 Related publications

The material presented in this thesis is in the most part based on the following papers.

Chapter 2.
A. A. Ahmadi, A. Olshevsky, P. A. Parrilo, and J. N. Tsitsiklis. NP-hardness of deciding convexity of quartic polynomials and related problems. *Mathematical Programming*, 2011. Accepted for publication. Online version available at arXiv:.1012.1908.

Chapter 3.
A. A. Ahmadi and P. A. Parrilo. A convex polynomial that is not sos-convex. *Mathematical Programming*, 2011. DOI: 10.1007/s10107-011-0457-z.

A. A. Ahmadi and P. A. Parrilo. A complete characterization of the gap between convexity and sos-convexity. In preparation, 2011.

A. A. Ahmadi, G. Blekherman, and P. A.Parrilo. Convex ternary quartics are sos-convex. In preparation, 2011.

Chapter 4.
A. A. Ahmadi and P. A. Parrilo. Converse results on existence of sum of squares Lyapunov functions. In *Proceedings of the 50th IEEE Conference on Decision and Control*, 2011.

A. A. Ahmadi, M. Krstic, and P. A. Parrilo. A globally asymptotically stable polynomial vector field with no polynomial Lyapunov function. In *Proceedings of the 50th IEEE Conference on Decision and Control*, 2011.

Chapter 5.
A. A. Ahmadi, R. Jungers, P. A. Parrilo, and M. Roozbehani. Analysis of the joint spectral radius via Lyapunov functions on path-complete graphs. In *Hybrid Systems: Computation and Control 2011*, Lecture Notes in Computer Science. Springer, 2011.

Part I:
Computational and Algebraic Aspects of Convexity

Chapter 2

Complexity of Deciding Convexity

In this chapter, we characterize the computational complexity of deciding convexity and many of its variants in polynomial optimization. The material presented in this chapter is based on the work in [5].

2.1 Introduction

The role of *convexity* in modern day mathematical programming has proven to be remarkably fundamental, to the point that tractability of an optimization problem is nowadays assessed, more often than not, by whether or not the problem benefits from some sort of underlying convexity. In the famous words of Rockafellar [143]:

> "In fact the great watershed in optimization isn't between linearity and nonlinearity, but convexity and nonconvexity."

But how easy is it to distinguish between convexity and nonconvexity? Can we decide in an efficient manner if a given optimization problem is convex?

A class of optimization problems that allow for a rigorous study of this question from a computational complexity viewpoint is the class of polynomial optimization problems. These are optimization problems where the objective is given by a polynomial function and the feasible set is described by polynomial inequalities. Our research in this direction was motivated by a concrete question of N. Z. Shor that appeared as one of seven open problems in complexity theory for numerical optimization put together by Pardalos and Vavasis in 1992 [117]:

> "Given a degree-4 polynomial in n variables, what is the complexity of determining whether this polynomial describes a convex function?"

As we will explain in more detail shortly, the reason why Shor's question is specifically about degree 4 polynomials is that deciding convexity of odd degree polynomials is trivial and deciding convexity of degree 2 (quadratic) polynomials can be reduced to the simple task of checking whether a constant matrix is positive semidefinite. So, the first interesting case really occurs for degree 4 (quartic) polynomials. Our main contribution in this chapter (Theorem 2.1 in Section 2.2.3) is to show that deciding convexity of polynomials is strongly NP-hard already for polynomials of degree 4.

The implication of NP-hardness of this problem is that unless P=NP, there exists no algorithm that can take as input the (rational) coefficients of a quartic polynomial, have running time bounded by a polynomial in the number of bits needed to represent the coefficients, and output correctly on every instance whether or not the polynomial is convex. Furthermore, the fact that our NP-hardness result is in the strong sense (as opposed to weakly NP-hard problems such as KNAPSACK) implies, roughly speaking, that the problem remains NP-hard even when the magnitude of the coefficients of the polynomial are restricted to be "small." For a strongly NP-hard problem, even a pseudo-polynomial time algorithm cannot exist unless P=NP. See [61] for precise definitions and more details.

There are many areas of application where one would like to establish convexity of polynomials. Perhaps the simplest example is in global minimization of polynomials, where it could be very useful to decide first whether the polynomial to be optimized is convex. Once convexity is verified, then every local minimum is global and very basic techniques (e.g., gradient descent) can find a global minimum—a task that is in general NP-hard in the absence of convexity [124], [109]. As another example, if we can certify that a homogeneous polynomial is convex, then we define a gauge (or Minkowski) norm based on its convex sublevel sets, which may be useful in many applications. In several other problems of practical relevance, we might not just be interested in checking whether a given polynomial is convex, but to *parameterize* a family of convex polynomials and perhaps search or optimize over them. For example we might be interested in approximating the convex envelope of a complicated nonconvex function with a convex polynomial, or in fitting a convex polynomial to a set of data points with minimum error [100]. Not surprisingly, if testing membership to the set of convex polynomials is hard, searching and optimizing over that set also turns out to be a hard problem.

We also extend our hardness result to some variants of convexity, namely, the problems of deciding *strict convexity*, *strong convexity*, *pseudoconvexity*, and *quasiconvexity* of polynomials. Strict convexity is a property that is often useful to check because it guarantees uniqueness of the optimal solution in optimization problems. The notion of strong convexity is a common assumption in convergence analysis of many iterative Newton-type algorithms in optimization theory; see, e.g., [38, Chaps. 9–11]. So, in order to ensure the theoretical convergence rates promised by many of these algorithms, one needs to first make sure that the objective function is strongly convex. The problem of checking quasiconvexity (convexity of sublevel sets) of polynomials also arises frequently in practice. For instance, if the feasible set of an optimization problem is defined by polynomial inequalities, by certifying quasiconvexity of the defining polynomials we can ensure that the feasible set is convex. In several statistics and clustering problems, we are interested in finding minimum volume convex sets that contain a set of data points in space. This problem can be tackled by searching over the set of quasiconvex polynomials [100]. In economics, quasiconcave functions are prevalent as desirable utility functions [92], [18]. In control and systems theory, it is useful at times to search for quasiconvex Lyapunov functions whose convex sublevel sets contain relevant information about the trajectories of a dynamical system [44], [8]. Finally, the notion of pseudoconvexity is a natural generalization of convexity that inherits many of the attractive properties of convex functions. For example, every stationary point or every local minimum of a pseudoconvex function must be a global minimum. Because of these nice features, pseudoconvex programs have been studied extensively in nonlinear programming [101], [48].

As an outcome of close to a century of research in convex analysis, numerous necessary, sufficient, and exact conditions for convexity and all of its variants are available; see, e.g., [38, Chap. 3], [104], [60], [49], [92], [102] and references therein for a by no means exhaustive list. Our results suggest that none of the exact characterizations of these notions can be efficiently checked for polynomials. In fact, when turned upside down, many of these equivalent formulations reveal new NP-hard problems; see, e.g., Corollary 2.6 and 2.8.

2.1.1 Related Literature

There are several results in the literature on the complexity of various special cases of polynomial optimization problems. The interested reader can find many of these results in the edited volume of Pardalos [116] or in the survey papers of de Klerk [54], and Blondel and Tsitsiklis [36]. A very general and fundamental concept in certifying feasibility of polynomial equations and inequalities is the Tarski–Seidenberg quantifier elimination theory [158], [154], from which it follows that all of the problems that we consider in this chapter are algorithmically *decidable*. This means that there are algorithms that on all instances of our problems of interest halt in finite time and always output the correct yes–no answer. Unfortunately, algorithms based on quantifier elimination or similar decision algebra techniques have running times that are at least exponential in the number of variables [24], and in practice can only solve problems with very few parameters.

When we turn to the issue of polynomial time solvability, perhaps the most relevant result for our purposes is the NP-hardness of deciding nonnegativity of quartic polynomials and biquadratic forms (see Definition 2.2); the main reduction that we give in this chapter will in fact be from the latter problem. As we will see in Section 2.2.3, it turns out that deciding convexity of quartic forms is equivalent to checking nonnegativity of a special class of biquadratic forms, which are themselves a special class of quartic forms. The NP-hardness of checking nonnegativity of quartic forms follows, e.g., as a direct consequence of NP-hardness of testing matrix copositivity, a result proven by Murty and Kabadi [109]. As for the hardness of checking nonnegativity of biquadratic forms, we know of two different proofs. The first one is due to Gurvits [70], who proves that the entanglement problem in quantum mechanics (i.e., the problem of distinguishing separable quantum states from entangled ones) is NP-hard. A dual reformulation of this result shows directly that checking nonnegativity of biquadratic forms is NP-hard; see [59]. The second proof is due to Ling et al. [97], who use a theorem of Motzkin and Straus to give a very short and elegant reduction from the maximum clique problem in graphs.

The only work in the literature on the hardness of deciding polynomial convexity that we are aware of is the work of Guo on the complexity of deciding convexity of quartic polynomials over simplices [69]. Guo discusses some of the difficulties that arise from this problem, but he does not prove that deciding convexity of polynomials over simplices is NP-hard. Canny shows in [40] that the existential theory of the real numbers can be decided in PSPACE. From this, it follows that testing several properties of polynomials, including nonnegativity and convexity, can be done in polynomial space. In [112], Nie proves that the related notion of *matrix convexity* is NP-hard for polynomial matrices whose entries are quadratic forms.

On the algorithmic side, several techniques have been proposed both for testing convexity of sets and convexity of functions. Rademacher and Vempala present and analyze randomized algorithms for testing the relaxed notion of *approximate convexity* [135]. In [91], Lasserre proposes a semidefinite programming hierarchy for testing convexity of basic closed semialgebraic sets; a problem that we also prove to be NP-hard (see Corollary 2.8). As for testing convexity of functions, an approach that some convex optimization parsers (e.g., `CVX` [66]) take is to start with some ground set of convex functions and then check whether the desired function can be obtained by applying a set of convexity preserving operations to the functions in the ground set [50], [38, p. 79]. Techniques of this type that are based on the calculus of convex functions are successful for a large range of applications. However, when applied to general polynomial functions, they can only detect a subclass of convex polynomials.

Related to convexity of polynomials, a concept that has attracted recent attention is the algebraic notion of *sos-convexity* (see Definition 2.4) [75], [89], [90],

[8], [100], [44], [11]. This is a powerful sufficient condition for convexity that relies on an appropriately defined sum of squares decomposition of the Hessian matrix, and can be efficiently checked by solving a single semidefinite program. The study of sos-convexity will be the main focus of our next chapter. In particular, we will present explicit counterexamples to show that not every convex polynomial is sos-convex. The NP-hardness result in this chapter certainly justifies the existence of such counterexamples and more generally suggests that *any* polynomial time algorithm attempted for checking polynomial convexity is doomed to fail on some hard instances.

2.1.2 Contributions and organization of this chapter

The main contribution of this chapter is to establish the computational complexity of deciding convexity, strict convexity, strong convexity, pseudoconvexity, and quasiconvexity of polynomials for any given degree. (See Table 2.1 in Section 2.5 for a quick summary.) The results are mainly divided in three sections, with Section 2.2 covering convexity, Section 2.3 covering strict and strong convexity, and Section 2.4 covering quasiconvexity and pseudoconvexity. These three sections follow a similar pattern and are each divided into three parts: first, the definitions and basics, second, the degrees for which the questions can be answered in polynomial time, and third, the degrees for which the questions are NP-hard.

Our main reduction, which establishes NP-hardness of checking convexity of quartic forms, is given in Section 2.2.3. This hardness result is extended to strict and strong convexity in Section 2.3.3, and to quasiconvexity and pseudoconvexity in Section 2.4.3. By contrast, we show in Section 2.4.2 that quasiconvexity and pseudoconvexity of odd degree polynomials can be decided in polynomial time. A summary of the chapter and some concluding remarks are presented in Section 2.5.

2.2 Complexity of deciding convexity

2.2.1 Definitions and basics

A (multivariate) *polynomial* $p(x)$ in variables $x := (x_1, \ldots, x_n)^T$ is a function from $\mathbb{R}^n$ to $\mathbb{R}$ that is a finite linear combination of monomials:

$$p(x) = \sum_{\alpha} c_\alpha x^\alpha = \sum_{\alpha_1,\ldots,\alpha_n} c_{\alpha_1,\ldots,\alpha_n} x_1^{\alpha_1} \cdots x_n^{\alpha_n}, \tag{2.1}$$

where the sum is over n-tuples of nonnegative integers α_i. An algorithm for testing some property of polynomials will have as its input an ordered list of the coefficients c_α. Since our complexity results are based on models of digital computation, where the input must be represented by a finite number of bits, the coefficients c_α for us will always be rational numbers, which upon clearing the denominators can be taken to be integers. So, for the remainder of the chapter, even when not explicitly stated, we will always have $c_\alpha \in \mathbb{Z}$.

The *degree* of a monomial x^α is equal to $\alpha_1 + \cdots + \alpha_n$. The degree of a polynomial $p(x)$ is defined to be the highest degree of its component monomials. A simple counting argument shows that a polynomial of degree d in n variables has $\binom{n+d}{d}$ coefficients. A *homogeneous polynomial* (or a *form*) is a polynomial where all the monomials have the same degree. A form $p(x)$ of degree d is a homogeneous function of degree d (since it satisfies $p(\lambda x) = \lambda^d p(x)$), and has $\binom{n+d-1}{d}$ coefficients.

A polynomial $p(x)$ is said to be *nonnegative* or *positive semidefinite (psd)* if $p(x) \geq 0$ for all $x \in \mathbb{R}^n$. Clearly, a necessary condition for a polynomial to be psd is for its degree to be even. We say that $p(x)$ is a *sum of squares (sos)*, if there exist polynomials $q_1(x), \ldots, q_m(x)$ such that $p(x) = \sum_{i=1}^m q_i^2(x)$. Every sos polynomial is obviously psd. A *polynomial matrix* $P(x)$ is a matrix with polynomial entries. We say that a polynomial matrix $P(x)$ is *PSD* (denoted $P(x) \succeq 0$) if it is positive semidefinite in the matrix sense for every value of the indeterminates x. (Note the upper case convention for matrices.) It is easy to see that $P(x)$ is PSD if and only if the scalar polynomial $y^T P(x) y$ in variables $(x; y)$ is psd.

We recall that a polynomial $p(x)$ is convex if and only if its Hessian matrix, which will be generally denoted by $H(x)$, is PSD.

2.2.2 Degrees that are easy

The question of deciding convexity is trivial for odd degree polynomials. Indeed, it is easy to check that linear polynomials ($d = 1$) are always convex and that polynomials of odd degree $d \geq 3$ can never be convex. The case of quadratic polynomials ($d = 2$) is also straightforward. A quadratic polynomial $p(x) = \frac{1}{2}x^T Q x + q^T x + c$ is convex if and only if the constant matrix Q is positive semidefinite. This can be decided in polynomial time for example by performing Gaussian pivot steps along the main diagonal of Q [109] or by computing the characteristic polynomial of Q exactly and then checking that the signs of its coefficients alternate [79, p. 403].

Unfortunately, the results that come next suggest that the case of quadratic polynomials is essentially the only nontrivial case where convexity can be efficiently decided.

2.2.3 Degrees that are hard

The main hardness result of this chapter is the following theorem.

Theorem 2.1. *Deciding convexity of degree four polynomials is strongly NP-hard. This is true even when the polynomials are restricted to be homogeneous.*

We will give a reduction from the problem of deciding nonnegativity of biquadratic forms. We start by recalling some basic facts about biquadratic forms and sketching the idea of the proof.

Definition 2.2. *A* biquadratic form $b(x;y)$ *is a form in the variables* $x = (x_1, \ldots, x_n)^T$ *and* $y = (y_1, \ldots, y_m)^T$ *that can be written as*

$$b(x;y) = \sum_{i \leq j,\, k \leq l} \alpha_{ijkl} x_i x_j y_k y_l. \tag{2.2}$$

Note that for fixed x, $b(x;y)$ becomes a quadratic form in y, and for fixed y, it becomes a quadratic form in x. Every biquadratic form is a quartic form, but the converse is of course not true. It follows from a result of Ling et al. [97] that deciding nonnegativity of biquadratic forms is strongly NP-hard. This claim is not precisely stated in this form in [97]. For the convenience of the reader, let us make the connection more explicit before we proceed, as this result underlies everything that follows.

The argument in [97] is based on a reduction from CLIQUE (given a graph $G(V,E)$ and a positive integer $k \leq |V|$, decide whether G contains a clique of size k or more) whose (strong) NP-hardness is well-known [61]. For a given graph $G(V,E)$ on n nodes, if we define the biquadratic form $b_G(x;y)$ in the variables $x = (x_1, \ldots, x_n)^T$ and $y = (y_1, \ldots, y_n)^T$ by

$$b_G(x;y) = -2 \sum_{(i,j) \in E} x_i x_j y_i y_j,$$

then Ling et al. [97] use a theorem of Motzkin and Straus [108] to show

$$\min_{||x||=||y||=1} b_G(x;y) = -1 + \frac{1}{\omega(G)}. \tag{2.3}$$

Here, $\omega(G)$ denotes the clique number of the graph G, i.e., the size of a maximal clique.[1] From this, we see that for any value of k, $\omega(G) \leq k$ if and only if

$$\min_{||x||=||y||=1} b_G(x;y) \geq \frac{1-k}{k},$$

[1] Equation (2.3) above is stated in [97] with the stability number $\alpha(G)$ in place of the clique number $\omega(G)$. This seems to be a minor typo.

which by homogenization holds if and only if the biquadratic form

$$\hat{b}_G(x;y) = -2k \sum_{(i,j)\in E} x_i x_j y_i y_j - (1-k)\left(\sum_{i=1}^n x_i^2\right)\left(\sum_{i=1}^n y_i^2\right)$$

is nonnegative. Hence, by checking nonnegativity of $\hat{b}_G(x;y)$ for all values of $k \in \{1,\ldots,n-1\}$, we can find the exact value of $\omega(G)$. It follows that deciding nonnegativity of biquadratic forms is NP-hard, and in view of the fact that the coefficients of $\hat{b}_G(x;y)$ are all integers with absolute value at most $2n-2$, the NP-hardness claim is in the strong sense. Note also that the result holds even when $n = m$ in Definition 2.2. In the sequel, we will always have $n = m$.

It is not difficult to see that any biquadratic form $b(x;y)$ can be written in the form

$$b(x;y) = y^T A(x) y \tag{2.4}$$

(or of course as $x^T B(y) x$) for some symmetric polynomial matrix $A(x)$ whose entries are quadratic forms. Therefore, it is strongly NP-hard to decide whether a symmetric polynomial matrix with quadratic form entries is PSD. One might hope that this would lead to a quick proof of NP-hardness of testing convexity of quartic forms, because the Hessian of a quartic form is exactly a symmetric polynomial matrix with quadratic form entries. However, the major problem that stands in the way is that not every polynomial matrix is a *valid Hessian*. Indeed, if any of the partial derivatives between the entries of $A(x)$ do not commute (e.g., if $\frac{\partial A_{11}(x)}{\partial x_2} \neq \frac{\partial A_{12}(x)}{\partial x_1}$), then $A(x)$ cannot be the matrix of second derivatives of some polynomial. This is because all mixed third partial derivatives of polynomials must commute.

Our task is therefore to prove that even with these additional constraints on the entries of $A(x)$, the problem of deciding positive semidefiniteness of such matrices remains NP-hard. We will show that any given symmetric $n \times n$ matrix $A(x)$, whose entries are quadratic forms, can be embedded in a $2n \times 2n$ polynomial matrix $H(x,y)$, again with quadratic form entries, so that $H(x,y)$ is a valid Hessian and $A(x)$ is PSD if and only if $H(x,y)$ is. In fact, we will directly construct the polynomial $f(x,y)$ whose Hessian is the matrix $H(x,y)$. This is done in the next theorem, which establishes the correctness of our main reduction. Once this theorem is proven, the proof of Theorem 2.1 will become immediate.

Theorem 2.3. *Given a biquadratic form $b(x;y)$, define the the $n \times n$ polynomial matrix $C(x,y)$ by setting*

$$[C(x,y)]_{ij} := \frac{\partial b(x;y)}{\partial x_i \partial y_j}, \tag{2.5}$$

and let γ be the largest coefficient, in absolute value, of any monomial present in some entry of the matrix $C(x,y)$. Let f be the form given by

$$f(x,y) := b(x;y) + \frac{n^2\gamma}{2}\Big(\sum_{i=1}^{n} x_i^4 + \sum_{i=1}^{n} y_i^4 + \sum_{\substack{i,j=1,\dots,n\\ i<j}} x_i^2 x_j^2 + \sum_{\substack{i,j=1,\dots,n\\ i<j}} y_i^2 y_j^2\Big). \quad (2.6)$$

Then, $b(x;y)$ is psd if and only if $f(x,y)$ is convex.

Proof. Before we prove the claim, let us make a few observations and try to shed light on the intuition behind this construction. We will use $H(x,y)$ to denote the Hessian of f. This is a $2n \times 2n$ polynomial matrix whose entries are quadratic forms. The polynomial f is convex if and only if $z^T H(x,y) z$ is psd. For bookkeeping purposes, let us split the variables z as $z := (z_x, z_y)^T$, where z_x and z_y each belong to $\mathbb{R}^n$. It will also be helpful to give a name to the second group of terms in the definition of $f(x,y)$ in (2.6). So, let

$$g(x,y) := \frac{n^2\gamma}{2}\Big(\sum_{i=1}^{n} x_i^4 + \sum_{i=1}^{n} y_i^4 + \sum_{\substack{i,j=1,\dots,n\\ i<j}} x_i^2 x_j^2 + \sum_{\substack{i,j=1,\dots,n\\ i<j}} y_i^2 y_j^2\Big). \quad (2.7)$$

We denote the Hessian matrices of $b(x,y)$ and $g(x,y)$ with $H_b(x,y)$ and $H_g(x,y)$ respectively. Thus, $H(x,y) = H_b(x,y) + H_g(x,y)$. Let us first focus on the structure of $H_b(x,y)$. Observe that if we define

$$[A(x)]_{ij} = \frac{\partial b(x;y)}{\partial y_i \partial y_j},$$

then $A(x)$ depends only on x, and

$$\frac{1}{2} y^T A(x) y = b(x;y). \quad (2.8)$$

Similarly, if we let

$$[B(y)]_{ij} = \frac{\partial b(x;y)}{\partial x_i \partial x_j},$$

then $B(y)$ depends only on y, and

$$\frac{1}{2} x^T B(y) x = b(x;y). \quad (2.9)$$

From Eq. (2.8), we have that $b(x;y)$ is psd if and only if $A(x)$ is PSD; from Eq. (2.9), we see that $b(x;y)$ is psd if and only if $B(y)$ is PSD.

Putting the blocks together, we have

$$H_b(x,y) = \begin{bmatrix} B(y) & C(x,y) \\ C^T(x,y) & A(x) \end{bmatrix}. \tag{2.10}$$

The matrix $C(x,y)$ is not in general symmetric. The entries of $C(x,y)$ consist of square-free monomials that are each a multiple of $x_i y_j$ for some i, j, with $1 \leq i,j \leq n$; (see (2.2) and (2.5)).

The Hessian $H_g(x,y)$ of the polynomial $g(x,y)$ in (2.7) is given by

$$H_g(x,y) = \frac{n^2\gamma}{2} \begin{bmatrix} H_g^{11}(x) & 0 \\ 0 & H_g^{22}(y) \end{bmatrix}, \tag{2.11}$$

where

$$H_g^{11}(x) = \begin{bmatrix} 12x_1^2 + 2\sum\limits_{\substack{i=1,\dots,n \\ i\neq 1}} x_i^2 & 4x_1x_2 & \cdots & 4x_1x_n \\ 4x_1x_2 & 12x_2^2 + 2\sum\limits_{\substack{i=1,\dots,n \\ i\neq 2}} x_i^2 & \cdots & 4x_2x_n \\ \vdots & \vdots & \ddots & \vdots \\ 4x_1x_n & \cdots & 4x_{n-1}x_n & 12x_n^2 + 2\sum\limits_{\substack{i=1,\dots,n \\ i\neq n}} x_i^2 \end{bmatrix}, \tag{2.12}$$

and

$$H_g^{22}(y) = \begin{bmatrix} 12y_1^2 + 2\sum\limits_{\substack{i=1,\dots,n \\ i\neq 1}} y_i^2 & 4y_1y_2 & \cdots & 4y_1y_n \\ 4y_1y_2 & 12y_2^2 + 2\sum\limits_{\substack{i=1,\dots,n \\ i\neq 2}} y_i^2 & \cdots & 4y_2y_n \\ \vdots & \vdots & \ddots & \vdots \\ 4y_1y_n & \cdots & 4y_{n-1}y_n & 12y_n^2 + 2\sum\limits_{\substack{i=1,\dots,n \\ i\neq n}} y_i^2 \end{bmatrix}. \tag{2.13}$$

Note that all diagonal elements of $H_g^{11}(x)$ and $H_g^{22}(y)$ contain the square of every variable $x_1, \dots, x_n$ and $y_1, \dots, y_n$ respectively.

We fist give an intuitive summary of the rest of the proof. If $b(x;y)$ is not psd, then $B(y)$ and $A(x)$ are not PSD and hence $H_b(x,y)$ is not PSD. Moreover, adding $H_g(x,y)$ to $H_b(x,y)$ cannot help make $H(x,y)$ PSD because the dependence of the diagonal blocks of $H_b(x,y)$ and $H_g(x,y)$ on x and y runs backwards. On the other hand, if $b(x;y)$ is psd, then $H_b(x,y)$ will have PSD diagonal blocks. In principle, $H_b(x,y)$ might still not be PSD because of the off-diagonal block $C(x,y)$. However, the squares in the diagonal elements of $H_g(x,y)$ will be shown to dominate the monomials of $C(x,y)$ and make $H(x,y)$ PSD.

Let us now prove the theorem formally. One direction is easy: if $b(x;y)$ is not psd, then $f(x,y)$ is not convex. Indeed, if there exist $\bar{x}$ and $\bar{y}$ in $\mathbb{R}^n$ such that $b(\bar{x};\bar{y}) < 0$, then

$$z^T H(x,y) z \Big|_{z_x=0, x=\bar{x}, y=0, z_y=\bar{y}} = \bar{y}^T A(\bar{x}) \bar{y} = 2b(\bar{x};\bar{y}) < 0.$$

For the converse, suppose that $b(x;y)$ is psd; we will prove that $z^T H(x,y) z$ is psd and hence $f(x,y)$ is convex. We have

$$\begin{aligned} z^T H(x,y) z &= z_y^T A(x) z_y + z_x^T B(y) z_x + 2 z_x^T C(x,y) z_y \\ &\quad + \tfrac{n^2\gamma}{2} z_x^T H_g^{11}(x) z_x + \tfrac{n^2\gamma}{2} z_y^T H_g^{22}(y) z_y. \end{aligned}$$

Because $z_y^T A(x) z_y$ and $z_x^T B(y) z_x$ are psd by assumption (see (2.8) and (2.9)), it suffices to show that $z^T H(x,y) z - z_y^T A(x) z_y - z_x^T B(y) z_x$ is psd. In fact, we will show that $z^T H(x,y) z - z_y^T A(x) z_y - z_x^T B(y) z_x$ is a sum of squares.

After some regrouping of terms we can write

$$z^T H(x,y) z - z_y^T A(x) z_y - z_x^T B(y) z_x = p_1(x,y,z) + p_2(x,z_x) + p_3(y,z_y), \quad (2.14)$$

where

$$p_1(x,y,z) = 2 z_x^T C(x,y) z_y + n^2\gamma \Big(\sum_{i=1}^n z_{x,i}^2 \Big) \Big(\sum_{i=1}^n x_i^2 \Big) + n^2\gamma \Big(\sum_{i=1}^n z_{y,i}^2 \Big) \Big(\sum_{i=1}^n y_i^2 \Big), \quad (2.15)$$

$$p_2(x,z_x) = n^2\gamma z_x^T \begin{bmatrix} 5x_1^2 & 2x_1x_2 & \cdots & 2x_1x_n \\ 2x_1x_2 & 5x_2^2 & \cdots & 2x_2x_n \\ \vdots & \vdots & \ddots & \vdots \\ 2x_1x_n & \cdots & 2x_{n-1}x_n & 5x_n^2 \end{bmatrix} z_x, \quad (2.16)$$

and

$$p_3(y,z_y) = n^2\gamma z_y^T \begin{bmatrix} 5y_1^2 & 2y_1y_2 & \cdots & 2y_1y_n \\ 2y_1y_2 & 5y_2^2 & \cdots & 2y_2y_n \\ \vdots & \vdots & \ddots & \vdots \\ 2y_1y_n & \cdots & 2y_{n-1}y_n & 5y_n^2 \end{bmatrix} z_y. \quad (2.17)$$

We show that (2.14) is sos by showing that p_1, p_2, and p_3 are each individually sos. To see that p_2 is sos, simply note that we can rewrite it as

$$p_2(x, z_x) = n^2\gamma \left[3 \sum_{k=1}^{n} z_{x,k}^2 x_k^2 + 2\Big(\sum_{k=1}^{n} z_{x,k} x_k\Big)^2\right].$$

The argument for p_3 is of course identical. To show that p_1 is sos, we argue as follows. If we multiply out the first term $2z_x^T C(x,y) z_y$, we obtain a polynomial with monomials of the form

$$\pm 2\beta_{i,j,k,l} z_{x,k} x_i y_j z_{y,l}, \tag{2.18}$$

where $0 \le \beta_{i,j,k,l} \le \gamma$, by the definition of γ. Since

$$\pm 2\beta_{i,j,k,l} z_{x,k} x_i y_j z_{y,l} + \beta_{i,j,k,l} z_{x,k}^2 x_i^2 + \beta_{i,j,k,l} y_j^2 z_{y,l}^2 = \beta_{i,j,k,l}(z_{x,k} x_i \pm y_j z_{y,l})^2, \tag{2.19}$$

by pairing up the terms of $2z_x^T C(x,y) z_y$ with fractions of the squared terms $z_{x,k}^2 x_i^2$ and $z_{y,l}^2 y_j^2$, we get a sum of squares. Observe that there are more than enough squares for each monomial of $2z_x^T C(x,y) z_y$ because each such monomial $\pm 2\beta_{i,j,k,l} z_{x,k} x_i y_j z_{y,l}$ occurs at most once, so that each of the terms $z_{x,k}^2 x_i^2$ and $z_{y,l}^2 y_j^2$ will be needed at most n^2 times, each time with a coefficient of at most γ. Therefore, p_1 is sos, and this completes the proof. □

We can now complete the proof of strong NP-hardness of deciding convexity of quartic forms.

Proof of Theorem 2.1. As we remarked earlier, deciding nonnegativity of biquadratic forms is known to be strongly NP-hard [97]. Given such a biquadratic form $b(x;y)$, we can construct the polynomial $f(x,y)$ as in (2.6). Note that $f(x,y)$ has degree four and is homogeneous. Moreover, the reduction from $b(x;y)$ to $f(x,y)$ runs in polynomial time as we are only adding to $b(x;y)$ $2n + 2\binom{n}{2}$ new monomials with coefficient $\frac{n^2\gamma}{2}$, and the size of γ is by definition only polynomially larger than the size of any coefficient of $b(x;y)$. Since by Theorem 2.3 convexity of $f(x,y)$ is equivalent to nonnegativity of $b(x;y)$, we conclude that deciding convexity of quartic forms is strongly NP-hard. □

An algebraic version of the reduction. Before we proceed further with our results, we make a slight detour and present an algebraic analogue of this reduction, which relates sum of squares biquadratic forms to sos-convex polynomials. Both of these concepts are well-studied in the literature, in particular in regards to their connection to semidefinite programming; see, e.g., [97], [11], and references therein.

Definition 2.4. *A polynomial $p(x)$, with its Hessian denoted by $H(x)$, is* sos-convex *if the polynomial $y^T H(x)y$ is a sum of squares in variables (x;y).*[2]

Theorem 2.5. *Given a biquadratic form $b(x;y)$, let $f(x,y)$ be the quartic form defined as in (2.6). Then $b(x;y)$ is a sum of squares if and only if $f(x,y)$ is sos-convex.*

Proof. The proof is very similar to the proof of Theorem 2.3 and is left to the reader. □

We will revisit Theorem 2.5 in the next chapter when we study the connection between convexity and sos-convexity.

Some NP-hardness results, obtained as corollaries. NP-hardness of checking convexity of quartic forms directly establishes NP-hardness[3] of several problems of interest. Here, we mention a few examples.

Corollary 2.6. *It is NP-hard to decide nonnegativity of a homogeneous polynomial q of degree four, of the form*

$$q(x,y) = \frac{1}{2}p(x) + \frac{1}{2}p(y) - p\left(\tfrac{x+y}{2}\right),$$

for some homogeneous quartic polynomial p.

Proof. Nonnegativity of q is equivalent to convexity of p, and the result follows directly from Theorem 2.1. □

Definition 2.7. *A set $\mathcal{S} \subset \mathbb{R}^n$ is* basic closed semialgebraic *if it can be written as*

$$\mathcal{S} = \{x \in \mathbb{R}^n |\ f_i(x) \geq 0,\ i = 1, \ldots, m\}, \tag{2.20}$$

for some positive integer m and some polynomials $f_i(x)$.

Corollary 2.8. *Given a basic closed semialgebraic set $\mathcal{S}$ as in (2.20), where at least one of the defining polynomials $f_i(x)$ has degree four, it is NP-hard to decide whether $\mathcal{S}$ is a convex set.*

Proof. Given a quartic polynomial $p(x)$, consider the basic closed semialgebraic set

$$\mathcal{E}_p = \{(x,t) \in \mathbb{R}^{n+1} |\ t - p(x) \geq 0\},$$

describing the epigraph of $p(x)$. Since $p(x)$ is convex if and only if its epigraph is a convex set, the result follows.[4] □

---

[2] Three other equivalent definitions of sos-convexity are presented in the next chapter.

[3] All of our NP-hardness results in this chapter are in the strong sense. For the sake of brevity, from now on we refer to strongly NP-hard problems simply as NP-hard problems.

[4] Another proof of this corollary is given by the NP-hardness of checking convexity of sublevel sets of quartic polynomials (Theorem 2.24 in Section 2.4.3).

Convexity of polynomials of even degree larger than four. We end this section by extending our hardness result to polynomials of higher degree.

Corollary 2.9. *It is NP-hard to check convexity of polynomials of any fixed even degree $d \geq 4$.*

Proof. We have already established the result for polynomials of degree four. Given such a degree four polynomial $p(x) := p(x_1, \ldots, x_n)$ and an even degree $d \geq 6$, consider the polynomial

$$q(x, x_{n+1}) = p(x) + x_{n+1}^d$$

in $n+1$ variables. It is clear (e.g., from the block diagonal structure of the Hessian of q) that $p(x)$ is convex if and only if $q(x)$ is convex. The result follows. □

2.3 Complexity of deciding strict convexity and strong convexity

2.3.1 Definitions and basics

Definition 2.10. *A function $f : \mathbb{R}^n \to \mathbb{R}$ is* strictly convex *if for all $x \neq y$ and all $\lambda \in (0, 1)$, we have*

$$f(\lambda x + (1 - \lambda)y) < \lambda f(x) + (1 - \lambda)f(y). \tag{2.21}$$

Definition 2.11. *A twice differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ is* strongly convex *if its Hessian $H(x)$ satisfies*

$$H(x) \succeq mI, \tag{2.22}$$

for a scalar $m > 0$ and for all x.

We have the standard implications

$$\text{strong convexity} \implies \text{strict convexity} \implies \text{convexity}, \tag{2.23}$$

but none of the converse implications is true.

2.3.2 Degrees that are easy

From the implications in (2.23) and our previous discussion, it is clear that odd degree polynomials can never be strictly convex or strongly convex. We cover the case of quadratic polynomials in the following straightforward proposition.

Proposition 2.12. *For a quadratic polynomial $p(x) = \frac{1}{2}x^TQx + q^Tx + c$, the notions of strict convexity and strong convexity are equivalent, and can be decided in polynomial time.*

Proof. Strong convexity always implies strict convexity. For the reverse direction, assume that $p(x)$ is not strongly convex. In view of (2.22), this means that the matrix Q is not positive definite. If Q has a negative eigenvalue, $p(x)$ is not convex, let alone strictly convex. If Q has a zero eigenvalue, let $\bar{x} \neq 0$ be the corresponding eigenvector. Then $p(x)$ restricted to the line from the origin to $\bar{x}$ is linear and hence not strictly convex.

To see that these properties can be checked in polynomial time, note that $p(x)$ is strongly convex if and only if the symmetric matrix Q is positive definite. By Sylvester's criterion, positive definiteness of an $n \times n$ symmetric matrix is equivalent to positivity of its n leading principal minors, each of which can be computed in polynomial time. □

■ 2.3.3 Degrees that are hard

With little effort, we can extend our NP-hardness result in the previous section to address strict convexity and strong convexity.

Proposition 2.13. *It is NP-hard to decide strong convexity of polynomials of any fixed even degree $d \geq 4$.*

Proof. We give a reduction from the problem of deciding convexity of quartic forms. Given a homogenous quartic polynomial $p(x) := p(x_1, \ldots, x_n)$ and an even degree $d \geq 4$, consider the polynomial

$$q(x, x_{n+1}) := p(x) + x_{n+1}^d + \tfrac{1}{2}(x_1^2 + \cdots + x_n^2 + x_{n+1}^2) \tag{2.24}$$

in $n+1$ variables. We claim that p is convex if and only if q is strongly convex. Indeed, if $p(x)$ is convex, then so is $p(x) + x_{n+1}^d$. Therefore, the Hessian of $p(x) + x_{n+1}^d$ is PSD. On the other hand, the Hessian of the term $\frac{1}{2}(x_1^2 + \cdots + x_n^2 + x_{n+1}^2)$ is the identity matrix. So, the minimum eigenvalue of the Hessian of $q(x, x_{n+1})$ is positive and bounded below by one. Hence, q is strongly convex.

Now suppose that $p(x)$ is not convex. Let us denote the Hessians of p and q respectively by H_p and H_q. If p is not convex, then there exists a point $\bar{x} \in \mathbb{R}^n$ such that

$$\lambda_{\min}(H_p(\bar{x})) < 0,$$

where $\lambda_{\min}$ here denotes the minimum eigenvalue. Because $p(x)$ is homogenous of degree four, we have

$$\lambda_{\min}(H_p(c\bar{x})) = c^2 \lambda_{\min}(H_p(\bar{x})),$$

for any scalar $c \in \mathbb{R}$. Pick c large enough such that $\lambda_{\min}(H_p(c\bar{x})) < 1$. Then it is easy to see that $H_q(c\bar{x}, 0)$ has a negative eigenvalue and hence q is not convex, let alone strongly convex. □

Remark 2.3.1. It is worth noting that homogeneous polynomials of degree $d > 2$ can never be strongly convex (because their Hessians vanish at the origin). Not surprisingly, the polynomial q in the proof of Proposition 2.13 is not homogeneous.

Proposition 2.14. *It is NP-hard to decide strict convexity of polynomials of any fixed even degree $d \geq 4$.*

Proof. The proof is almost identical to the proof of Proposition 2.13. Let q be defined as in (2.24). If p is convex, then we established that q is strongly convex and hence also strictly convex. If p is not convex, we showed that q is not convex and hence also not strictly convex. □

2.4 Complexity of deciding quasiconvexity and pseudoconvexity

2.4.1 Definitions and basics

Definition 2.15. *A function $f : \mathbb{R}^n \to \mathbb{R}$ is* quasiconvex *if its sublevel sets*

$$\mathcal{S}(\alpha) := \{x \in \mathbb{R}^n \mid f(x) \leq \alpha\}, \tag{2.25}$$

for all $\alpha \in \mathbb{R}$, are convex.

Definition 2.16. *A differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ is* pseudoconvex *if the implication*

$$\nabla f(x)^T (y - x) \geq 0 \implies f(y) \geq f(x) \tag{2.26}$$

holds for all x and y in $\mathbb{R}^n$.

The following implications are well-known (see e.g. [25, p. 143]):

$$\text{convexity} \implies \text{pseudoconvexity} \implies \text{quasiconvexity}, \tag{2.27}$$

but the converse of neither implication is true in general.

2.4.2 Degrees that are easy

As we remarked earlier, linear polynomials are always convex and hence also pseudoconvex and quasiconvex. Unlike convexity, however, it is possible for polynomials of odd degree $d \geq 3$ to be pseudoconvex or quasiconvex. We will show in this section that somewhat surprisingly, quasiconvexity and pseudoconvexity of polynomials of any fixed odd degree can be decided in polynomial time. Before we present these results, we will cover the easy case of quadratic polynomials.

Proposition 2.17. *For a quadratic polynomial* $p(x) = \frac{1}{2}x^TQx + q^Tx + c$*, the notions of convexity, pseudoconvexity, and quasiconvexity are equivalent, and can be decided in polynomial time.*

Proof. We argue that the quadratic polynomial $p(x)$ is convex if and only if it is quasiconvex. Indeed, if $p(x)$ is not convex, then Q has a negative eigenvalue; letting $\bar{x}$ be a corresponding eigenvector, we have that $p(t\bar{x})$ is a quadratic polynomial in t, with negative leading coefficient, so $p(t\bar{x})$ is not quasiconvex, as a function of t. This, however, implies that $p(x)$ is not quasiconvex.

We have already argued in Section 2.2.2 that convexity of quadratic polynomials can be decided in polynomial time. □

Quasiconvexity of polynomials of odd degree

In this subsection, we provide a polynomial time algorithm for checking whether an odd-degree polynomial is quasiconvex. Towards this goal, we will first show that quasiconvex polynomials of odd degree have a very particular structure (Proposition 2.20).

Our first lemma concerns quasiconvex polynomials of odd degree in one variable. The proof is easy and left to the reader. A version of this lemma is provided in [38, p. 99], though there also without proof.

Lemma 2.18. *Suppose that* $p(t)$ *is a quasiconvex univariate polynomial of odd degree. Then,* $p(t)$ *is monotonic.*

Next, we use the preceding lemma to characterize the complements of sublevel sets of quasiconvex polynomials of odd degree.

Lemma 2.19. *Suppose that* $p(x)$ *is a quasiconvex polynomial of odd degree* d*. Then the set* $\{x \mid p(x) \geq \alpha\}$ *is convex.*

Proof. Suppose not. In that case, there exist x, y, z such that z is on the line segment connecting x and y, and such that $p(x), p(y) \geq \alpha$ but $p(z) < \alpha$. Consider the polynomial

$$q(t) = p(x + t(y - x)).$$

This is, of course, a quasiconvex polynomial with $q(0) = p(x)$, $q(1) = p(y)$, and $q(t') = p(z)$, for some $t' \in (0, 1)$. If $q(t)$ has degree d, then, by Lemma 2.18, it must be monotonic, which immediately provides a contradiction.

Suppose now that $q(t)$ has degree less than d. Let us attempt to perturb x to $x + x'$, and y to $y + y'$, so that the new polynomial

$$\hat{q}(t) = p\left(x + x' + t(y + y' - x - x')\right)$$

has the following two properties: (i) $\hat{q}(t)$ is a polynomial of degree d, and (ii) $\hat{q}(0) > \hat{q}(t')$, $\hat{q}(1) > \hat{q}(t')$. If such perturbation vectors x', y' can be found, then we obtain a contradiction as in the previous paragraph.

To satisfy condition (ii), it suffices (by continuity) to take x', y' with $\|x'\|, \|y'\|$ small enough. Thus, we only need to argue that we can find arbitrarily small x', y' that satisfy condition (i). Observe that the coefficient of t^d in the polynomial $\hat{q}(t)$ is a nonzero polynomial in $x + x', y + y'$; let us denote that coefficient as $r(x + x', y + y')$. Since r is a nonzero polynomial, it cannot vanish at all points of any given ball. Therefore, even when considering a small ball around (x, y) (to satisfy condition (ii)), we can find $(x+x', y+y')$ in that ball, with $r(x+x', y+y') \neq 0$, thus establishing that the degree of $\hat{q}$ is indeed d. This completes the proof.

□

We now proceed to a characterization of quasiconvex polynomials of odd degree.

Proposition 2.20. *Let $p(x)$ be a polynomial of odd degree d. Then, $p(x)$ is quasiconvex if and only if it can be written as*

$$p(x) = h(\xi^T x), \tag{2.28}$$

for some nonzero $\xi \in \mathbb{R}^n$, and for some monotonic univariate polynomial $h(t)$ of degree d. If, in addition, we require the nonzero component of ξ with the smallest index to be equal to unity, then ξ and $h(t)$ are uniquely determined by $p(x)$.

Proof. It is easy to see that any polynomial that can be written in the above form is quasiconvex. In order to prove the converse, let us assume that $p(x)$ is quasiconvex. By the definition of quasiconvexity, the closed set $\mathcal{S}(\alpha) = \{x \mid p(x) \leq \alpha\}$ is convex. On the other hand, Lemma 2.19 states that the closure of the complement of $\mathcal{S}(\alpha)$ is also convex. It is not hard to verify that, as a consequence of these two properties, the set $\mathcal{S}(\alpha)$ must be a halfspace. Thus, for any given α, the sublevel set $\mathcal{S}(\alpha)$ can be written as $\{x \mid \xi(\alpha)^T x \leq c(\alpha)\}$ for some $\xi(\alpha) \in \mathbb{R}^n$ and $c(\alpha) \in \mathbb{R}$. This of course implies that the level sets $\{x \mid p(x) = \alpha\}$ are hyperplanes of the form $\{x \mid \xi(\alpha)^T x = c(\alpha)\}$.

We note that the sublevel sets are necessarily nested: if $\alpha < \beta$, then $\mathcal{S}(\alpha) \subseteq \mathcal{S}(\beta)$. An elementary consequence of this property is that the hyperplanes must be collinear, i.e., that the vectors $\xi(\alpha)$ must be positive multiples of each other. Thus, by suitably scaling the coefficients $c(\alpha)$, we can assume, without loss of generality, that $\xi(\alpha) = \xi$, for some $\xi \in \mathbb{R}^n$, and for all α. We then have that $\{x \mid p(x) = \alpha\} = \{x \mid \xi^T x = c(\alpha)\}$. Clearly, there is a one-to-one correspondence between α and $c(\alpha)$, and therefore the value of $p(x)$ is completely determined by

$\xi^T x$. In particular, there exists a function $h(t)$ such that $p(x) = h(q^T x)$. Since $p(x)$ is a polynomial of degree d, it follows that $h(t)$ is a univariate polynomial of degree d. Finally, we observe that if $h(t)$ is not monotonic, then $p(x)$ is not quasiconvex. This proves that a representation of the desired form exists. Note that by suitably scaling ξ, we can also impose the condition that the nonzero component of ξ with the smallest index is equal to one.

Suppose that now that $p(x)$ can also be represented in the form $p(x) = \bar{h}(\bar{\xi}^T x)$ for some other polynomial $\bar{h}(t)$ and vector $\bar{\xi}$. Then, the gradient vector of $p(x)$ must be proportional to both ξ and $\bar{\xi}$. The vectors ξ and $\bar{\xi}$ are therefore collinear. Once we impose the requirement that the nonzero component of ξ with the smallest index is equal to one, we obtain that $\xi = \bar{\xi}$ and, consequently, $h = \bar{h}$. This establishes the claimed uniqueness of the representation. □

Remark. It is not hard to see that if $p(x)$ is homogeneous and quasiconvex, then one can additionally conclude that $h(t)$ can be taken to be $h(t) = t^d$, where d is the degree of $p(x)$.

Theorem 2.21. *For any fixed odd degree d, the quasiconvexity of polynomials of degree d can be checked in polynomial time.*

Proof. The algorithm consists of attempting to build a representation of $p(x)$ of the form given in Proposition 2.20. The polynomial $p(x)$ is quasiconvex if and only if the attempt is successful.

Let us proceed under the assumption that $p(x)$ is quasiconvex. We differentiate $p(x)$ symbolically to obtain its gradient vector. Since a representation of the form given in Proposition 2.20 exists, the gradient is of the form $\nabla p(x) = \xi h'(\xi^T x)$, where $h'(t)$ is the derivative of $h(t)$. In particular, the different components of the gradient are polynomials that are proportional to each other. (If they are not proportional, we conclude that $p(x)$ is not quasiconvex, and the algorithm terminates.) By considering the ratios between different components, we can identify the vector ξ, up to a scaling factor. By imposing the additional requirement that the nonzero component of ξ with the smallest index is equal to one, we can identify ξ uniquely.

We now proceed to identify the polynomial $h(t)$. For $k = 1, \ldots, d+1$, we evaluate $p(k\xi)$, which must be equal to $h(\xi^T \xi k)$. We thus obtain the values of $h(t)$ at $d+1$ distinct points, from which $h(t)$ is completely determined. We then verify that $h(\xi^T x)$ is indeed equal to $p(x)$. This is easily done, in polynomial time, by writing out the $O(n^d)$ coefficients of these two polynomials in x and verifying that they are equal. (If they are not all equal, we conclude that $p(x)$ is not quasiconvex, and the algorithm terminates.)

Finally, we test whether the above constructed univariate polynomial h is monotonic, i.e., whether its derivative $h'(t)$ is either nonnegative or nonpositive. This can be accomplished, e.g., by quantifier elimination or by other well-known algebraic techniques for counting the number and the multiplicity of real roots of univariate polynomials; see [24]. Note that this requires only a constant number of arithmetic operations since the degree d is fixed. If h fails this test, then $p(x)$ is not quasiconvex. Otherwise, our attempt has been successful and we decide that $p(x)$ is indeed quasiconvex. □

Pseudoconvexity of polynomials of odd degree

In analogy to Proposition 2.20, we present next a characterization of odd degree pseudoconvex polynomials, which gives rise to a polynomial time algorithm for checking this property.

Corollary 2.22. *Let $p(x)$ be a polynomial of odd degree d. Then, $p(x)$ is pseudoconvex if and only if $p(x)$ can be written in the form*

$$p(x) = h(\xi^T x), \tag{2.29}$$

for some $\xi \in \mathbb{R}^n$ and some univariate polynomial h of degree d such that its derivative $h'(t)$ has no real roots.

Remark. Observe that polynomials h with h' having no real roots comprise a subset of the set of monotonic polynomials.

Proof. Suppose that $p(x)$ is pseudoconvex. Since a pseudoconvex polynomial is quasiconvex, it admits a representation $h(\xi^T x)$ where h is monotonic. If $h'(t) = 0$ for some t, then picking $a = t \cdot \xi / \|\xi\|_2^2$, we have that $\nabla p(a) = 0$, so that by pseudoconvexity, $p(x)$ is minimized at a. This, however, is impossible since an odd degree polynomial is never bounded below. Conversely, suppose $p(x)$ can be represented as in Eq. (2.29). Fix some x, y, and define the polynomial $u(t) = p(x + t(y - x))$. Since $u(t) = h(\xi^T x + t\xi^T(y - x))$, we have that either (i) $u(t)$ is constant, or (ii) $u'(t)$ has no real roots. Now if $\nabla p(x)(y - x) \geq 0$, then $u'(0) \geq 0$. Regardless of whether (i) or (ii) holds, this implies that $u'(t) \geq 0$ everywhere, so that $u(1) \geq u(0)$ or $p(y) \geq p(x)$. □

Corollary 2.23. *For any fixed odd degree d, the pseudoconvexity of polynomials of degree d can be checked in polynomial time.*

Proof. This is a simple modification of our algorithm for testing quasiconvexity (Theorem 2.21). The first step of the algorithm is in fact identical: once we impose the additional requirement that the nonzero component of ξ with the smallest index should be equal to one, we can uniquely determine the vector ξ and the coefficients of the univariate polynomial $h(t)$ that satisfy Eq. (2.29) . (If we fail, $p(x)$ is not quasiconvex and hence also not pseudoconvex.) Once we have $h(t)$, we can check whether $h'(t)$ has no real roots e.g. by computing the signature of the Hermite form of $h'(t)$; see [24].

□

Remark 2.4.1. Homogeneous polynomials of odd degree $d \geq 3$ are never pseudoconvex. The reason is that the gradient of these polynomials vanishes at the origin, but yet the origin is not a global minimum since odd degree polynomials are unbounded below.

2.4.3 Degrees that are hard

The main result of this section is the following theorem.

Theorem 2.24. *It is NP-hard to check quasiconvexity/pseudoconvexity of degree four polynomials. This is true even when the polynomials are restricted to be homogeneous.*

In view of Theorem 2.1, which established NP-hardness of deciding convexity of *homogeneous* quartic polynomials, Theorem 2.24 follows immediately from the following result.

Theorem 2.25. *For a homogeneous polynomial $p(x)$ of even degree d, the notions of convexity, pseudoconvexity, and quasiconvexity are all equivalent.*[5]

We start the proof of this theorem by first proving an easy lemma.

Lemma 2.26. *Let $p(x)$ be a quasiconvex homogeneous polynomial of even degree $d \geq 2$. Then $p(x)$ is nonnegative.*

Proof. Suppose, to derive a contradiction, that there exist some $\epsilon > 0$ and $\bar{x} \in \mathbb{R}^n$ such that $p(\bar{x}) = -\epsilon$. Then by homogeneity of even degree we must have $p(-\bar{x}) = p(\bar{x}) = -\epsilon$. On the other hand, homogeneity of p implies that $p(0) = 0$. Since the origin is on the line between $\bar{x}$ and $-\bar{x}$, this shows that the sublevel set $\mathcal{S}(-\epsilon)$ is not convex, contradicting the quasiconvexity of p. □

[5] The result is more generally true for differentiable functions that are homogeneous of even degree. Also, the requirements of homogeneity and having an even degree both need to be present. Indeed, x^3 and $x^4 - 8x^3 + 18x^2$ are both quasiconvex but not convex, the first being homogeneous of odd degree and the second being nonhomogeneous of even degree.

Proof of Theorem 2.25. We show that a quasiconvex homogeneous polynomial of even degree is convex. In view of implication (2.27), this proves the theorem.

Suppose that $p(x)$ is a quasiconvex polynomial. Define $\mathcal{S} = \{x \in \mathbb{R}^n \mid p(x) \leq 1\}$. By homogeneity, for any $a \in \mathbb{R}^n$ with $p(a) > 0$, we have that

$$\frac{a}{p(a)^{1/d}} \in \mathcal{S}.$$

By quasiconvexity, this implies that for any a, b with $p(a), p(b) > 0$, any point on the line connecting $a/p(a)^{1/d}$ and $b/p(b)^{1/d}$ is in $\mathcal{S}$. In particular, consider

$$c = \frac{a+b}{p(a)^{1/d} + p(b)^{1/d}}.$$

Because c can be written as

$$c = \left(\frac{p(a)^{1/d}}{p(a)^{1/d} + p(b)^{1/d}}\right)\left(\frac{a}{p(a)^{1/d}}\right) + \left(\frac{p(b)^{1/d}}{p(a)^{1/d} + p(b)^{1/d}}\right)\left(\frac{b}{p(b)^{1/d}}\right),$$

we have that $c \in \mathcal{S}$, i.e., $p(c) \leq 1$. By homogeneity, this inequality can be restated as

$$p(a+b) \leq (p(a)^{1/d} + p(b)^{1/d})^d,$$

and therefore

$$p\Big(\frac{a+b}{2}\Big) \leq \left(\frac{p(a)^{1/d} + p(b)^{1/d}}{2}\right)^d \leq \frac{p(a)+p(b)}{2}, \tag{2.30}$$

where the last inequality is due to the convexity of x^d.

Finally, note that for any polynomial p, the set $\{x \mid p(x) \neq 0\}$ is dense in $\mathbb{R}^n$ (here we again appeal to the fact that the only polynomial that is zero on a ball of positive radius is the zero polynomial); and since p is nonnegative due to Lemma 2.26, the set $\{x \mid p(x) > 0\}$ is dense in $\mathbb{R}^n$. Using the continuity of p, it follows that Eq. (2.30) holds not only when a, b satisfy $p(a), p(b) > 0$, but for all a, b. Appealing to the continuity of p again, we see that for all a, b, $p(\lambda a + (1-\lambda)b) \leq \lambda p(a) + (1-\lambda)p(b)$, for all $\lambda \in [0,1]$. This establishes that p is convex.

□

Quasiconvexity/pseudoconvexity of polynomials of even degree larger than four.

Corollary 2.27. *It is NP-hard to decide quasiconvexity of polynomials of any fixed even degree $d \geq 4$.*

Proof. We have already proved the result for $d = 4$. To establish the result for even degree $d \geq 6$, recall that we have established NP-hardness of deciding convexity of homogeneous quartic polynomials. Given such a quartic form $p(x) := p(x_1, \ldots, x_n)$, consider the polynomial

$$q(x_1, \ldots, x_{n+1}) = p(x_1, \ldots, x_n) + x_{n+1}^d. \tag{2.31}$$

We claim that q is quasiconvex if and only if p is convex. Indeed, if p is convex, then obviously so is q, and therefore q is quasiconvex. Conversely, if p is not convex, then by Theorem 2.25, it is not quasiconvex. So, there exist points $a, b, c \in \mathbb{R}^n$, with c on the line connecting a and b, such that $p(a) \leq 1$, $p(b) \leq 1$, but $p(c) > 1$. Considering points $(a, 0)$, $(b, 0)$, $(c, 0)$, we see that q is not quasiconvex. It follows that it is NP-hard to decide quasiconvexity of polynomials of even degree four or larger. □

Corollary 2.28. *It is NP-hard to decide pseudoconvexity of polynomials of any fixed even degree $d \geq 4$.*

Proof. The proof is almost identical to the proof of Corollary 2.27. Let q be defined as in (2.31). If p is convex, then q is convex and hence also pseudoconvex. If p is not convex, we showed that q is not quasiconvex and hence also not pseudoconvex. □

■ 2.5 Summary and conclusions

In this chapter, we studied the computational complexity of testing convexity and some of its variants, for polynomial functions. The notions that we considered and the implications among them are summarized below:

strong convexity $\Longrightarrow$ strict convexity $\Longrightarrow$ convexity $\Longrightarrow$ pseudoconvexity $\Longrightarrow$ quasiconvexity.

Our complexity results as a function of the degree of the polynomial are listed in Table 2.1. We gave polynomial time algorithms for checking pseudoconvexity and quasiconvexity of odd degree polynomials that can be useful in many applications. Our negative results, on the other hand, imply (under P$\neq$NP) the impossibility of a polynomial time (or even pseudo-polynomial time) algorithm for testing any of the properties listed in Table 2.1 for polynomials of even degree four or larger. Although the implications of convexity are very significant in optimization theory, our results suggest that unless additional structure is present, ensuring the mere presence of convexity is likely an intractable task. It is therefore natural to wonder whether there are other properties of optimization problems

property vs. degree	1	2	odd ≥ 3	even ≥ 4
strong convexity	no	P	no	strongly NP-hard
strict convexity	no	P	no	strongly NP-hard
convexity	yes	P	no	strongly NP-hard
pseudoconvexity	yes	P	P	strongly NP-hard
quasiconvexity	yes	P	P	strongly NP-hard

Table 2.1. Summary of our complexity results. A yes (no) entry means that the question is trivial for that particular entry because the answer is always yes (no) independent of the input. By P, we mean that the problem can be solved in polynomial time.

that share some of the attractive consequences of convexity, but are easier to check.

The hardness results of this chapter also lay emphasis on the need for finding good approximation algorithms for recognizing convexity that can deal with a large number of instances. This is our motivation for the next chapter as we turn our attention to the study of algebraic counterparts of convexity that can be efficiently checked with semidefinite programming.

Chapter 3

Convexity and SOS-Convexity

The overall contribution of this chapter is a complete characterization of the containment of the sets of convex and sos-convex polynomials in every degree and dimension. The content of this chapter is mostly based on the work in [9], but also includes parts of [11] and [2].

3.1 Introduction

3.1.1 Nonnegativity and sum of squares

One of the cornerstones of real algebraic geometry is Hilbert's seminal paper in 1888 [77], where he gives a complete characterization of the degrees and dimensions in which nonnegative polynomials can be written as sums of squares of polynomials. In particular, Hilbert proves in [77] that there exist nonnegative polynomials that are not sums of squares, although explicit examples of such polynomials appeared only about 80 years later and the study of the gap between nonnegative and sums of squares polynomials continues to be an active area of research to this day.

Motivated by a wealth of new applications and a modern viewpoint that emphasizes efficient computation, there has also been a great deal of recent interest from the optimization community in the representation of nonnegative polynomials as sums of squares (sos). Indeed, many fundamental problems in applied and computational mathematics can be reformulated as either deciding whether certain polynomials are nonnegative or searching over a family of nonnegative polynomials. It is well-known however that if the degree of the polynomial is four or larger, deciding nonnegativity is an NP-hard problem. (As we mentioned in the last chapter, this follows e.g. as an immediate corollary of NP-hardness of deciding matrix copositivity [109].) On the other hand, it is also well-known that deciding whether a polynomial can be written as a sum of squares can be reduced to solving a semidefinite program, for which efficient algorithms e.g. based on interior point methods is available. The general machinery of the so-called "sos relaxation" has therefore been to replace the intractable nonnegativity requirements with the more tractable sum of squares requirements that obviously provide a sufficient condition for polynomial nonnegativity.

Some relatively recent applications that sum of squares relaxations have found span areas as diverse as control theory [118], [76], quantum computation [59], polynomial games [120], combinatorial optimization [71], geometric theorem proving [123], and many others.

3.1.2 Convexity and sos-convexity

Aside from nonnegativity, *convexity* is another fundamental property of polynomials that is of both theoretical and practical significance. In the previous chapter, we already listed a number of applications of establishing convexity of polynomials including global optimization, convex envelope approximation, Lyapunov analysis, data fitting, defining norms, etc. Unfortunately, however, we also showed that just like nonnegativity, convexity of polynomials is NP-hard to decide for polynomials of degree as low as four. Encouraged by the success of sum of squares methods as a viable substitute for nonnegativity, our focus in this chapter will be on the analogue of sum of squares for polynomial convexity: a notion known as *sos-convexity*.

As we mentioned in our previous chapters in passing, sos-convexity (which gets its name from the work of Helton and Nie in [75]) is a sufficient condition for convexity of polynomials based on an appropriately defined sum of squares decomposition of the Hessian matrix; see the equivalent Definitions 2.4 and 3.4. The main computational advantage of sos-convexity stems from the fact that the problem of deciding whether a given polynomial is sos-convex amounts to solving a single semidefinite program. We will explain how this is exactly done in Section 3.2 of this chapter where we briefly review the well-known connection between sum of squares decomposition and semidefinite programming.

Besides its computational implications, sos-convexity is an appealing concept since it bridges the geometric and algebraic aspects of convexity. Indeed, while the usual definition of convexity is concerned only with the geometry of the epigraph, in sos-convexity this geometric property (or the nonnegativity of the Hessian) must be certified through a "simple" algebraic identity, namely the sum of squares factorization of the Hessian. The original motivation of Helton and Nie for defining sos-convexity was in relation to the question of semidefinite representability of convex sets [75]. But this notion has already appeared in the literature in a number of other settings [89], [90], [100], [44]. In particular, there has been much recent interest in the role of convexity in semialgebraic geometry [89], [26], [55], [91] and sos-convexity is a recurrent figure in this line of research.

3.1.3 Contributions and organization of this chapter

The main contribution of this chapter is to establish the counterpart of Hilbert's characterization of the gap between nonnegativity and sum of squares for the notions of convexity and sos-convexity. We start by presenting some background material in Section 3.2. In Section 3.3, we prove an algebraic analogue of a classical result in convex analysis, which provides three equivalent characterizations for sos-convexity (Theorem 3.5). This result substantiates the fact that sos-convexity is *the* right sos relaxation for convexity. In Section 3.4, we present two explicit examples of convex polynomials that are not sos-convex, one of them being the first known such example. In Section 3.5, we provide the characterization of the gap between convexity and sos-convexity (Theorem 3.8 and Theorem 3.9). Subsection 3.5.1 includes the proofs of the cases where convexity and sos-convexity are equivalent and Subsection 3.5.2 includes the proofs of the cases where they are not. In particular, Theorem 3.16 and Theorem 3.17 present explicit examples of convex but not sos-convex polynomials that have dimension and degree as low as possible, and Theorem 3.18 provides a general construction for producing such polynomials in higher degrees. Some concluding remarks and an open problem are presented in Section 3.6.

This chapter also includes two appendices. In Appendix A, we explain how the first example of a convex but not sos-convex polynomial was found with software using sum of squares programming techniques and the duality theory of semidefinite optimization. As a byproduct of this numerical procedure, we obtain a simple method for searching over a restricted family of nonnegative polynomials that are not sums of squares. In Appendix B, we give a formal (computer assisted) proof of validity of one of our minimal convex but not sos-convex polynomials.

3.2 Preliminaries

3.2.1 Background on nonnegativity and sum of squares

For the convenience of the reader, we recall some basic concepts from the previous chapter and then introduce some new ones. We will be concerned throughout this chapter with polynomials with real coefficients. The ring of polynomials in n variables with real coefficients is denoted by $\mathbb{R}[x]$. A polynomial p is said to be *nonnegative* or *positive semidefinite (psd)* if $p(x) \geq 0$ for all $x \in \mathbb{R}^n$. We say that p is a *sum of squares (sos)*, if there exist polynomials $q_1, \ldots, q_m$ such that $p = \sum_{i=1}^m q_i^2$. We denote the set of psd (resp. sos) polynomials in n variables and degree d by $\tilde{P}_{n,d}$ (resp. $\tilde{\Sigma}_{n,d}$). Any sos polynomial is clearly psd, so we have $\tilde{\Sigma}_{n,d} \subseteq \tilde{P}_{n,d}$. Recall that a *homogeneous polynomial* (or a *form*) is a polynomial where all the monomials have the same degree. A form p of degree d is a homogeneous function of degree d since it satisfies $p(\lambda x) = \lambda^d p(x)$ for any scalar $\lambda \in \mathbb{R}$. We say that a form p is *positive definite* if $p(x) > 0$ for all $x \neq 0$ in $\mathbb{R}^n$. Following standard notation, we denote the set of psd (resp. sos) homogeneous polynomials in n variables and degree d by $P_{n,d}$ (resp. $\Sigma_{n,d}$). Once again, we have the obvious inclusion $\Sigma_{n,d} \subseteq P_{n,d}$. All of the four sets $\Sigma_{n,d}, P_{n,d}, \tilde{\Sigma}_{n,d}, \tilde{P}_{n,d}$ are closed convex cones. The closedness of the sum of squares cone may not be so obvious. This fact was first proved by Robinson [141]. We will make crucial use of it in the proof of Theorem 3.5 in the next section.

Any form of degree d in n variables can be “dehomogenized” into a polynomial of degree $\leq d$ in $n-1$ variables by setting $x_n = 1$. Conversely, any polynomial p of degree d in n variables can be “homogenized” into a form p_h of degree d in $n+1$ variables, by adding a new variable y, and letting

$$p_h(x_1, \ldots, x_n, y) := y^d \, p\left(x_1/y, \ldots, x_n/y\right).$$

The properties of being psd and sos are preserved under homogenization and dehomogenization [138].

A very natural and fundamental question that as we mentioned earlier was answered by Hilbert is to understand in what dimensions and degrees nonnegative polynomials (or forms) can be represented as sums of squares, i.e, for what values of n and d we have $\tilde{\Sigma}_{n,d} = \tilde{P}_{n,d}$ or $\Sigma_{n,d} = P_{n,d}$. Note that because of the argument in the last paragraph, we have $\tilde{\Sigma}_{n,d} = \tilde{P}_{n,d}$ if and only if $\Sigma_{n+1,d} = P_{n+1,d}$. Hence, it is enough to answer the question just for polynomials or just for forms and the answer to the other one comes for free.

Theorem 3.1 (Hilbert, [77])**.** *$\tilde{\Sigma}_{n,d} = \tilde{P}_{n,d}$ if and only if $n = 1$ or $d = 2$ or $(n,d) = (2,4)$. Equivalently, $\Sigma_{n,d} = P_{n,d}$ if and only if $n = 2$ or $d = 2$ or $(n,d) = (3,4)$.*

The proofs of $\tilde{\Sigma}_{1,d} = \tilde{P}_{1,d}$ and $\tilde{\Sigma}_{n,2} = \tilde{P}_{n,2}$ are relatively simple and were known before Hilbert. On the other hand, the proof of the fairly surprising fact that $\tilde{\Sigma}_{2,4} = \tilde{P}_{2,4}$ (or equivalently $\Sigma_{3,4} = P_{3,4}$) is rather involved. We refer the interested reader to [130], [128], [46], and references in [138] for some modern expositions and alternative proofs of this result. Hilbert's other main contribution was to show that these are the only cases where nonnegativity and sum of squares are equivalent by giving a nonconstructive proof of existence of polynomials in $\tilde{P}_{2,6} \setminus \tilde{\Sigma}_{2,6}$ and $\tilde{P}_{3,4} \setminus \tilde{\Sigma}_{3,4}$ (or equivalently forms in $P_{3,6} \setminus \Sigma_{3,6}$ and $P_{4,4} \setminus \Sigma_{4,4}$). From this, it follows with simple arguments that in all higher dimensions and degrees there must also be psd but not sos polynomials; see [138]. Explicit examples of such polynomials appeared in the 1960s starting from the celebrated Motzkin form [107]:

$$M(x_1, x_2, x_3) = x_1^4 x_2^2 + x_1^2 x_2^4 - 3x_1^2 x_2^2 x_3^2 + x_3^6, \tag{3.1}$$

which belongs to $P_{3,6} \setminus \Sigma_{3,6}$, and continuing a few years later with the Robinson form [141]:

$$R(x_1, x_2, x_3, x_4) = x_1^2(x_1 - x_4)^2 + x_2^2(x_2 - x_4)^2 + x_3^2(x_3 - x_4)^2 + 2x_1x_2x_3(x_1 + x_2 + x_3 - 2x_4), \tag{3.2}$$

which belongs to $P_{4,4} \setminus \Sigma_{4,4}$.

Several other constructions of psd polynomials that are not sos have appeared in the literature since. An excellent survey is [138]. See also [139] and [27].

3.2.2 Connection to semidefinite programming and matrix generalizations

As we remarked before, what makes sum of squares an appealing concept from a computational viewpoint is its relation to semidefinite programming. It is well-known (see e.g. [118], [119]) that a polynomial p in n variables and of even degree d is a sum of squares if and only if there exists a positive semidefinite matrix Q (often called the Gram matrix) such that

$$p(x) = z^T Q z,$$

where z is the vector of monomials of degree up to $d/2$

$$z = [1, x_1, x_2, \ldots, x_n, x_1x_2, \ldots, x_n^{d/2}]. \tag{3.3}$$

The set of all such matrices Q is the feasible set of a semidefinite program (SDP). For fixed d, the size of this semidefinite program is polynomial in n. Semidefinite programs can be solved with arbitrary accuracy in polynomial time. There are several implementations of semidefinite programming solvers, based on interior point algorithms among others, that are very efficient in practice and widely used; see [162] and references therein.

The notions of positive semidefiniteness and sum of squares of scalar polynomials can be naturally extended to polynomial matrices, i.e., matrices with entries in $\mathbb{R}[x]$. We say that a symmetric polynomial matrix $U(x) \in \mathbb{R}[x]^{m\times m}$ is positive semidefinite if $U(x)$ is positive semidefinite in the matrix sense for all $x \in \mathbb{R}^n$, i.e, if $U(x)$ has nonnegative eigenvalues for all $x \in \mathbb{R}^n$. It is straightforward to see that this condition holds if and only if the polynomial $y^T U(x) y$ in $m + n$ variables $[x; y]$ is psd. A homogeneous polynomial matrix $U(x)$ is said to be positive definite, if it is positive definite in the matrix sense, i.e., has positive eigenvalues, for all $x \neq 0$ in $\mathbb{R}^n$. The definition of an sos-matrix is as follows [88], [62], [152].

Definition 3.2. *A symmetric polynomial matrix* $U(x) \in \mathbb{R}[x]^{m\times m}$, $x \in \mathbb{R}^n$, *is an* sos-matrix *if there exists a polynomial matrix* $V(x) \in \mathbb{R}[x]^{s\times m}$ *for some* $s \in \mathbb{N}$, *such that* $U(x) = V^T(x)V(x)$.

It turns out that a polynomial matrix $U(x) \in \mathbb{R}[x]^{m\times m}$, $x \in \mathbb{R}^n$, is an sos-matrix if and only if the scalar polynomial $y^T U(x) y$ is a sum of squares in $\mathbb{R}[x; y]$; see [88]. This is a useful fact because in particular it gives us an easy way of checking whether a polynomial matrix is an sos-matrix by solving a semidefinite program. Once again, it is obvious that being an sos-matrix is a sufficient condition for a polynomial matrix to be positive semidefinite.

3.2.3 Background on convexity and sos-convexity

A polynomial p is (globally) convex if for all x and y in $\mathbb{R}^n$ and all $\lambda \in [0, 1]$, we have

$$p(\lambda x + (1-\lambda)y) \leq \lambda p(x) + (1-\lambda)p(y). \tag{3.4}$$

Since polynomials are continuous functions, the inequality in (3.4) holds if and only if it holds for a fixed value of $\lambda \in (0, 1)$, say, $\lambda = \frac{1}{2}$. In other words, p is convex if and only if

$$p\left(\tfrac{1}{2}x + \tfrac{1}{2}y\right) \leq \tfrac{1}{2}p(x) + \tfrac{1}{2}p(y) \tag{3.5}$$

for all x and y; see e.g. [148, p. 71]. Recall from the previous chapter that except for the trivial case of linear polynomials, an odd degree polynomial is clearly never convex.

For the sake of direct comparison with a result that we derive in the next section (Theorem 3.5), we recall next a classical result from convex analysis on the first and second order characterization of convexity. The proof can be found in many convex optimization textbooks, e.g. [38, p. 70]. The theorem is of course true for any twice differentiable function, but for our purposes we state it for polynomials.

Theorem 3.3. *Let* $p := p(x)$ *be a polynomial. Let* $\nabla p := \nabla p(x)$ *denote its gradient and let* $H := H(x)$ *be its Hessian, i.e., the* $n \times n$ *symmetric matrix of second derivatives. Then the following are equivalent.*

(a) $p\left(\frac{1}{2}x + \frac{1}{2}y\right) \leq \frac{1}{2}p(x) + \frac{1}{2}p(y), \quad \forall x, y \in \mathbb{R}^n$; *(i.e., p is convex).*

(b) $p(y) \geq p(x) + \nabla p(x)^T (y - x), \quad \forall x, y \in \mathbb{R}^n$.

(c) $y^T H(x) y \geq 0, \quad \forall x, y \in \mathbb{R}^n$; *(i.e., $H(x)$ is a positive semidefinite polynomial matrix).*

Helton and Nie proposed in [75] the notion of *sos-convexity* as an sos relaxation for the second order characterization of convexity (condition **(c)** above).

Definition 3.4. *A polynomial p is* sos-convex *if its Hessian $H := H(x)$ is an sos-matrix.*

With what we have discussed so far, it should be clear that sos-convexity is a sufficient condition for convexity of polynomials that can be checked with semidefinite programming. In the next section, we will show some other natural sos relaxations for polynomial convexity, which will turn out to be equivalent to sos-convexity.

We end this section by introducing some final notation: $\tilde{C}_{n,d}$ and $\tilde{\Sigma C}_{n,d}$ will respectively denote the set of convex and sos-convex polynomials in n variables and degree d; $C_{n,d}$ and $\Sigma C_{n,d}$ will respectively denote set of convex and sos-convex homogeneous polynomials in n variables and degree d. Again, these four sets are closed convex cones and we have the obvious inclusions $\tilde{\Sigma C}_{n,d} \subseteq \tilde{C}_{n,d}$ and $\Sigma C_{n,d} \subseteq C_{n,d}$.

3.3 Equivalent algebraic relaxations for convexity of polynomials

An obvious way to formulate alternative sos relaxations for convexity of polynomials is to replace every inequality in Theorem 3.3 with its sos version. In this section we examine how these relaxations relate to each other. We also comment on the size of the resulting semidefinite programs.

Our result below can be thought of as an algebraic analogue of Theorem 3.3.

Theorem 3.5. *Let $p := p(x)$ be a polynomial of degree d in n variables with its gradient and Hessian denoted respectively by $\nabla p := \nabla p(x)$ and $H := H(x)$. Let g_λ, g_∇, and g_{∇^2} be defined as*

$$\begin{array}{rcl} g_\lambda(x,y) & = & (1-\lambda)p(x) + \lambda p(y) - p((1-\lambda)x + \lambda y), \\ g_\nabla(x,y) & = & p(y) - p(x) - \nabla p(x)^T(y-x), \\ g_{\nabla^2}(x,y) & = & y^T H(x) y. \end{array} \tag{3.6}$$

Then the following are equivalent:

(a) $g_{\frac{1}{2}}(x,y)$ *is sos*[1].

(b) $g_\nabla(x,y)$ *is sos.*

(c) $g_{\nabla^2}(x,y)$ *is sos; (i.e., $H(x)$ is an sos-matrix).*

Proof. **(a)⇒(b)**: Assume $g_{\frac{1}{2}}$ is sos. We start by proving that $g_{\frac{1}{2^k}}$ will also be sos for any integer $k \geq 2$. A little bit of straightforward algebra yields the relation

$$g_{\frac{1}{2^{k+1}}}(x,y) = \tfrac{1}{2} g_{\frac{1}{2^k}}(x,y) + g_{\frac{1}{2}}\left(x, \tfrac{2^k-1}{2^k}x + \tfrac{1}{2^k}y\right). \tag{3.7}$$

[1]The constant $\frac{1}{2}$ in $g_{\frac{1}{2}}(x,y)$ of condition **(a)** is arbitrary and is chosen for convenience. One can show that $g_{\frac{1}{2}}$ being sos implies that g_λ is sos for any fixed $\lambda \in [0,1]$. Conversely, if g_λ is sos for some $\lambda \in (0,1)$, then $g_{\frac{1}{2}}$ is sos. The proofs are similar to the proof of **(a)⇒(b)**.

The second term on the right hand side of (3.7) is always sos because $g_{\frac{1}{2}}$ is sos. Hence, this relation shows that for any k, if $g_{\frac{1}{2^k}}$ is sos, then so is $g_{\frac{1}{2^{k+1}}}$. Since for $k = 1$, both terms on the right hand side of (3.7) are sos by assumption, induction immediately gives that $g_{\frac{1}{2^k}}$ is sos for all k.

Now, let us rewrite g_λ as

$$g_\lambda(x, y) = p(x) + \lambda(p(y) - p(x)) - p(x + \lambda(y - x)).$$

We have

$$\frac{g_\lambda(x, y)}{\lambda} = p(y) - p(x) - \frac{p(x + \lambda(y - x)) - p(x)}{\lambda}. \tag{3.8}$$

Next, we take the limit of both sides of (3.8) by letting $\lambda = \frac{1}{2^k} \to 0$ as $k \to \infty$. Because p is differentiable, the right hand side of (3.8) will converge to g_∇. On the other hand, our preceding argument implies that $\frac{g_\lambda}{\lambda}$ is an sos polynomial (of degree d in $2n$ variables) for any $\lambda = \frac{1}{2^k}$. Moreover, as λ goes to zero, the coefficients of $\frac{g_\lambda}{\lambda}$ remain bounded since the limit of this sequence is g_∇, which must have bounded coefficients (see (3.6)). By closedness of the sos cone, we conclude that the limit g_∇ must be sos.

(b)⇒(a): Assume g_∇ is sos. It is easy to check that

$$g_{\frac{1}{2}}(x, y) = \tfrac{1}{2} g_\nabla\left(\tfrac{1}{2}x + \tfrac{1}{2}y, x\right) + \tfrac{1}{2} g_\nabla\left(\tfrac{1}{2}x + \tfrac{1}{2}y, y\right),$$

and hence $g_{\frac{1}{2}}$ is sos.

(b)⇒(c): Let us write the second order Taylor approximation of p around x:

$$\begin{aligned} p(y) = \ & p(x) + \nabla^T p(x)(y - x) \\ & + \tfrac{1}{2}(y - x)^T H(x)(y - x) + o(||y - x||^2). \end{aligned}$$

After rearranging terms, letting $y = x + \epsilon z$ (for $\epsilon > 0$), and dividing both sides by ϵ^2 we get:

$$(p(x + \epsilon z) - p(x))/\epsilon^2 - \nabla^T p(x) z/\epsilon = \frac{1}{2} z^T H(x) z + 1/\epsilon^2 o(\epsilon^2 ||z||^2). \tag{3.9}$$

The left hand side of (3.9) is $g_\nabla(x, x + \epsilon z)/\epsilon^2$ and therefore for any fixed $\epsilon > 0$, it is an sos polynomial by assumption. As we take $\epsilon \to 0$, by closedness of the sos cone, the left hand side of (3.9) converges to an sos polynomial. On the other hand, as the limit is taken, the term $\frac{1}{\epsilon^2} o(\epsilon^2 ||z||^2)$ vanishes and hence we have that $z^T H(x) z$ must be sos.

(c)⇒(b): Following the strategy of the proof of the classical case in [160, p. 165], we start by writing the Taylor expansion of p around x with the integral form of the remainder:

$$p(y) = p(x) + \nabla^T p(x)(y - x) + \int_0^1 (1 - t)(y - x)^T H(x + t(y - x))(y - x) dt. \tag{3.10}$$

Since $y^T H(x)y$ is sos by assumption, for any $t \in [0, 1]$ the integrand

$$(1-t)(y-x)^T H(x+t(y-x))(y-x)$$

is an sos polynomial of degree d in x and y. From (3.10) we have

$$g_\nabla = \int_0^1 (1-t)(y-x)^T H(x+t(y-x))(y-x)dt.$$

It then follows that g_∇ is sos because integrals of sos polynomials, if they exist, are sos.

□

We conclude that conditions **(a)**, **(b)**, and **(c)** are equivalent sufficient conditions for convexity of polynomials, and can each be checked with a semidefinite program as explained in Subsection 3.2.2. It is easy to see that all three polynomials $g_{\frac{1}{2}}(x, y)$, $g_\nabla(x, y)$, and $g_{\nabla^2}(x, y)$ are polynomials in $2n$ variables and of degree d. (Note that each differentiation reduces the degree by one.) Each of these polynomials have a specific structure that can be exploited for formulating smaller SDPs. For example, the symmetries $g_{\frac{1}{2}}(x, y) = g_{\frac{1}{2}}(y, x)$ and $g_{\nabla^2}(x, -y) = g_{\nabla^2}(x, y)$ can be taken advantage of via symmetry reduction techniques developed in [62].

The issue of symmetry reduction aside, we would like to point out that formulation **(c)** (which was the original definition of sos-convexity) can be significantly more efficient than the other two conditions. The reason is that the polynomial $g_{\nabla^2}(x, y)$ is always quadratic and homogeneous in y and of degree $d-2$ in x. This makes $g_{\nabla^2}(x, y)$ much more sparse than $g_\nabla(x, y)$ and $g_{\nabla^2}(x, y)$, which have degree d both in x and in y. Furthermore, because of the special bipartite structure of $y^T H(x)y$, only monomials of the form $x_i^k y_j$ will appear in the vector of monomials (3.3). This in turn reduces the size of the Gram matrix, and hence the size of the SDP. It is perhaps not too surprising that the characterization of convexity based on the Hessian matrix is a more efficient condition to check. After all, this is a local condition (curvature at every point in every direction must be nonnegative), whereas conditions **(a)** and **(b)** are both global.

Remark 3.3.1. There has been yet another proposal for an sos relaxation for convexity of polynomials in [44]. However, we have shown in [8] that the condition in [44] is at least as conservative as the three conditions in Theorem 3.5 and also significantly more expensive to check.

Remark 3.3.2. Just like convexity, the property of sos-convexity is preserved under restrictions to affine subspaces. This is perhaps most directly seen through characterization **(a)** of sos-convexity in Theorem 3.5, by also noting that sum of squares is preserved under restrictions. Unlike convexity however, if a polynomial is sos-convex on every line (or even on every proper affine subspace), this does not imply that the polynomial is sos-convex.

As an application of Theorem 3.5, we use our new characterization of sos-convexity to give a short proof of an interesting lemma of Helton and Nie.

Lemma 3.6. (Helton and Nie [75, Lemma 8]). *Every sos-convex form is sos.*

Proof. Let p be an sos-convex form of degree d. We know from Theorem 3.5 that sos-convexity of p is equivalent to the polynomial $g_{\frac{1}{2}}(x,y) = \frac{1}{2}p(x) + \frac{1}{2}p(y) - p\left(\frac{1}{2}x + \frac{1}{2}y\right)$ being sos. But since sos is preserved under restrictions and $p(0) = 0$, this implies that

$$g_{\frac{1}{2}}(x,0) = \tfrac{1}{2}p(x) - p(\tfrac{1}{2}x) = \left(\tfrac{1}{2} - (\tfrac{1}{2})^d\right) p(x)$$

is sos. □

Note that the same argument also shows that convex forms are psd.

3.4 Some constructions of convex but not sos-convex polynomials

It is natural to ask whether sos-convexity is not only a sufficient condition for convexity of polynomials but also a necessary one. In other words, could it be the case that if the Hessian of a polynomial is positive semidefinite, then it must factor? To give a negative answer to this question, one has to prove existence of a convex polynomial that is not sos-convex, i.e, a polynomial p for which one (and hence all) of the three polynomials $g_{\frac{1}{2}}, g_{\nabla}$, and g_{∇^2} in (3.6) are psd but not sos. Note that existence of psd but not sos polynomials does not imply existence of convex but not sos-convex polynomials on its own. The reason is that the polynomials $g_{\frac{1}{2}}, g_{\nabla}$, and g_{∇^2} all possess a very special structure.[2] For example, $y^T H(x) y$ has the structure of being quadratic in y and a Hessian in x. (Not every polynomial matrix is a valid Hessian.) The Motzkin or the Robinson polynomials in (3.1) and (3.2) for example are clearly not of this structure.

[2]There are many situations where requiring a specific structure on polynomials makes psd equivalent to sos. As an example, we know that there are forms in $P_{4,4} \setminus \Sigma_{4,4}$. However, if we require the forms to have only even monomials, then all such nonnegative forms in 4 variables and degree 4 are sums of squares [57].

3.4.1 The first example

In [11],[7], we presented the first example of a convex polynomial that is not sos-convex[3]:

$$\begin{aligned} p(x_1,x_2,x_3) \;=\;& 32x_1^8+118x_1^6x_2^2+40x_1^6x_3^2+25x_1^4x_2^4-43x_1^4x_2^2x_3^2-35x_1^4x_3^4 \\ & +3x_1^2x_2^4x_3^2-16x_1^2x_2^2x_3^4+24x_1^2x_3^6+16x_2^8+44x_2^6x_3^2+70x_2^4x_3^4 \\ & +60x_2^2x_3^6+30x_3^8. \end{aligned} \tag{3.11}$$

As we will see later in this chapter, this form which lives in $C_{3,8}\setminus\Sigma C_{3,8}$ turns out to be an example in the smallest possible number of variables but not in the smallest degree.

In Appendix A, we will explain how the polynomial in (3.11) was found. The proof that this polynomial is convex but not sos-convex is omitted and can be found in [11]. However, we would like to highlight an idea behind this proof that will be used again in this chapter. As the following lemma demonstrates, one way to ensure a polynomial is *not sos-convex* is by enforcing one of the principal minors of its Hessian matrix to be *not sos*.

Lemma 3.7. *If $P(x)\in\mathbb{R}[x]^{m\times m}$ is an sos-matrix, then all its 2^m-1 principal minors[4] are sos polynomials. In particular,* $\det(P)$ *and the diagonal elements of P must be sos polynomials.*

Proof. We first prove that $\det(P)$ is sos. By Definition 3.2, we have $P(x)=M^T(x)M(x)$ for some $s\times m$ polynomial matrix $M(x)$. If $s=m$, we have

$$\det(P)=\det(M^T)\det(M)=(\det(M))^2$$

and the result is immediate. If $s>m$, the result follows from the Cauchy-Binet

[3] Assuming P≠NP, and given the NP-hardness of deciding polynomial convexity proven in the previous chapter, one would expect to see convex polynomials that are not sos-convex. However, we found the first such polynomial before we had proven the NP-hardness result. Moreover, from complexity considerations, even assuming P≠NP, one cannot conclude existence of convex but not sos-convex polynomials for any fixed finite value of the number of variables n.

[4] The principal minors of an $m\times m$ matrix A are the determinants of all $k\times k$ $(1\leq k\leq m)$ sub-blocks whose rows and columns come from the same index set $S\subset\{1,\ldots,m\}$.

formula[5]. We have

$$\begin{aligned}\det(P) &= \textstyle\sum_S \det(M^T)_S \det(M_S) \\ &= \textstyle\sum_S \det(M_S)^T \det(M_S) \\ &= \textstyle\sum_S (\det(M_S))^2.\end{aligned}$$

Finally, when $s < m$, $\det(P)$ is zero which is trivially sos. In fact, the Cauchy-Binet formula also holds for $s = m$ and $s < m$, but we have separated these cases for clarity of presentation.

Next, we need to prove that the minors corresponding to smaller principal blocks of P are also sos. Define $\mathcal{M} = \{1, \ldots, m\}$, and let I and J be nonempty subsets of $\mathcal{M}$. Denote by P_{IJ} a sub-block of P with row indices from I and column indices from J. It is easy to see that

$$P_{JJ} = (M^T)_{J\mathcal{M}} M_{\mathcal{M}J} = (M_{\mathcal{M}J})^T M_{\mathcal{M}J}.$$

Therefore, P_{JJ} is an sos-matrix itself. By the proceeding argument $\det(P_{JJ})$ must be sos, and hence all the principal minors are sos. □

Remark 3.4.1. Interestingly, the converse of Lemma 3.7 does not hold. A counterexample is the Hessian of the form f in (3.15) that we will present in the next section. All 7 principal minors of the 3×3 Hessian this form are sos polynomials, even though the Hessian is not an sos-matrix. This is in contrast with the fact that a polynomial matrix is positive semidefinite if and only if all its principal minors are psd polynomials. The latter statement follows immediately from the well-known fact that a constant matrix is positive semidefinite if and only if all its principal minors are nonnegative.

■ 3.4.2 A "clean" example

We next present another example of a convex but not sos-convex form whose construction is in fact related to our proof of NP-hardness of deciding convexity of quartic forms from Chapter 2. The example is in $C_{6,4} \setminus \Sigma C_{6,4}$ and by contrast

[5]Given matrices A and B of size $m \times s$ and $s \times m$ respectively, the Cauchy-Binet formula states that

$$\det(AB) = \sum_S \det(A_S)\det(B_S),$$

where S is a subset of $\{1, \ldots, s\}$ with m elements, A_S denotes the $m \times m$ matrix whose columns are the columns of A with index from S, and similarly B_S denotes the $m \times m$ matrix whose rows are the rows of B with index from S.

to the example of the previous subsection, it will turn out to be minimal in the degree but not in the number of variables. What is nice about this example is that unlike the other examples in this chapter it has not been derived with the assistance of a computer and semidefinite programming:

$$\begin{aligned} q(x_1,\ldots,x_6) \quad = \quad & x_1^4 + x_2^4 + x_3^4 + x_4^4 + x_5^4 + x_6^4 \\ & +2(x_1^2x_2^2 + x_1^2x_3^2 + x_2^2x_3^2 + x_4^2x_5^2 + x_4^2x_6^2 + x_5^2x_6^2) \\ & +\tfrac{1}{2}(x_1^2x_4^2 + x_2^2x_5^2 + x_3^2x_6^2) + x_1^2x_6^2 + x_2^2x_4^2 + x_3^2x_5^2 \\ & -(x_1x_2x_4x_5 + x_1x_3x_4x_6 + x_2x_3x_5x_6). \end{aligned} \tag{3.12}$$

The proof that this polynomial is convex but not sos-convex can be extracted from Theorems 2.3 and 2.5 of Chapter 2. The reader can observe that these two theorems put together give us a general procedure for producing convex but not sos-convex quartic forms from *any* example of a psd but not sos biquadratic form[6]. The biquadratic form that has led to the form above is that of Choi in [45].

The example in (3.12) also shows that convex forms that possess strong symmetry properties can still fail to be sos-convex. The symmetries in this form are inherited from the rich symmetry structure of the biquadratic form of Choi (see [62]). In general, symmetries are of interest in the study of positive semidefinite and sums of squares polynomials because the gap between psd and sos can often behave very differently depending on the symmetry properties; see e.g. [28].

3.5 Characterization of the gap between convexity and sos-convexity

Now that we know there exist convex polynomials that are not sos-convex, our final and main goal is to give a complete characterization of the degrees and dimensions in which such polynomials can exist. This is achieved in the next theorem.

Theorem 3.8. $\Sigma\tilde{C}_{n,d} = \tilde{C}_{n,d}$ *if and only if* $n = 1$ *or* $d = 2$ *or* $(n,d) = (2,4)$.

We would also like to have such a characterization for homogeneous polynomials. Although convexity is a property that is in some sense more meaningful for nonhomogeneous polynomials than for forms, one motivation for studying convexity of forms is in their relation to norms [140]. Also, in view of the fact that

[6] The reader can refer to Definition 2.2 of the previous chapter to recall the definition of a biquadratic form.

we have a characterization of the gap between nonnegativity and sums of squares both for polynomials and for forms, it is very natural to inquire the same result for convexity and sos-convexity. The next theorem presents this characterization for forms.

Theorem 3.9. *$\Sigma C_{n,d} = C_{n,d}$ if and only if $n = 2$ or $d = 2$ or $(n, d) = (3, 4)$.*

The result $\Sigma C_{3,4} = C_{3,4}$ of this theorem is to be presented in full detail in [2]. The remainder of this chapter is solely devoted to the proof of Theorem 3.8 and the proof of Theorem 3.9 except for the case $(n, d) = (3, 4)$. Before we present these proofs, we shall make two important remarks.

Remark 3.5.1. **Difficulty with homogenization and dehomogenization.** Recall from Subsection 3.2.1 and Theorem 3.1 that characterizing the gap between nonnegativity and sum of squares for polynomials is equivalent to accomplishing this task for forms. Unfortunately, the situation is more complicated for convexity and sos-convexity and that is the reason why we are presenting Theorems 3.8 and 3.9 as separate theorems. The difficulty arises from the fact that unlike nonnegativity and sum of squares, convexity and sos-convexity are not always preserved under homogenization. (Or equivalently, the properties of being not convex and not sos-convex are not preserved under dehomogenization.) In fact, any convex polynomial that is not psd will no longer be convex after homogenization. This is because convex forms are psd but the homogenization of a non-psd polynomial is a non-psd form. Even if a convex polynomial is psd, its homogenization may not be convex. For example the univariate polynomial $10x_1^4 - 5x_1 + 2$ is convex and psd, but its homogenization $10x_1^4 - 5x_1x_2^3 + 2x_2^4$ is not convex.[7] To observe the same phenomenon for sos-convexity, consider the trivariate form p in (3.11) which is convex but not sos-convex and define $\tilde{p}(x_2, x_3) = p(1, x_2, x_3)$. Then, one can check that $\tilde{p}$ is sos-convex (i.e., its 2×2 Hessian factors) even though its homogenization which is p is not sos-convex [11].

Remark 3.5.2. **Resemblance to the result of Hilbert.** The reader may have noticed from the statements of Theorem 3.1 and Theorems 3.8 and 3.9 that the cases where convex polynomials (forms) are sos-convex are exactly the same cases where nonnegative polynomials are sums of squares! We shall emphasize that as far as we can tell, our results do not follow (except in the simplest cases) from Hilbert's result stated in Theorem 3.1. Note that the question of convexity or sos-convexity of a polynomial $p(x)$ in n variables and degree d is about the polynomials $g_{\frac{1}{2}}(x, y), g_{\nabla}(x, y)$, or $g_{\nabla^2}(x, y)$ defined in (3.6) being psd or sos. Even though these polynomials still have degree d, it is important to keep in mind that

[7] What is true however is that a nonnegative form of degree d is convex if and only if the d-th root of its dehomogenization is a convex function [140, Prop. 4.4].

they are polynomials *in* $2n$ *variables*. Therefore, there is no direct correspondence with the characterization of Hilbert. To make this more explicit, let us consider for example one particular claim of Theorem 3.9: $\Sigma C_{2,4} = C_{2,4}$. For a form p in 2 variables and degree 4, the polynomials $g_{\frac{1}{2}}, g_{\nabla}$, and g_{∇^2} will be forms in 4 variables and degree 4. We know from Hilbert's result that in this situation psd but not sos forms do in fact exist. However, for the forms in 4 variables and degree 4 that have the special structure of $g_{\frac{1}{2}}, g_{\nabla}$, or g_{∇^2}, psd turns out to be equivalent to sos.

The proofs of Theorems 3.8 and 3.9 are broken into the next two subsections. In Subsection 3.5.1, we provide the proofs for the cases where convexity and sos-convexity are equivalent. Then in Subsection 3.5.2, we prove that in all other cases there exist convex polynomials that are not sos-convex.

■ 3.5.1 Proofs of Theorems 3.8 and 3.9: cases where $\tilde{\Sigma C}_{n,d} = \tilde{C}_{n,d}$, $\Sigma C_{n,d} = C_{n,d}$

When proving equivalence of convexity and sos-convexity, it turns out to be more convenient to work with the second order characterization of sos-convexity, i.e., with the form $g_{\nabla^2}(x,y) = y^T H(x) y$ in (3.6). The reason for this is that this form is always quadratic in y, and this allows us to make use of the following key theorem, henceforth referred to as the "biform theorem".

Theorem 3.10 (e.g. [47])**.** *Let $f := f(u_1, u_2, v_1, \ldots, v_m)$ be a form in the variables $u := (u_1, u_2)^T$ and $v := (v_1, , \ldots, v_m)^T$ that is a quadratic form in v for fixed u and a form (of however large degree) in u for fixed v. Then f is psd if and only if it is sos.*[8]

The biform theorem has been proven independently by several authors. See [47] and [20] for more background on this theorem and in particular [47, Sec. 7] for a an elegant proof and some refinements. We now proceed with our proofs which will follow in a rather straightforward manner from the biform theorem.

Theorem 3.11. $\tilde{\Sigma C}_{1,d} = \tilde{C}_{1,d}$ *for all* d. $\Sigma C_{2,d} = C_{2,d}$ *for all* d.

Proof. For a univariate polynomial, convexity means that the second derivative, which is another univariate polynomial, is psd. Since $\tilde{\Sigma}_{1,d} = \tilde{P}_{1,d}$, the second derivative must be sos. Therefore, $\tilde{\Sigma C}_{1,d} = \tilde{C}_{1,d}$. To prove $\Sigma C_{2,d} = C_{2,d}$, suppose we have a convex bivariate form p of degree d in variables $x := (x_1, x_2)^T$. The Hessian $H := H(x)$ of p is a 2×2 matrix whose entries are forms of degree $d - 2$. If we let $y := (y_1, y_2)^T$, convexity of p implies that the form $y^T H(x) y$ is

[8]Note that the results $\Sigma_{2,d} = P_{2,d}$ and $\Sigma_{n,2} = P_{n,2}$ are both special cases of this theorem.

psd. Since $y^T H(x) y$ meets the requirements of the biform theorem above with $(u_1, u_2) = (x_1, x_2)$ and $(v_1, v_2) = (y_1, y_2)$, it follows that $y^T H(x) y$ is sos. Hence, p is sos-convex. □

Theorem 3.12. $\tilde{\Sigma C}_{n,2} = \tilde{C}_{n,2}$ *for all* n. $\Sigma C_{n,2} = C_{n,2}$ *for all* n.

Proof. Let $x := (x_1, \ldots, x_n)^T$ and $y := (y_1, \ldots, y_n)^T$. Let $p(x) = \frac{1}{2} x^T Q x + b^T x + c$ be a quadratic polynomial. The Hessian of p in this case is the constant symmetric matrix Q. Convexity of p implies that $y^T Q y$ is psd. But since $\Sigma_{n,2} = P_{n,2}$, $y^T Q y$ must be sos. Hence, p is sos-convex. The proof of $\Sigma C_{n,2} = C_{n,2}$ is identical. □

Theorem 3.13. $\tilde{\Sigma C}_{2,4} = \tilde{C}_{2,4}$.

Proof. Let $p(x) := p(x_1, x_2)$ be a convex bivariate quartic polynomial. Let $H := H(x)$ denote the Hessian of p and let $y := (y_1, y_2)^T$. Note that $H(x)$ is a 2×2 matrix whose entries are (not necessarily homogeneous) quadratic polynomials. Since p is convex, $y^T H(x) y$ is psd. Let $\bar{H}(x_1, x_2, x_3)$ be a 2×2 matrix whose entries are obtained by homogenizing the entries of H. It is easy to see that $y^T \bar{H}(x_1, x_2, x_3) y$ is then the form obtained by homogenizing $y^T H(x) y$ and is therefore psd. Now we can employ the biform theorem (Theorem 3.10) with $(u_1, u_2) = (y_1, y_2)$ and $(v_1, v_2, v_3) = (x_1, x_2, x_3)$ to conclude that $y^T \bar{H}(x_1, x_2, x_3) y$ is sos. But upon dehomogenizing by setting $x_3 = 1$, we conclude that $y^T H(x) y$ is sos. Hence, p is sos-convex. □

Theorem 3.14 (Ahmadi, Blekherman, Parrilo [2]). $\Sigma C_{3,4} = C_{3,4}$.

Unlike Hilbert's results $\tilde{\Sigma}_{2,4} = \tilde{P}_{2,4}$ and $\Sigma_{3,4} = P_{3,4}$ which are equivalent statements and essentially have identical proofs, the proof of $\Sigma C_{3,4} = C_{3,4}$ is considerably more involved than the proof of $\tilde{\Sigma C}_{2,4} = \tilde{C}_{2,4}$. Here, we briefly point out why this is the case and refer the reader to [2] for more details.

If $p(x) := p(x_1, x_2, x_3)$ is a ternary quartic form, its Hessian $H(x)$ is a 3×3 matrix whose entries are quadratic forms. In this case, we can no longer apply the biform theorem to the form $y^T H(x) y$. In fact, the matrix

$$C(x) = \begin{bmatrix} x_1^2 + 2x_2^2 & -x_1 x_2 & -x_1 x_3 \\ \\ -x_1 x_2 & x_2^2 + 2x_3^2 & -x_2 x_3 \\ \\ -x_1 x_3 & -x_2 x_3 & x_3^2 + 2x_1^2 \end{bmatrix}, \tag{3.13}$$

due to Choi [45] serves as an explicit example of a 3×3 matrix with quadratic form entries that is positive semidefinite but not an sos-matrix; i.e., the biquadratic form $y^T C(x) y$ is psd but not sos. However, the matrix $C(x)$ above is *not* a valid Hessian, i.e., it cannot be the matrix of the second derivatives of any polynomial. If this was the case, the third partial derivatives would commute. On the other hand, we have in particular

$$\frac{\partial C_{1,1}(x)}{\partial x_3} = 0 \neq -x_3 = \frac{\partial C_{1,3}(x)}{\partial x_1}.$$

A *biquadratic Hessian form* is a biquadratic form $y^T H(x) y$ where $H(x)$ is the Hessian of some quartic form. Biquadratic Hessian forms satisfy a special symmetry property. Let us call a biquadratic form $b(x; y)$ *symmetric* if it satisfies the symmetry relation $b(y; x) = b(x; y)$. It is an easy exercise to show that biquadratic Hessian forms satisfy $y^T H(x) y = x^T H(y) x$ and are therefore symmetric biquadratic forms. This symmetry property is a rather strong condition that is not satisfied e.g. by the Choi biquadratic form $y^T C(x) y$ in (3.13).

A simple dimension counting argument shows that the vector space of biquadratic forms, symmetric biquadratic forms, and biquadratic Hessian forms in variables $(x_1, x_2, x_3; y_1, y_2, y_3)$ respectively have dimensions 36, 21, and 15. Since the symmetry requirement drops the dimension of the space of biquadratic forms significantly, and since sos polynomials are known to generally cover much larger volume in the set of psd polynomials in presence of symmetries (see e.g. [28]), one may initially suspect (as we did) that the equivalence between psd and sos ternary Hessian biquadratic forms is a consequence of the symmetry property. Our next theorem shows that interestingly enough this is not the case.

Theorem 3.15. *There exist symmetric biquadratic forms in two sets of three variables that are positive semidefinite but not a sum of squares.*

Proof. We claim that the following biquadratic form has the required properties:

$$\begin{aligned} b(x_1,x_2,x_3;y_1,y_2,y_3) \;=\;& 4y_1y_2x_1^2+4x_1x_2y_1^2+9y_1y_3x_1^2+9x_1x_3y_1^2-10y_2y_3x_1^2 \\ & -10x_2x_3y_1^2+12y_1^2x_1^2+12y_2^2x_1^2+12x_2^2y_1^2+6y_3^2x_1^2 \\ & +6x_3^2y_1^2+23x_2^2y_1y_2+23y_2^2x_1x_2+13x_2^2y_1y_3+13x_1x_3y_2^2 \\ & +13y_2y_3x_2^2+13x_2x_3y_2^2+12x_2^2y_2^2+12x_2^2y_3^2+12y_2^2x_3^2 \\ & +5x_3^2y_1y_2+5y_3^2x_1x_2+12x_3^2y_3^2+3x_3^2y_1y_3+3y_3^2x_1x_3 \\ & +7x_3^2y_2y_3+7y_3^2x_2x_3+31y_1y_2x_1x_2-10x_1x_3y_1y_3 \\ & -11x_1x_3y_2y_3-11y_1y_3x_2x_3+5x_1x_2y_2y_3+5y_1y_2x_2x_3 \\ & +3x_1x_3y_1y_2+3y_1y_3x_1x_2-5x_2x_3y_2y_3. \end{aligned} \tag{3.14}$$

The fact that $b(x;y)=b(y;x)$ can readily be seen from the order in which we have written the monomials. The proof that $b(x;y)$ is psd but not sos is given in [2] and omitted from here. □

In view of the above theorem, it is rather remarkable that all positive semidefinite biquadratic Hessian forms in $(x_1,x_2,x_3;y_1,y_2,y_3)$ turn out to be sums of squares, i.e., that $\Sigma C_{3,4}=C_{3,4}$.

3.5.2 Proofs of Theorems 3.8 and 3.9: cases where $\tilde{\Sigma C}_{n,d}\subset\tilde{C}_{n,d}$, $\Sigma C_{n,d}\subset C_{n,d}$

The goal of this subsection is to establish that the cases presented in the previous subsection are the *only* cases where convexity and sos-convexity are equivalent. We will first give explicit examples of convex but not sos-convex polynomials/forms that are "minimal" jointly in the degree and dimension and then present an argument for all dimensions and degrees higher than those of the minimal cases.

Minimal convex but not sos-convex polynomials/forms

The minimal examples of convex but not sos-convex polynomials (resp. forms) turn out to belong to $\tilde{C}_{2,6}\setminus\tilde{\Sigma C}_{2,6}$ and $\tilde{C}_{3,4}\setminus\tilde{\Sigma C}_{3,4}$ (resp. $C_{3,6}\setminus\Sigma C_{3,6}$ and $C_{4,4}\setminus\Sigma C_{4,4}$). Recall from Remark 3.5.1 that we lack a general argument for going from convex but not sos-convex forms to polynomials or vice versa. Because of this, one would need to present four different polynomials in the sets mentioned above and prove that each polynomial is (i) convex and (ii) not sos-convex. This is a total of eight arguments to make which is quite cumbersome. However, as we will see in the proof of Theorem 3.16 and 3.17 below, we have been able to find examples that act "nicely" with respect to particular ways of dehomogenization. This will allow us to reduce the total number of claims we have to prove from eight to four.

The polynomials that we are about to present next have been found with the assistance of a computer and by employing some "tricks" with semidefinite programming similar to those presented in Appendix A.[9] In this process, we have made use of software packages YALMIP [98], SOSTOOLS [132], and the SDP solver SeDuMi [157], which we acknowledge here. To make the chapter relatively self-contained and to emphasize the fact that using *rational sum of squares certificates* one can make such computer assisted proofs fully formal, we present the proof of Theorem 3.16 below in the Appendix B. On the other hand, the proof of Theorem 3.17, which is very similar in style to the proof of Theorem 3.16, is largely omitted to save space. All of the proofs are available in electronic form and in their entirety at `http://aaa.lids.mit.edu/software`.

Theorem 3.16. *$\tilde{\Sigma C}_{2,6}$ is a proper subset of $\tilde{C}_{2,6}$. $\Sigma C_{3,6}$ is a proper subset of $C_{3,6}$.*

Proof. We claim that the form

$$\begin{aligned} f(x_1,x_2,x_3) \;=\; & 77x_1^6 - 155x_1^5x_2 + 445x_1^4x_2^2 + 76x_1^3x_2^3 + 556x_1^2x_2^4 + 68x_1x_2^5 \\ & +240x_2^6 - 9x_1^5x_3 - 1129x_1^3x_2^2x_3 + 62x_1^2x_2^3x_3 + 1206x_1x_2^4x_3 \\ & -343x_2^5x_3 + 363x_1^4x_3^2 + 773x_1^3x_2x_3^2 + 891x_1^2x_2^2x_3^2 - 869x_1x_2^3x_3^2 \\ & +1043x_2^4x_3^2 - 14x_1^3x_3^3 - 1108x_1^2x_2x_3^3 - 216x_1x_2^2x_3^3 - 839x_2^3x_3^3 \\ & +721x_1^2x_3^4 + 436x_1x_2x_3^4 + 378x_2^2x_3^4 + 48x_1x_3^5 - 97x_2x_3^5 + 89x_3^6 \end{aligned} \tag{3.15}$$

belongs to $C_{3,6} \setminus \Sigma C_{3,6}$, and the polynomial[10]

$$\tilde{f}(x_1,x_2) = f(x_1,x_2,1-\frac{1}{2}x_2) \tag{3.16}$$

[9]The approach of Appendix A, however, does not lead to examples that are minimal. But the idea is similar.

[10]The polynomial $f(x_1,x_2,1)$ turns out to be sos-convex, and therefore does not do the job. One can of course change coordinates, and then in the new coordinates perform the dehomogenization by setting $x_3 = 1$.

belongs to $\tilde{C}_{2,6} \setminus \Sigma\tilde{C}_{2,6}$. Note that since convexity and sos-convexity are both preserved under restrictions to affine subspaces (recall Remark 3.3.2), it suffices to show that the form f in (3.15) is convex and the polynomial $\tilde{f}$ in (3.16) is not sos-convex. Let $x := (x_1, x_2, x_2)^T$, $y := (y_1, y_2, y_3)^T$, $\tilde{x} := (x_1, x_2)^T$, $\tilde{y} := (y_1, y_2)^T$, and denote the Hessian of f and $\tilde{f}$ respectively by H_f and $H_{\tilde{f}}$. In Appendix B, we provide rational Gram matrices which prove that the form

$$(x_1^2 + x_2^2) \cdot y^T H_f(x) y \tag{3.17}$$

is sos. This, together with nonnegativity of $x_1^2 + x_2^2$ and continuity of $y^T H_f(x) y$, implies that $y^T H_f(x) y$ is psd. Therefore, f is convex. The proof that $\tilde{f}$ is not sos-convex proceeds by showing that $H_{\tilde{f}}$ is not an sos-matrix via a separation argument. In Appendix B, we present a separating hyperplane that leaves the appropriate sos cone on one side and the polynomial

$$\tilde{y}^T H_{\tilde{f}}(\tilde{x}) \tilde{y} \tag{3.18}$$

on the other. $\square$

Theorem 3.17. *$\Sigma\tilde{C}_{3,4}$ is a proper subset of $\tilde{C}_{3,4}$. $\Sigma C_{4,4}$ is a proper subset of $C_{4,4}$.*

Proof. We claim that the form

$$\begin{aligned} h(x_1, \dots, x_4) \;=\;& 1671x_1^4 - 4134x_1^3x_2 - 3332x_1^3x_3 + 5104x_1^2x_2^2 + 4989x_1^2x_2x_3 \\ & +3490x_1^2x_3^2 - 2203x_1x_2^3 - 3030x_1x_2^2x_3 - 3776x_1x_2x_3^2 \\ & -1522x_1x_3^3 + 1227x_2^4 - 595x_2^3x_3 + 1859x_2^2x_3^2 + 1146x_2x_3^3 \\ & +1195728x_4^4 - 1932x_1x_4^3 - 2296x_2x_4^3 - 3144x_3x_4^3 + 1465x_1^2x_4^2 \\ & -1376x_1^3x_4 - 263x_1x_2x_4^2 + 2790x_1^2x_2x_4 + 2121x_2^2x_4^2 + 979x_3^4 \\ & -292x_1x_2^2x_4 - 1224x_2^3x_4 + 2404x_1x_3x_4^2 + 2727x_2x_3x_4^2 \\ & -2852x_1x_3^2x_4 - 388x_2x_3^2x_4 - 1520x_3^3x_4 + 2943x_1^2x_3x_4 \\ & -5053x_1x_2x_3x_4 + 2552x_2^2x_3x_4 + 3512x_3^2x_4^2 \end{aligned} \tag{3.19}$$

belongs to $C_{4,4} \setminus \Sigma C_{4,4}$, and the polynomial

$$\tilde{h}(x_1, x_2, x_3) = h(x_1, x_2, x_3, 1) \tag{3.20}$$

belongs to $\tilde{C}_{3,4} \setminus \Sigma\tilde{C}_{3,4}$. Once again, it suffices to prove that h is convex and $\tilde{h}$ is not sos-convex. Let $x := (x_1, x_2, x_3, x_4)^T$, $y := (y_1, y_2, y_3, y_4)^T$, and denote the Hessian of h and $\tilde{h}$ respectively by H_h and $H_{\tilde{h}}$. The proof that h is convex is done by showing that the form

$$(x_2^2 + x_3^2 + x_4^2) \cdot y^T H_h(x) y \tag{3.21}$$

is sos.[11] The proof that $\tilde{h}$ is not sos-convex is done again by means of a separating hyperplane. □

Convex but not sos-convex polynomials/forms in all higher degrees and dimensions

Given a convex but not sos-convex polynomial (form) in n variables , it is very easy to argue that such a polynomial (form) must also exist in a larger number of variables. If $p(x_1, \ldots, x_n)$ is a form in $C_{n,d} \setminus \Sigma C_{n,d}$, then

$$\bar{p}(x_1, \ldots, x_{n+1}) = p(x_1, \ldots, x_n) + x_{n+1}^d$$

belongs to $C_{n+1,d} \setminus \Sigma C_{n+1,d}$. Convexity of $\bar{p}$ is obvious since it is a sum of convex functions. The fact that $\bar{p}$ is not sos-convex can also easily be seen from the block diagonal structure of the Hessian of $\bar{p}$: if the Hessian of $\bar{p}$ were to factor, it would imply that the Hessian of p should also factor. The argument for going from $\tilde{C}_{n,d} \setminus \Sigma\tilde{C}_{n,d}$ to $\tilde{C}_{n+1,d} \setminus \Sigma\tilde{C}_{n+1,d}$ is identical.

Unfortunately, an argument for increasing the degree of convex but not sos-convex forms seems to be significantly more difficult to obtain. In fact, we have been unable to come up with a natural operation that would produce a from in $C_{n,d+2} \setminus \Sigma C_{n,d+2}$ from a form in $C_{n,d} \setminus \Sigma C_{n,d}$. We will instead take a different route: we are going to present a general procedure for going from a form in $P_{n,d} \setminus \Sigma_{n,d}$ to a form in $C_{n,d+2} \setminus \Sigma C_{n,d+2}$. This will serve our purpose of constructing convex but not sos-convex forms in higher degrees and is perhaps also of independent interest in itself. For instance, it can be used to construct convex but not sos-convex forms that inherit structural properties (e.g. symmetry) of the known examples of psd but not sos forms. The procedure is constructive modulo the value of two positive constants (γ and α below) whose existence will be shown nonconstructively.

Although the proof of the general case is no different, we present this construction for the case $n = 3$. The reason is that it suffices for us to construct forms in $C_{3,d} \setminus \Sigma C_{3,d}$ for d even and ≥ 8. These forms together with the two forms in $C_{3,6} \setminus \Sigma C_{3,6}$ and $C_{4,4} \setminus \Sigma C_{4,4}$ presented in (3.15) and (3.19), and with the simple procedure for increasing the number of variables cover all the values of n and d for which convex but not sos-convex forms exist.

[11] The choice of multipliers in (3.17) and (3.21) is motivated by a result of Reznick in [137] explained in Appendix A.

For the remainder of this section, let $x := (x_1, x_2, x_3)^T$ and $y := (y_1, y_2, y_3)^T$.

Theorem 3.18. *Let $m := m(x)$ be a ternary form of degree d (with d necessarily even and ≥ 6) satisfying the following three requirements:*

R1: *m is positive definite.*

R2: *m is not a sum of squares.*

R3: *The Hessian H_m of m is positive definite at the point $(1, 0, 0)^T$.*

Let $g := g(x_2, x_3)$ be any bivariate form of degree $d + 2$ whose Hessian is positive definite.
Then, there exists a constant $\gamma > 0$, such that the form f of degree $d+2$ given by

$$f(x) = \int_0^{x_1} \int_0^{s} m(t, x_2, x_3) dt ds + \gamma g(x_2, x_3) \tag{3.22}$$

is convex but not sos-convex.

Before we prove this theorem, let us comment on how one can get examples of forms m and g that satisfy the requirements of the theorem. The choice of g is in fact very easy. We can e.g. take

$$g(x_2, x_3) = (x_2^2 + x_3^2)^{\frac{d+2}{2}},$$

which has a positive definite Hessian. As for the choice of m, essentially any psd but not sos ternary form can be turned into a form that satisfies requirements **R1**, **R2**, and **R3**. Indeed if the Hessian of such a form is positive definite at just one point, then that point can be taken to $(1, 0, 0)^T$ by a change of coordinates without changing the properties of being psd and not sos. If the form is not positive definite, then it can made so by adding a small enough multiple of a positive definite form to it. For concreteness, we construct in the next lemma a family of forms that together with the above theorem will give us convex but not sos-convex ternary forms of any degree ≥ 8.

Lemma 3.19. *For any even degree $d \geq 6$, there exists a constant $\alpha > 0$, such that the form*

$$m(x) = x_1^{d-6}(x_1^2 x_2^4 + x_1^4 x_2^2 - 3x_1^2 x_2^2 x_3^2 + x_3^6) + \alpha(x_1^2 + x_2^2 + x_3^2)^{\frac{d}{2}} \tag{3.23}$$

satisfies the requirements **R1**, **R2**, *and* **R3** *of Theorem 3.18.*

Proof. The form
$$x_1^2x_2^4 + x_1^4x_2^2 - 3x_1^2x_2^2x_3^2 + x_3^6$$
is the familiar Motzkin form in (3.1) that is psd but not sos [107]. For any even degree $d \geq 6$, the form

$$x_1^{d-6}(x_1^2x_2^4 + x_1^4x_2^2 - 3x_1^2x_2^2x_3^2 + x_3^6)$$

is a form of degree d that is clearly still psd and less obviously still not sos; see [138]. This together with the fact that $\Sigma_{n,d}$ is a closed cone implies existence of a small positive value of α for which the form m in (3.23) is positive definite but not a sum of squares, hence satisfying requirements **R1** and **R2**.

Our next claim is that for any positive value of α, the Hessian H_m of the form m in (3.23) satisfies

$$H_m(1,0,0) = \begin{bmatrix} c_1 & 0 & 0 \\ 0 & c_2 & 0 \\ 0 & 0 & c_3 \end{bmatrix} \tag{3.24}$$

for some positive constants c_1, c_2, c_3, therefore also passing requirement **R3**. To see the above equality, first note that since m is a form of degree d, its Hessian H_m will have entries that are forms of degree $d-2$. Therefore, the only monomials that can survive in this Hessian after setting x_2 and x_3 to zero are multiples of x_1^{d-2}. It is easy to see that an x_1^{d-2} monomial in an off-diagonal entry of H_m would lead to a monomial in m that is not even. On the other hand, the form m in (3.23) only has even monomials. This explains why the off-diagonal entries of the right hand side of (3.24) are zero. Finally, we note that for any positive value of α, the form m in (3.23) includes positive multiples of x_1^d, $x_1^{d-2}x_2^2$, and $x_1^{d-2}x_3^2$, which lead to positive multiples of x_1^{d-2} on the diagonal of H_m. Hence, c_1, c_2, and c_3 are positive. □

Next, we state a lemma that will be employed in the proof of Theorem 3.18.

Lemma 3.20. *Let m be a trivariate form satisfying the requirements* **R1** *and* **R3** *of Theorem 3.18. Let $H_{\hat{m}}$ denote the Hessian of the form $\int_0^{x_1}\int_0^s m(t, x_2, x_3)dtds$. Then, there exists a positive constant δ, such that*

$$y^T H_{\hat{m}}(x) y > 0$$

on the set

$$\mathcal{S} := \{(x,y) \mid ||x|| = 1, ||y|| = 1, \ (x_2^2 + x_3^2 < \delta \ or \ y_2^2 + y_3^2 < \delta)\}. \tag{3.25}$$

Proof. We observe that when $y_2^2 + y_3^2 = 0$, we have

$$y^T H_{\hat{m}}(x) y = y_1^2 m(x),$$

which by requirement **R1** is positive when $||x|| = ||y|| = 1$. By continuity of the form $y^T H_{\hat{m}}(x) y$, we conclude that there exists a small positive constant δ_y such that $y^T H_{\hat{m}}(x) y > 0$ on the set

$$\mathcal{S}_y := \{(x, y) \mid ||x|| = 1, ||y|| = 1,\ y_2^2 + y_3^2 < \delta_y\}.$$

Next, we leave it to the reader to check that

$$H_{\hat{m}}(1, 0, 0) = \frac{1}{d(d-1)} H_m(1, 0, 0).$$

Therefore, when $x_2^2 + x_3^2 = 0$, requirement **R3** implies that $y^T H_{\hat{m}}(x) y$ is positive when $||x|| = ||y|| = 1$. Appealing to continuity again, we conclude that there exists a small positive constant δ_x such that $y^T H_{\hat{m}}(x) y > 0$ on the set

$$\mathcal{S}_x := \{(x, y) \mid ||x|| = 1, ||y|| = 1,\ x_2^2 + x_3^2 < \delta_x\}.$$

If we now take $\delta = \min\{\delta_y, \delta_x\}$, the lemma is established. $\square$

We are now ready to prove Theorem 3.18.

Proof of Theorem 3.18. We first prove that the form f in (3.22) is not sos-convex. By Lemma 3.7, if f was sos-convex, then all diagonal elements of its Hessian would have to be sos polynomials. On the other hand, we have from (3.22) that

$$\frac{\partial f(x)}{\partial x_1 \partial x_1} = m(x),$$

which by requirement **R2** is not sos. Therefore f is not sos-convex.

It remains to show that there exists a positive value of γ for which f becomes convex. Let us denote the Hessians of f, $\int_0^{x_1} \int_0^s m(t, x_2, x_3) dt ds$, and g, by H_f, $H_{\hat{m}}$, and H_g respectively. So, we have

$$H_f(x) = H_{\hat{m}}(x) + \gamma H_g(x_2, x_3).$$

(Here, H_g is a 3×3 matrix whose first row and column are zeros.) Convexity of f is of course equivalent to nonnegativity of the form $y^T H_f(x) y$. Since this form is bi-homogeneous in x and y, it is nonnegative if and only if $y^T H_f(x) y \geq 0$ on the bi-sphere

$$\mathcal{B} := \{(x, y) \mid ||x|| = 1, ||y|| = 1\}.$$

Let us decompose the bi-sphere as

$$\mathcal{B} = \mathcal{S} \cup \bar{\mathcal{S}},$$

where $\mathcal{S}$ is defined in (3.25) and

$$\bar{\mathcal{S}} := \{(x, y) \mid ||x|| = 1, ||y|| = 1, x_2^2 + x_3^2 \geq \delta, y_2^2 + y_3^2 \geq \delta\}.$$

Lemma 3.20 together with positive definiteness of H_g imply that $y^T H_f(x) y$ is positive on $\mathcal{S}$. As for the set $\bar{\mathcal{S}}$, let

$$\beta_1 = \min_{x,y,\in\bar{\mathcal{S}}} y^T H_{\hat{m}}(x) y,$$

and

$$\beta_2 = \min_{x,y,\in\bar{\mathcal{S}}} y^T H_g(x_2, x_3) y.$$

By the assumption of positive definiteness of H_g, we have $\beta_2 > 0$. If we now let

$$\gamma > \frac{|\beta_1|}{\beta_2},$$

then

$$\min_{x,y,\in\bar{\mathcal{S}}} y^T H_f(x) y > \beta_1 + \frac{|\beta_1|}{\beta_2} \beta_2 \geq 0.$$

Hence $y^T H_f(x) y$ is nonnegative (in fact positive) everywhere on $\mathcal{B}$ and the proof is completed. □

Finally, we provide an argument for existence of bivariate polynomials of degree $8, 10, 12, \ldots$ that are convex but not sos-convex.

Corollary 3.21. *Consider the form f in (3.22) constructed as described in Theorem 3.18. Let*

$$\tilde{f}(x_1, x_2) = f(x_1, x_2, 1).$$

Then, $\tilde{f}$ is convex but not sos-convex.

Proof. The polynomial $\tilde{f}$ is convex because it is the restriction of a convex function. It is not difficult to see that

$$\frac{\partial \tilde{f}(x_1, x_2)}{\partial x_1 \partial x_1} = m(x_1, x_2, 1),$$

which is not sos. Therefore from Lemma 3.7 $\tilde{f}$ is not sos-convex. □

Corollary 3.21 together with the two polynomials in $\tilde{C}_{2,6} \backslash \Sigma\tilde{C}_{2,6}$ and $\tilde{C}_{3,4} \backslash \Sigma\tilde{C}_{3,4}$ presented in (3.16) and (3.20), and with the simple procedure for increasing the number of variables described at the beginning of Subsection 3.5.2 cover all the values of n and d for which convex but not sos-convex polynomials exist.

3.6 Concluding remarks and an open problem

convex=sos-convex?

Polynomials

n,d	2	4	≥6
1	yes	yes	yes
2	yes	yes	no
3	yes	no	no
≥4	yes	no	no

Forms

n,d	2	4	≥6
1	yes	yes	yes
2	yes	yes	yes
3	yes	yes	no
≥4	yes	no	no

Figure 3.1. The tables answer whether every convex polynomial (form) in n variables and of degree d is sos-convex.

A summary of the results of this chapter is given in Figure 3.1. To conclude, we would like to point out some similarities between nonnegativity and convexity that deserve attention: (i) both nonnegativity and convexity are properties that only hold for even degree polynomials, (ii) for quadratic forms, nonnegativity is in fact equivalent to convexity, (iii) both notions are NP-hard to check exactly for degree 4 and larger, and most strikingly (iv) nonnegativity is equivalent to sum of squares *exactly* in dimensions and degrees where convexity is equivalent to sos-convexity. It is unclear to us whether there can be a deeper and more unifying reason explaining these observations, in particular, the last one which was the main result of this chapter.

Another intriguing question is to investigate whether one can give a direct argument proving the fact that $\Sigma\tilde{C}_{n,d} = \tilde{C}_{n,d}$ if and only if $\Sigma C_{n+1,d} = C_{n+1,d}$. This would eliminate the need for studying polynomials and forms separately, and in particular would provide a short proof of the result $\Sigma_{3,4} = C_{3,4}$ given in [2].

Finally, an open problem related to the work in this chapter is to *find an explicit example of a convex form that is not a sum of squares*. Blekherman [26] has shown via volume arguments that for degree $d \geq 4$ and asymptotically for large n such forms must exist, although no examples are known. In particular, it would interesting to determine the smallest value of n for which such a form exists. We know from Lemma 3.6 that a convex form that is not sos must necessarily be not sos-convex. Although our several constructions of convex but not sos-convex polynomials pass this necessary condition, the polynomials themselves are all sos. The question is particularly interesting from an optimization viewpoint because it implies that the well-known sum of squares relaxation for minimizing polynomials [155], [124] may not be exact even for the easy case of minimizing convex polynomials.

3.7 Appendix A: How the first convex but not sos-convex polynomial was found

In this appendix, we explain how the polynomial in (3.11) was found by solving a carefully designed sos-program[12]. The simple methodology described here allows one to search over a restricted family of nonnegative polynomials that are not sums of squares. The procedure can potentially be useful in many different settings and this is our main motivation for presenting this appendix.

Our goal is to find a polynomial $p := p(x)$ whose Hessian $H := H(x)$ satisfies:

$$y^T H(x) y \quad \text{psd but not sos.} \tag{3.26}$$

Unfortunately, a constraint of type (3.26) that requires a polynomial to be psd but *not* sos is a non-convex constraint and cannot be easily handled with sos-programming. This is easy to see from a geometric viewpoint. The feasible set of an sos-program, being a semidefinite program, is always a convex set. On the other hand, for a fixed degree and dimension, the set of psd polynomials that are not sos is non-convex. Nevertheless, we describe a technique that allows one to search over a *convex subset* of the set of psd but not sos polynomials using sos-programming. Our strategy can simply be described as follows: (i) Impose the constraint that the polynomial should *not* be sos by using a separating hyperplane (dual functional) for the sos cone. (ii) Impose the constraint that the polynomial should be psd by requiring that the polynomial times a nonnegative multiplier is sos.

By definition, the dual cone $\Sigma_{n,d}^*$ of the sum of squares cone $\Sigma_{n,d}$ is the set of all linear functionals μ that take nonnegative values on it, i.e,

$$\Sigma_{n,d}^* := \{\mu \in \mathcal{H}_{n,d}^*, \quad \langle \mu, p \rangle \geq 0 \ \ \forall p \in \Sigma_{n,d}\}.$$

Here, the dual space $\mathcal{H}_{n,d}^*$ denotes the space of all linear functionals on the space $\mathcal{H}_{n,d}$ of forms in n variables and degree d, and $\langle ., . \rangle$ represents the pairing between elements of the primal and the dual space. If a form is not sos, we can find a dual functional $\mu \in \Sigma_{n,d}^*$ that separates it from the closed convex cone $\Sigma_{n,d}$. The basic idea behind this is the well known separating hyperplane theorem in convex analysis; see e.g. [38, 142].

As for step (ii) of our strategy above, our approach for guaranteeing that of a form g is nonnegative will be to require $g(x) \cdot (\sum_i x_i^2)^r$ be sos for some integer $r \geq 1$. Our choice of the multiplier $(\sum_i x_i^2)^r$ as opposed to any other psd multiplier is motivated by a result of Reznick [137] on Hilbert's 17th problem. The 17th

[12] The term "sos-program" is usually used to refer to semidefinite programs that have sum of squares constraints.

problem, which was answered in the affirmative by Artin [19], asks whether every psd form must be a sum of squares of rational functions. The affirmative answer to this question implies that if a form g is psd, then there must exist an sos form s, such that $g \cdot s$ is sos. Reznick showed in [137] that if g is positive definite, one can always take $s(x) = (\sum_i x_i^2)^r$, for sufficiently large r. For all polynomials that we needed prove psd in this chapter, taking $r = 1$ has been good enough.

For our particular purpose of finding a convex but not sos-convex polynomial, we apply the strategy outlined above to make the first diagonal element of the Hessian psd but not sos (recall Lemma 3.7). More concretely, the polynomial in (3.11) was derived from a feasible solution to the following sos-program:

– Parameterize $p \in \mathcal{H}_{3,8}$ and compute its Hessian $H = \frac{\partial^2 p}{\partial x^2}$.

– Impose the constraints

$$(x_1^2 + x_2^2 + x_3^2) \cdot y^T H(x) y \quad \text{sos}, \tag{3.27}$$

$$\langle \mu, H_{1,1} \rangle = -1 \tag{3.28}$$

$$\text{(for some dual functional } \mu \in \Sigma_{3,6}^*\text{)}.$$

The decision variables of this sos-program are the coefficients of the polynomial p that also appear in the entries of the Hessian matrix H. (The polynomial $H_{1,1}$ in (3.28) denotes the first diagonal element of H.) The dual functional μ must be fixed a priori as explained in the sequel. Note that all the constraints are linear in the decision variables and indeed the feasible set described by these constraints is a convex set. Moreover, the reader should be convinced by now that if the above sos-program is feasible, then the solution p is a convex polynomial that is not sos-convex.

The reason why we chose to parameterize p as a form in $\mathcal{H}_{3,8}$ is that a minimal case where a diagonal element of the Hessian (which has 2 fewer degree) can be psd but not sos is among the forms in $\mathcal{H}_{3,6}$. The role of the dual functional $\mu \in \Sigma_{3,6}^*$ in (3.28) is to separate the polynomial $H_{1,1}$ from $\Sigma_{3,6}$. Once an ordering on the monomials of $H_{1,1}$ is fixed, this constraint can be imposed numerically as

$$\langle \mu, H_{1,1} \rangle = b^T \vec{H}_{1,1} = -1, \tag{3.29}$$

where $\vec{H}_{1,1}$ denotes the vector of coefficients of the polynomial $H_{1,1}$ and $b \in \mathbb{R}^{28}$ represents our separating hyperplane, which must be computed prior to solving the above sos-program.

There are several ways to obtain a separating hyperplane for $\Sigma_{3,6}$. Our approach was to find a hyperplane that separates the Motzkin form M in (3.1) from

$\Sigma_{3,6}$. This can be done in at least a couple of different ways. For example, we can formulate a semidefinite program that requires the Motzkin form to be sos. This program is clearly infeasible. Any feasible solution to its dual semidefinite program will give us the desired separating hyperplane. Alternatively, we can set up an sos-program that finds the Euclidean projection $M^p := M^p(x)$ of the Motzkin form M onto the cone $\Sigma_{3,6}$. Since the projection is done onto a convex set, the hyperplane tangent to $\Sigma_{3,6}$ at M^p will be supporting $\Sigma_{3,6}$, and can serve as our separating hyperplane.

To conclude, we remark that in contrast to previous techniques of constructing examples of psd but not sos polynomials that are usually based on some obstructions associated with the number of zeros of polynomials (see e.g. [138]), our approach has the advantage that the resulting polynomials are positive definite. Furthermore, additional linear or semidefinite constraints can easily be incorporated in the search process to impose e.g. various symmetry or sparsity patterns on the polynomial of interest.

3.8 Appendix B: Certificates complementing the proof of Theorem 3.16

Let $x := (x_1, x_2, x_2)^T$, $y := (y_1, y_2, y_3)^T$, $\tilde{x} := (x_1, x_2)^T$, $\tilde{y} := (y_1, y_2)^T$, and let $f, \tilde{f}, H_f$, and $H_{\tilde{f}}$ be as in the proof of Theorem 3.16. This appendix proves that the form $(x_1^2 + x_2^2) \cdot y^T H_f(x) y$ in (3.17) is sos and that the polynomial $\tilde{y}^T H_{\tilde{f}}(\tilde{x})\tilde{y}$ in (3.18) is not sos, hence proving respectively that f is convex and $\tilde{f}$ is not sos-convex.

A rational sos decomposition of $(x_1^2 + x_2^2) \cdot y^T H_f(x) y$, which is a form in 6 variables of degree 8, is as follows:

$$(x_1^2 + x_2^2) \cdot y^T H_f(x) y = \frac{1}{84} z^T Q z,$$

where z is the vector of monomials

$$\begin{aligned} z = \; & [x_2x_3^2y_3, x_2x_3^2y_2, x_2x_3^2y_1, x_2^2x_3y_3, x_2^2x_3y_2, x_2^2x_3y_1, x_2^3y_3, x_2^3y_2, x_2^3y_1, \\ & x_1x_3^2y_3, x_1x_3^2y_2, x_1x_3^2y_1, x_1x_2x_3y_3, x_1x_2x_3y_2, x_1x_2x_3y_1, x_1x_2^2y_3, x_1x_2^2y_2, x_1x_2^2y_1, \\ & x_1^2x_3y_3, x_1^2x_3y_2, x_1^2x_3y_1, x_1^2x_2y_3, x_1^2x_2y_2, x_1^2x_2y_1, x_1^3y_3, x_1^3y_2, x_1^3y_1]^T, \end{aligned}$$

and Q is the 27×27 positive definite matrix[13] presented on the next page

$$Q = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix},$$

[13]Whenever we state a matrix is positive definite, this claim is backed up by a rational LDL^T factorization of the matrix that the reader can find online at `http://aaa.lids.mit.edu/software`.

$$Q_1 = \begin{bmatrix}
224280 & -40740 & 20160 & -81480 & 139692 & 93576 & -50540 & -27804 & -48384 & 0 & -29400 & -32172 & 21252 & 103404 \\
-40740 & 63504 & 36624 & 114324 & -211428 & -8316 & -67704 & 15372 & -47376 & 29400 & 0 & 16632 & 75936 & 15540 \\
20160 & 36624 & 121128 & 52920 & -27972 & -93072 & -42252 & -77196 & -58380 & 32172 & -16632 & 0 & 214284 & -57960 \\
-81480 & 114324 & 52920 & 482104 & -538776 & 36204 & -211428 & 362880 & -70644 & 19068 & 13524 & -588 & 179564 & 20258 \\
139692 & -211428 & -27972 & -538776 & 1020600 & -94416 & 338016 & -288120 & 188748 & -46368 & -33684 & -46620 & -113638 & -119112 \\
93576 & -8316 & -93072 & 36204 & -94416 & 266448 & -75348 & 216468 & 5208 & 3360 & -33432 & -31080 & -221606 & 254534 \\
-50540 & -67704 & -42252 & -211428 & 338016 & -75348 & 175224 & -144060 & 101304 & -20692 & 7826 & -4298 & -77280 & -108192 \\
-27804 & 15372 & -77196 & 362880 & -288120 & 216468 & -144060 & 604800 & 28560 & -35350 & -840 & -35434 & -132804 & 134736 \\
-48384 & -47376 & -58380 & -70644 & 188748 & 5208 & 101304 & 28560 & 93408 & -21098 & 2786 & -11088 & -104496 & -22680 \\
0 & 29400 & 32172 & 19068 & -46368 & 3360 & -20692 & -35350 & -21098 & 224280 & -40740 & 20160 & 35028 & 89964 \\
-29400 & 0 & -16632 & 13524 & -33684 & -33432 & 7826 & -840 & 2786 & -40740 & 63504 & 36624 & 51828 & -196476 \\
-32172 & 16632 & 0 & -588 & -46620 & -31080 & -4298 & -35434 & -11088 & 20160 & 36624 & 121128 & 29148 & -9408 \\
21252 & 75936 & 214284 & 179564 & -113638 & -221606 & -77280 & -132804 & -104496 & 35028 & 51828 & 29148 & 782976 & -463344 \\
103404 & 15540 & -57960 & 20258 & -119112 & 254534 & -108192 & 134736 & -22680 & 89964 & -196476 & -9408 & -463344 & 1167624 \\
267456 & -48132 & 27552 & -49742 & 78470 & 124236 & -100464 & 61404 & -90384 & 55524 & -50064 & -145908 & 41016 & -15456 \\
60872 & -25690 & -142478 & 22848 & -113820 & 259980 & -72996 & 237972 & 20412 & -95580 & -47964 & -27780 & -438732 & 514500 \\
37730 & -99036 & -10150 & -83160 & 473088 & 34188 & 167244 & 57120 & 159264 & 10752 & -93048 & -183540 & 230832 & -49980 \\
-119210 & 9170 & 81648 & 244356 & -41664 & -194124 & -9996 & 214368 & 19152 & -89184 & 2940 & -48480 & 204708 & -85344 \\
-116508 & 81564 & 26124 & 155832 & -308280 & -78180 & -74088 & 14616 & -49644 & 40320 & 87108 & 225456 & 135744 & 8568 \\
30660 & -14952 & 11844 & -21420 & 62604 & 14364 & 13608 & 1176 & 5124 & 59388 & -18144 & -99624 & 31332 & -178248 \\
35700 & 11340 & 52836 & -70788 & 86184 & 9396 & 12264 & -108024 & -11256 & 259056 & -86520 & -3528 & -19334 & 142128 \\
102156 & 2856 & 64536 & 22176 & -4200 & 77532 & -70896 & 54348 & -49616 & 72744 & -78876 & -144998 & 29316 & 23856 \\
-86100 & 47148 & 71820 & 230916 & -223692 & -131628 & -72156 & 59640 & -31416 & -75096 & -39396 & -44520 & 158508 & 308196 \\
95364 & -504 & -8412 & -23100 & 28140 & 81648 & -26768 & -25200 & -13944 & -51002 & -39228 & 71232 & 130298 & 298956 \\
11256 & 5208 & 32158 & -33264 & 45444 & 3122 & 6888 & -34440 & -5628 & 61320 & -19152 & 8988 & 18060 & -19467 \\
0 & -1344 & -3696 & -34692 & 33768 & 5964 & 9492 & -20244 & 5208 & -30072 & -9912 & 58884 & -50883 & 151956 \\
-51422 & 49056 & 32592 & 160370 & -229068 & -36792 & -68796 & 57708 & -39564 & 55944 & 31164 & -8008 & 141876 & -126483
\end{bmatrix},$$

$$Q_2 = \begin{bmatrix}
267456 & 60872 & 37730 & -119210 & -116508 & 30660 & 35700 & 102156 & -86100 & 95364 & 11256 & 0 & -51422 \\
-48132 & -25690 & -99036 & 9170 & 81564 & -14952 & 11340 & 2856 & 47148 & -504 & 5208 & -1344 & 49056 \\
27552 & -142478 & -10150 & 81648 & 26124 & 11844 & 52836 & 64536 & 71820 & -8412 & 32158 & -3696 & 32592 \\
-49742 & 22848 & -83160 & 244356 & 155832 & -21420 & -70788 & 22176 & 230916 & -23100 & -33264 & -34692 & 160370 \\
78470 & -113820 & 473088 & -41664 & -308280 & 62604 & 86184 & -4200 & -223692 & 28140 & 45444 & 33768 & -229068 \\
124236 & 259980 & 34188 & -194124 & -78180 & 14364 & 9396 & 77532 & -131628 & 81648 & 3122 & 5964 & -36792 \\
-100464 & -72996 & 167244 & -9996 & -74088 & 13608 & 12264 & -70896 & -72156 & -26768 & 6888 & 9492 & -68796 \\
61404 & 237972 & 57120 & 214368 & 14616 & 1176 & -108024 & 54348 & 59640 & -25200 & -34440 & -20244 & 57708 \\
-90384 & 20412 & 159264 & 19152 & -49644 & 5124 & -11256 & -49616 & -31416 & -13944 & -5628 & 5208 & -39564 \\
55524 & -95580 & 10752 & -89184 & 40320 & 59388 & 259056 & 72744 & -75096 & -51002 & 61320 & -30072 & 55944 \\
-50064 & -47964 & -93048 & 2940 & 87108 & -18144 & -86520 & -78876 & -39396 & -39228 & -19152 & -9912 & 31164 \\
-145908 & -27780 & -183540 & -48480 & 225456 & -99624 & -3528 & -144998 & -44520 & 71232 & 8988 & 58884 & -8008 \\
41016 & -438732 & 230832 & 204708 & 135744 & 31332 & -19334 & 29316 & 158508 & 130298 & 18060 & -50883 & 141876 \\
-15456 & 514500 & -49980 & -85344 & 8568 & -178248 & 142128 & 23856 & 308196 & 298956 & -19467 & 151956 & -126483 \\
610584 & 21840 & 127932 & -65184 & -323834 & 195636 & 90972 & 339794 & -100716 & -96012 & 24864 & -114219 & 36876 \\
21840 & 466704 & -110628 & -106820 & -54012 & -90636 & -111790 & -14952 & 63672 & 107856 & -67788 & 61404 & -88284 \\
127932 & -110628 & 1045968 & 142632 & -410592 & 171024 & 86268 & 176820 & 96516 & 199752 & 13524 & -70784 & -42756 \\
-65184 & -106820 & 142632 & 569856 & 21518 & -30156 & -159264 & -23016 & 410004 & -71484 & -62076 & -13860 & 74032 \\
-323834 & -54012 & -410592 & 21518 & 604128 & -229992 & -75516 & -297276 & 182385 & 75684 & -3528 & 94500 & 138432 \\
195636 & -90636 & 171024 & -30156 & -229992 & 169512 & 104748 & 187341 & -136332 & -145719 & 35364 & -94836 & 24612 \\
90972 & -111790 & 86268 & -159264 & -75516 & 104748 & 381920 & 147168 & -182595 & -36876 & 105504 & -24612 & -7560 \\
339794 & -14952 & 176820 & -23016 & -297276 & 187341 & 147168 & 346248 & -59304 & -137928 & 64932 & -90888 & 28392 \\
-100716 & 63672 & 96516 & 410004 & 182385 & -136332 & -182595 & -59304 & 776776 & 48972 & -98784 & 19152 & 180852 \\
-96012 & 107856 & 199752 & -71484 & 75684 & -145719 & -36876 & -137928 & 48972 & 494536 & -28392 & 118188 & -130200 \\
24864 & -67788 & 13524 & -62076 & -3528 & 35364 & 105504 & 64932 & -98784 & -28392 & 60984 & 0 & -3780 \\
-114219 & 61404 & -70784 & -13860 & 94500 & -94836 & -24612 & -90888 & 19152 & 118188 & 0 & 74760 & -65100 \\
36876 & -88284 & -42756 & 74032 & 138432 & 24612 & -7560 & 28392 & 180852 & -130200 & -3780 & -65100 & 194040
\end{bmatrix}.$$

Next, we prove that the polynomial $\tilde{y}^T H_{\tilde{f}}(\tilde{x})\tilde{y}$ in (3.18) is not sos. Let us first present this polynomial and give it a name:

$$\begin{aligned}
t(\tilde{x},\tilde{y}) := \tilde{y}^T H_{\tilde{f}}(\tilde{x})\tilde{y} \;=\;& 294x_1x_2y_2^2 - 6995x_2^4y_1y_2 - 10200x_1y_1y_2 - 4356x_1^2x_2y_1^2 \\
& -2904x_1^3y_1y_2 - 11475x_1x_2^2y_1^2 + 13680x_2^3y_1y_2 + 4764x_1x_2y_1^2 \\
& +4764x_1^2y_1y_2 + 6429x_1^2x_2^2y_1^2 + 294x_2^2y_1y_2 - 13990x_1x_2^3y_2^2 \\
& -12123x_1^2x_2y_2^2 - 3872x_2y_1y_2 + \tfrac{2143}{2}x_1^4y_2^2 + 20520x_1x_2^2y_2^2 \\
& +29076x_1x_2y_1y_2 - 24246x_1x_2^2y_1y_2 + 14901x_1x_2^3y_1y_2 \\
& +15039x_1^2x_2^2y_1y_2 + 8572x_1^3x_2y_1y_2 + \tfrac{44703}{4}x_1^2x_2^2y_2^2 + 1442y_1^2 \\
& -12360x_2y_2^2 - 5100x_2y_1^2 + \tfrac{147513}{4}x_2^2y_2^2 + 7269x_2^2y_1^2 \\
& +\tfrac{772965}{32}x_2^4y_2^2 + \tfrac{14901}{8}x_2^4y_1^2 - 1936x_1y_2^2 - 84x_1y_1^2 + \tfrac{3817}{2}y_2^2 \\
& +7269x_1^2y_2^2 + 4356x_1^2y_1^2 - 3825x_1^3y_2^2 - 180x_1^3y_1^2 + 632y_1y_2 \\
& +2310x_1^4y_1^2 + 5013x_1x_2^3y_1^2 - 22950x_1^2x_2y_1y_2 - 45025x_2^3y_2^2 \\
& -1505x_1^4y_1y_2 - 4041x_2^3y_1^2 - 3010x_1^3x_2y_1^2 + 5013x_1^3x_2y_2^2.
\end{aligned}$$

Note that t is a polynomial in 4 variables of degree 6 that is quadratic in $\tilde{y}$. Let us denote the cone of sos polynomials in 4 variables $(\tilde{x},\tilde{y})$ that have degree 6 and are quadratic in $\tilde{y}$ by $\hat{\Sigma}_{4,6}$, and its dual cone by $\hat{\Sigma}^*_{4,6}$. Our proof will simply proceed by presenting a dual functional $\xi \in \hat{\Sigma}^*_{4,6}$ that takes a negative value on the polynomial t. We fix the following ordering of monomials in what follows:

$$\begin{aligned}
v = \; & [y_2^2, y_1y_2, y_1^2, x_2y_2^2, x_2y_1y_2, x_2y_1^2, x_2^2y_2^2, x_2^2y_1y_2, x_2^2y_1^2, x_2^3y_2^2, x_2^3y_1y_2, x_2^3y_1^2, x_2^4y_2^2, \\
& x_2^4y_1y_2, x_2^4y_1^2, x_1y_2^2, x_1y_1y_2, x_1y_1^2, x_1x_2y_2^2, x_1x_2y_1y_2, x_1x_2y_1^2, x_1x_2^2y_2^2, x_1x_2^2y_1y_2, \\
& x_1x_2^2y_1^2, x_1x_2^3y_2^2, x_1x_2^3y_1y_2, x_1x_2^3y_1^2, x_1^2y_2^2, x_1^2y_1y_2, x_1^2y_1^2, x_1^2x_2y_2^2, x_1^2x_2y_1y_2, x_1^2x_2y_1^2, \\
& x_1^2x_2^2y_2^2, x_1^2x_2^2y_1y_2, x_1^2x_2^2y_1^2, x_1^3y_2^2, x_1^3y_1y_2, x_1^3y_1^2, x_1^3x_2y_2^2, x_1^3x_2y_1y_2, x_1^3x_2y_1^2, x_1^4y_2^2, \\
& x_1^4y_1y_2, x_1^4y_1^2]^T.
\end{aligned} \tag{3.30}$$

Let $\vec{t}$ represent the vector of coefficients of t ordered according to the list of monomials above; i.e., $t = \vec{t}^T v$. Using the same ordering, we can represent our dual functional ξ with the vector

$$c = \begin{array}{l}[19338, -2485, 17155, 6219, -4461, 11202, 4290, -5745, 13748, 3304, -5404, \\ 13227, 3594, -4776, 19284, 2060, 3506, 5116, 366, -2698, 6231, -487, -2324, \\ 4607, 369, -3657, 3534, 6122, 659, 7057, 1646, 1238, 1752, 2797, -940, 4608, \\ -200, 1577, -2030, -513, -3747, 2541, 15261, 220, 7834]^T.\end{array}$$

We have

$$\langle \xi, t \rangle = c^T \vec{t} = -\frac{364547}{16} < 0.$$

On the other hand, we claim that $\xi \in \hat{\Sigma}^*_{4,6}$; i.e., for any form $w \in \hat{\Sigma}_{4,6}$, we should have

$$\langle \xi, w \rangle = c^T \vec{w} \geq 0, \tag{3.31}$$

where $\vec{w}$ here denotes the coefficients of w listed according to the ordering in (3.30). Indeed, if w is sos, then it can be written in the form

$$w(x) = \tilde{z}^T \tilde{Q} \tilde{z} = \mathrm{Tr}\ \tilde{Q} \cdot \tilde{z}\tilde{z}^T,$$

for some symmetric positive semidefinite matrix $\tilde{Q}$, and a vector of monomials

$$\tilde{z} = [y_2, y_1, x_2y_2, x_2y_1, x_1y_2, x_1y_1, x_2^2y_2, x_2^2y_1, x_1x_2y_2, x_1x_2y_1, x_1^2y_2, x_1^2y_1]^T.$$

It is not difficult to see that

$$c^T \vec{w} = \mathrm{Tr}\, \tilde{Q} \cdot (\tilde{z}\tilde{z}^T)|_c, \tag{3.32}$$

where by $(\tilde{z}\tilde{z}^T)|_c$ we mean a matrix where each monomial in $\tilde{z}\tilde{z}^T$ is replaced with the corresponding element of the vector c. This yields the matrix

$$(\tilde{z}\tilde{z}^T)|_c = \begin{bmatrix} 19338 & -2485 & 6219 & -4461 & 2060 & 3506 & 4290 & -5745 & 366 & -2698 & 6122 & 659 \\ -2485 & 17155 & -4461 & 11202 & 3506 & 5116 & -5745 & 13748 & -2698 & 6231 & 659 & 7057 \\ 6219 & -4461 & 4290 & -5745 & 366 & -2698 & 3304 & -5404 & -487 & -2324 & 1646 & 1238 \\ -4461 & 11202 & -5745 & 13748 & -2698 & 6231 & -5404 & 13227 & -2324 & 4607 & 1238 & 1752 \\ 2060 & 3506 & 366 & -2698 & 6122 & 659 & -487 & -2324 & 1646 & 1238 & -200 & 1577 \\ 3506 & 5116 & -2698 & 6231 & 659 & 7057 & -2324 & 4607 & 1238 & 1752 & 1577 & -2030 \\ 4290 & -5745 & 3304 & -5404 & -487 & -2324 & 3594 & -4776 & 369 & -3657 & 2797 & -940 \\ -5745 & 13748 & -5404 & 13227 & -2324 & 4607 & -4776 & 19284 & -3657 & 3534 & -940 & 4608 \\ 366 & -2698 & -487 & -2324 & 1646 & 1238 & 369 & -3657 & 2797 & -940 & -513 & -3747 \\ -2698 & 6231 & -2324 & 4607 & 1238 & 1752 & -3657 & 3534 & -940 & 4608 & -3747 & 2541 \\ 6122 & 659 & 1646 & 1238 & -200 & 1577 & 2797 & -940 & -513 & -3747 & 15261 & 220 \\ 659 & 7057 & 1238 & 1752 & 1577 & -2030 & -940 & 4608 & -3747 & 2541 & 220 & 7834 \end{bmatrix},$$

which is positive definite. Therefore, equation (3.32) along with the fact that $\tilde{Q}$ is positive semidefinite implies that (3.31) holds. This completes the proof.

Part II:
Lyapunov Analysis and Computation

Chapter 4

Lyapunov Analysis of Polynomial Differential Equations

In the last two chapters of this thesis, our focus will turn to Lyapunov analysis of dynamical systems. The current chapter presents new results on Lyapunov analysis of polynomial vector fields. The content here is based on the works in [10] and [4], as well as some more recent results.

4.1 Introduction

We will be concerned for the most part of this chapter with a continuous time dynamical system

$$\dot{x} = f(x), \tag{4.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}^n$ is a polynomial and has an equilibrium at the origin, i.e., $f(0) = 0$. Arguably, the class of polynomial differential equations are among the most widely encountered in engineering and sciences. For stability analysis of these systems, it is most common (and quite natural) to search for Lyapunov functions that are polynomials themselves. When such a candidate Lyapunov function is used, then conditions of Lyapunov's theorem reduce to a set of polynomial inequalities. For instance, if establishing global asymptotic stability of the origin is desired, one would require a radially unbounded polynomial Lyapunov candidate $V(x) : \mathbb{R}^n \to \mathbb{R}$ to vanish at the origin and satisfy

$$V(x) \;>\; 0 \quad \forall x \neq 0 \tag{4.2}$$

$$\dot{V}(x) = \langle \nabla V(x), f(x) \rangle \;<\; 0 \quad \forall x \neq 0. \tag{4.3}$$

Here, $\dot{V}$ denotes the time derivative of V along the trajectories of (4.1), $\nabla V(x)$ is the gradient vector of V, and $\langle ., . \rangle$ is the standard inner product in $\mathbb{R}^n$. In some other variants of the analysis problem, e.g. if LaSalle's invariance principle is to be used, or if the goal is to prove boundedness of trajectories of (4.1), then the inequality in (4.3) is replaced with

$$\dot{V}(x) \leq 0 \quad \forall x. \tag{4.4}$$

In any case, the problem arising from this analysis approach is that even though polynomials of a given degree are finitely parameterized, the computational problem of searching for a polynomial V satisfying inequalities of the type (4.2), (4.3), (4.4) is intractable. An approach pioneered in [118] and widely popular by now is to replace the positivity (or nonnegativity) conditions by the requirement of the existence of a sum of squares (sos) decomposition:

$$V \quad \text{sos} \tag{4.5}$$

$$-\dot{V} = -\langle \nabla V, f \rangle \quad \text{sos}. \tag{4.6}$$

As we saw in the previous chapter, sum of squares decomposition is a sufficient condition for polynomial nonnegativity that can be efficiently checked with semidefinite programming. For a fixed degree of a polynomial Lyapunov candidate V, the search for the coefficients of V subject to the constraints (4.5) and (4.6) is a semidefinite program (SDP). We call a Lyapunov function satisfying both sos conditions in (4.5) and (4.6) a *sum of squares Lyapunov function*. We emphasize that this is the sensible definition of a sum of squares Lyapunov function and not what the name may suggest, which is a Lyapunov function that is a sum of squares. Indeed, the underlying semidefinite program will find a Lyapunov function V if and only if V satisfies *both* conditions (4.5) and (4.6).

Over the last decade, the applicability of sum of squares Lyapunov functions has been explored and extended in many directions and a multitude of sos techniques have been developed to tackle a range of problems in systems and control. We refer the reader to the by no means exhaustive list of works [76], [41], [43], [83], [131], [114], [133], [42], [6], [21], [159] and references therein. Despite the wealth of research in this area, the converse question of whether the existence of a polynomial Lyapunov function implies the existence of a sum of squares Lyapunov function has remained elusive. This question naturally comes in two variants:

Problem 1: Does existence of a polynomial Lyapunov function of a given degree imply existence of a polynomial Lyapunov function of the *same degree* that satisfies the sos conditions in (4.5) and (4.6)?

Problem 2: Does existence of a polynomial Lyapunov function of a given degree imply existence of a polynomial Lyapunov function of *possibly higher degree* that satisfies the sos conditions in (4.5) and (4.6)?

The notion of stability of interest in this chapter, for which we will study the questions above, is global asymptotic stability (GAS); see e.g. [86, Chap. 4] for a precise definition. Of course, a fundamental question that comes before the problems mentioned above is the following:

Problem 0: If a polynomial dynamical system is globally asymptotically stable, does it admit a polynomial Lyapunov function?

4.1.1 Contributions and organization of this chapter

In this chapter, we give explicit counterexamples that answer **Problem 0** and **Problem 1** in the negative. This is done in Section 4.3 and Subsection 4.4.2 respectively. On the other hand, in Subsection 4.4.3, we give a positive answer to **Problem 2** for the case where the vector field is homogeneous (Theorem 4.8) or when it is planar and an additional mild assumption is met (Theorem 4.10). The proofs of these two theorems are quite simple and rely on powerful Positivstellensatz results due to Scheiderer (Theorems 4.7 and 4.9). In Section 4.5, we extend these results to derive a converse sos Lyapunov theorem for robust stability of switched linear systems. It will be proven that if such a system is stable under arbitrary switching, then it admits a common polynomial Lyapunov function that is sos and that the negative of its derivative is also sos (Theorem 4.11). We also show that for switched linear systems (both in discrete and continuous time), if the inequality on the decrease condition of a Lyapunov function is satisfied as a sum of squares, then the Lyapunov function itself is automatically a sum of squares (Propositions 4.14 and 4.15). We list a number of related open problems in Section 4.6.

Before these contributions are presented, we establish a hardness result for the problem of deciding asymptotic stability of cubic homogeneous vector fields in the next section. We also present some byproducts of this result, including a Lyapunov-inspired technique for proving positivity of forms.

4.2 Complexity considerations for deciding stability of polynomial vector fields

It is natural to ask whether stability of equilibrium points of polynomial vector fields can be decided in finite time. In fact, this is a well-known question of Arnold that appears in [17]:

> "Is the stability problem for stationary points algorithmically decidable? The well-known Lyapounov theorem[1] solves the problem in the absence of eigenvalues with zero real parts. In more complicated cases, where the stability

[1]The theorem that Arnold is referring to here is the indirect method of Lyapunov related to linearization. This is not to be confused with Lyapunov's direct method (or the second method), which is what we are concerned with in sections that follow.

> depends on higher order terms in the Taylor series, there exists no algebraic criterion.
>
> Let a vector field be given by polynomials of a fixed degree, with rational coefficients. Does an algorithm exist, allowing to decide, whether the stationary point is stable?"

Later in [51], the question of Arnold is quoted with more detail:

> "In my problem the coefficients of the polynomials of known degree and of a known number of variables are written on the tape of the standard Turing machine in the standard order and in the standard representation. The problem is whether there exists an algorithm (an additional text for the machine independent of the values of the coefficients) such that it solves the stability problem for the stationary point at the origin (i.e., always stops giving the answer "stable" or "unstable").
>
> I hope, this algorithm exists if the degree is one. It also exists when the dimension is one. My conjecture has always been that there is no algorithm for some sufficiently high degree and dimension, perhaps for dimension 3 and degree 3 or even 2. I am less certain about what happens in dimension 2. Of course the nonexistence of a general algorithm for a fixed dimension working for arbitrary degree or for a fixed degree working for an arbitrary dimension, or working for all polynomials with arbitrary degree and dimension would also be interesting."

To our knowledge, there has been no formal resolution to these questions, neither for the case of stability in the sense of Lyapunov, nor for the case of asymptotic stability (in its local or global version). In [51], da Costa and Doria show that if the right hand side of the differential equation contains elementary functions (sines, cosines, exponentials, absolute value function, etc.), then there is no algorithm for deciding whether the origin is stable or unstable. They also present a dynamical system in [52] where one cannot decide whether a Hopf bifurcation will occur or whether there will be parameter values such that a stable fixed point becomes unstable. In earlier work, Arnold himself demonstrates some of the difficulties that arise in stability analysis of polynomial systems by presenting a parametric polynomial system in 3 variables and degree 5, where the boundary between stability and instability in parameter space is not a semialgebraic set [16]. A relatively larger number of undecidability results are available for questions related to other properties of polynomial vector fields, such as reachability [73] or boundedness of domain of definition [65], or for questions about stability of hybrid systems [30], [35], [34], [29]. We refer the interested reader to the survey papers in [37], [73], [156], [33], [36].

We are also interested to know whether the answer to the undecidability question for asymptotic stability changes if the dynamics is restricted to be homogeneous. A polynomial vector field $\dot{x} = f(x)$ is *homogeneous* if all entries of f are homogeneous polynomials of the same degree. Homogeneous systems are extensively studied in the literature on nonlinear control [149], [14], [68], [23], [74], [146], [106], and some of the results of this chapter (both negative and positive) are derived specifically for this class of systems. A basic fact about homogeneous vector fields is that for these systems the notions of local and global stability are equivalent. Indeed, a homogeneous vector field of degree d satisfies $f(\lambda x) = \lambda^d f(x)$ for any scalar λ, and therefore the value of f on the unit sphere determines its value everywhere. It is also well-known that an asymptotically stable homogeneous system admits a homogeneous Lyapunov funciton [72],[146].

Naturally, questions regarding complexity of deciding asymptotic stability and questions about existence of Lyapunov functions are related. For instance, if one proves that for a class of polynomial vector fields, asymptotic stability implies existence of a polynomial Lyapunov function together with a computable upper bound on its degree, then the question of asymptotic stability for that class becomes decidable. This is due to the fact that given any polynomial system and any integer d, the question of deciding whether the system admits a polynomial Lyapunov function of degree d can be answered in finite time using quantifier elimination.

For the case of linear systems (i.e., homogeneous systems of degree 1), the situation is particularly nice. If such a system is asymptotically stable, then there always exists a *quadratic* Lyapunov function. Asymptotic stability of a linear system $\dot{x} = Ax$ is equivalent to the easily checkable algebraic criterion that the eigenvalues of A be in the open left half complex plane. Deciding this property of the matrix A can formally be done in polynomial time, e.g. by solving a Lyapunov equation [36].

Moving up in the degree, it is not difficult to show that if a homogeneous polynomial vector field has even degree, then it can never be asymptotically stable; see e.g. [72, p. 283]. So the next interesting case occurs for homogeneous vector fields of degree 3. We will prove below that determining asymptotic stability for such systems is strongly NP-hard. This gives a lower bound on the complexity of this problem. It is an interesting open question to investigate whether in this specific setting, the problem is also undecidable.

One implication of our NP-hardness result is that unless P=NP, we should not expect sum of squares Lyapunov functions of "low enough" degree to always exist, even when the analysis is restricted to cubic homogeneous vector fields. The semidefinite program arising from a search for an sos Lyapunov function of degree $2d$ for such a vector field in n variables has size in the order of $\binom{n+d}{d+1}$. This number is polynomial in n for fixed d (but exponential in n when d grows linearly in n). Therefore, unlike the case of linear systems, we should not hope to have a bound on the degree of sos Lyapunov functions that is independent of the dimension.

We postpone our study of existence of sos Lyapunov functions to Section 4.4 and proceed for now with the following complexity result.

Theorem 4.1. *Deciding asymptotic stability of homogeneous cubic polynomial vector fields is strongly NP-hard.*

The main intuition behind the proof of this theorem is the following idea: We will relate the solution of a combinatorial problem not to the behavior of the trajectories of a cubic vector field that are hard to get a handle on, but instead to properties of a Lyapunov function that proves asymptotic stability of this vector field. As we will see shortly, insights from Lyapunov theory make the proof of this theorem quite simple. The reduction is broken into two steps:

$$\begin{gathered}\text{ONE-IN-THREE 3SAT}\\ \downarrow\\ \text{positivity of quartic forms}\\ \downarrow\\ \text{asymptotic stability of cubic vector fields}\end{gathered}$$

In the course of presenting these reductions, we will also discuss some corollaries that are not directly related to our study of asymptotic stability, but are of independent interest.

4.2.1 Reduction from ONE-IN-THREE 3SAT to positivity of quartic forms

As we remarked in Chapter 2, NP-hardness of deciding nonnegativity (i.e., positive semidefiniteness) of quartic forms is well-known. The proof commonly cited in the literature is based on a reduction from the matrix copositivity problem [109]: given a symmetric $n \times n$ matrix Q, decide whether $x^TQx \geq 0$ for all x's that are elementwise nonnegative. Clearly, a matrix Q is copositive if and only if the quartic form z^TQz, with $z_i := x_i^2$, is nonnegative. The original reduction [109] proving NP-hardness of testing matrix copositivity is from the subset sum problem and only establishes weak NP-hardness. However, reductions from the stable set problem to matrix copositivity are also known [56], [58] and they result in NP-hardness in the strong sense. Alternatively, strong NP-hardness of deciding nonnegativity of quartic forms follows immediately from NP-hardness of deciding convexity of quartic forms (proven in Chapter 2) or from NP-hardness of deciding nonnegativity of biquadratic forms (proven in [97]).

For reasons that will become clear shortly, we are interested in showing hardness of deciding *positive definiteness* of quartic forms as opposed to positive semidefiniteness. This is in some sense even easier to accomplish. A very straightforward reduction from 3SAT proves NP-hardness of deciding positive definiteness of polynomials of degree 6. By using ONE-IN-THREE 3SAT instead, we will reduce the degree of the polynomial from 6 to 4.

Proposition 4.2. *It is strongly*[2] *NP-hard to decide whether a homogeneous polynomial of degree* 4 *is positive definite.*

Proof. We give a reduction from ONE-IN-THREE 3SAT which is known to be NP-complete [61, p. 259]. Recall that in ONE-IN-THREE 3SAT, we are given a 3SAT instance (i.e., a collection of clauses, where each clause consists of exactly three literals, and each literal is either a variable or its negation) and we are asked to decide whether there exists a $\{0,1\}$ assignment to the variables that makes the expression true with the additional property that each clause has *exactly one* true literal.

To avoid introducing unnecessary notation, we present the reduction on a specific instance. The pattern will make it obvious that the general construction is no different. Given an instance of ONE-IN-THREE 3SAT, such as the following

$$(x_1 \vee \bar{x}_2 \vee x_4) \wedge (\bar{x}_2 \vee \bar{x}_3 \vee x_5) \wedge (\bar{x}_1 \vee x_3 \vee \bar{x}_5) \wedge (x_1 \vee x_3 \vee x_4), \tag{4.7}$$

we define the quartic polynomial p as follows:

$$\begin{array}{rcl} p(x) & = & \sum_{i=1}^{5} x_i^2(1-x_i)^2 \\ & & +(x_1+(1-x_2)+x_4-1)^2+((1-x_2)+(1-x_3)+x_5-1)^2 \\ & & +((1-x_1)+x_3+(1-x_5)-1)^2+(x_1+x_3+x_4-1)^2. \end{array} \tag{4.8}$$

Having done so, our claim is that $p(x) > 0$ for all $x \in \mathbb{R}^5$ (or generally for all $x \in \mathbb{R}^n$) if and only if the ONE-IN-THREE 3SAT instance is not satisfiable. Note that p is a sum of squares and therefore nonnegative. The only possible locations for zeros of p are by construction among the points in $\{0,1\}^5$. If there is a satisfying Boolean assignment x to (4.7) with exactly one true literal per clause, then p will vanish at point x. Conversely, if there are no such satisfying assignments, then for any point in $\{0,1\}^5$, at least one of the terms in (4.8) will be positive and hence p will have no zeros.

It remains to make p homogeneous. This can be done via introducing a new scalar variable y. If we let

$$p_h(x,y) = y^4 p(\tfrac{x}{y}), \tag{4.9}$$

[2] Just like our results in Chapter 2, the NP-hardness results of this section will all be in the strong sense. From here on, we will drop the prefix "strong" for brevity.

then we claim that p_h (which is a quartic form) is positive definite if and only if p constructed as in (4.8) has no zeros.[3] Indeed, if p has a zero at a point x, then that zero is inherited by p_h at the point $(x, 1)$. If p has no zeros, then (4.9) shows that p_h can only possibly have zeros at points with $y = 0$. However, from the structure of p in (4.8) we see that

$$p_h(x, 0) = x_1^4 + \cdots + x_5^4,$$

which cannot be zero (except at the origin). This concludes the proof. □

We present a simple corollary of the reduction we just gave on a problem that is of relevance in polynomial integer programming.[4] Recall from Chapter 2 (Definition 2.7) that a basic semialgebraic set is a set defined by a finite number of polynomial inequalities:

$$\mathcal{S} = \{x \in \mathbb{R}^n |\ f_i(x) \geq 0,\ i = 1, \ldots, m\}. \tag{4.10}$$

Corollary 4.3. *Given a basic semialgebraic set, it is NP-hard to decide if the set contains a lattice point, i.e., a point with integer coordinates. This is true even when the set is defined by one constraint ($m = 1$) and the defining polynomial has degree 4.*

Proof. Given an instance of ONE-IN-THREE 3SAT, we define a polynomial p of degree 4 as in (4.8), and let the basic semialgebraic set be given by

$$\mathcal{S} = \{x \in \mathbb{R}^n |\ -p(x) \geq 0\}.$$

Then, by Proposition 4.2, if the ONE-IN-THREE 3SAT instance is not satisfiable, the set $\mathcal{S}$ is empty and hence has no lattice points. Conversely, if the instance is satisfiable, then $\mathcal{S}$ contains at least one point belonging to $\{0, 1\}^n$ and therefore has a lattice point. □

By using the celebrated result on undecidability of checking existence of integer solutions to polynomial equations (Hilbert's 10th problem), one can show that the problem considered in the corollary above is in fact undecidable [129]. The same is true for quadratic integer programming when both the dimension n and the

[3] In general, homogenization does not preserve positivity. For example, as shown in [138], the polynomial $x_1^2 + (1 - x_1x_2)^2$ has no zeros, but its homogenization $x_1^2y^2 + (y^2 - x_1x_2)^2$ has zeros at the points $(1, 0, 0)^T$ and $(0, 1, 0)^T$. Nevertheless, positivity is preserved under homogenization for the special class of polynomials constructed in this reduction, essentially because polynomials of type (4.8) have no zeros at infinity.

[4] We are thankful to Amitabh Basu and Jesús De Loera for raising this question during a visit at UC Davis, and for later insightful discussions.

number of constraints m are allowed to grow as part of the input [84]. The question of deciding existence of lattice points in polyhedra (i.e., the case where degree of f_i in (4.10) is 1 for all i) is also interesting and in fact very well-studied. For polyhedra, if both n and m are allowed to grow, then the problem is NP-hard. This can be seen e.g. as a corollary of the NP-hardness of the INTEGER KNAPSACK problem (though this is NP-hardness in the weak sense); see [61, p. 247]. However, if n is fixed and m grows, it follows from a result of Lenstra [94] that the problem can be decided in polynomial time. The same is true if m is fixed and n grows [153, Cor. 18.7c]. See also [115].

4.2.2 Reduction from positivity of quartic forms to asymptotic stability of cubic vector fields

We now present the second step of the reduction and finish the proof of Theorem 4.1.

Proof of Theorem 4.1. We give a reduction from the problem of deciding positive definiteness of quartic forms, whose NP-hardness was established in Proposition 4.2. Given a quartic form $V := V(x)$, we define the polynomial vector field

$$\dot{x} = -\nabla V(x). \tag{4.11}$$

Note that the vector field is homogeneous of degree 3. We claim that the above vector field is (locally or equivalently globally) asymptotically stable if and only if V is positive definite. First, we observe that by construction

$$\dot{V}(x) = \langle \nabla V(x), \dot{x} \rangle = -||\nabla V(x)||^2 \leq 0. \tag{4.12}$$

Suppose V is positive definite. By Euler's identity for homogeneous functions,[5] we have $V(x) = \frac{1}{4}x^T \nabla V(x)$. Therefore, positive definiteness of V implies that $\nabla V(x)$ cannot vanish anywhere except at the origin. Hence, $\dot{V}(x) < 0$ for all $x \neq 0$. In view of Lyapunov's theorem (see e.g. [86, p. 124]), and the fact that a positive definite homogeneous function is radially unbounded, it follows that the system in (4.11) is globally asymptotically stable.

For the converse direction, suppose (4.11) is GAS. Our first claim is that global asymptotic stability together with $\dot{V}(x) \leq 0$ implies that V must be positive semidefinite. This follows from the following simple argument, which we have also previously presented in [12] for a different purpose. Suppose for the sake of contradiction that for some $\hat{x} \in \mathbb{R}^n$ and some $\epsilon > 0$, we had $V(\hat{x}) = -\epsilon < 0$.

[5]Euler's identity is easily derived by differentiating both sides of the equation $V(\lambda x) = \lambda^d V(x)$ with respect to λ and setting $\lambda = 1$.

Consider a trajectory $x(t;\hat{x})$ of system (4.11) that starts at initial condition $\hat{x}$, and let us evaluate the function V on this trajectory. Since $V(\hat{x}) = -\epsilon$ and $\dot{V}(x) \leq 0$, we have $V(x(t;\hat{x})) \leq -\epsilon$ for all $t > 0$. However, this contradicts the fact that by global asymptotic stability, the trajectory must go to the origin, where V, being a form, vanishes.

To prove that V is positive definite, suppose by contradiction that for some nonzero point $x^* \in \mathbb{R}^n$ we had $V(x^*) = 0$. Since we just proved that V has to be positive semidefinite, the point x^* must be a global minimum of V. Therefore, as a necessary condition of optimality, we should have $\nabla V(x^*) = 0$. But this contradicts the system in (4.11) being GAS, since the trajectory starting at x^* stays there forever and can never go to the origin. $\square$

Perhaps of independent interest, the reduction we just gave suggests a method for proving positive definiteness of forms. Given a form V, we can construct a dynamical system as in (4.11), and then any method that we may have for proving stability of vector fields (e.g. the use of various kinds of Lyapunov functions) can serve as an algorithm for proving positivity of V. In particular, if we use a polynomial Lyapunov function W to prove stability of the system in (4.11), we get the following corollary.

Corollary 4.4. *Let V and W be two forms of possibly different degree. If W is positive definite, and $\langle \nabla W, \nabla V \rangle$ is positive definite, then V is positive definite.*

One interesting fact about this corollary is that its algebraic version with sum of squares replaced for positivity is not true. In other words, we can have W sos (and positive definite), $\langle \nabla W, \nabla V \rangle$ sos (and positive definite), but V not sos. This gives us a way of proving positivity of some polynomials that are not sos, using *only* sos certificates. Given a form V, since the expression $\langle \nabla W, \nabla V \rangle$ is linear in the coefficients of W, we can use sos programming to search for a form W that satisfies W sos and $\langle \nabla W, \nabla V \rangle$ sos, and this would prove positivity of V. The following example demonstrates the potential usefulness of this approach.

Example 4.2.1. Consider the following form of degree 6:

$$V(x) = x_1^4 x_2^2 + x_1^2 x_2^4 - 3x_1^2 x_2^2 x_3^2 + x_3^6 + \frac{1}{250}(x_1^2 + x_2^2 + x_3^2)^3. \tag{4.13}$$

One can check that this polynomial is not a sum of squares. (In fact, this is the Motzkin form presented in equation (3.1) of Chapter 3 slightly perturbed.) On the other hand, we can use YALMIP [98] together with the SDP solver SeDuMi [157] to search for a form W satisfying

$$\begin{array}{rl} W & \text{sos} \\ \langle \nabla W, \nabla V \rangle & \text{sos.} \end{array} \tag{4.14}$$

If we parameterize W as a quadratic form, no feasible solution will be returned form the solver. However, when we increase the degree of W from 2 to 4, the solver returns the following polynomial

$$\begin{aligned} W(x) &= 9x_2^4 + 9x_1^4 - 6x_1^2x_2^2 + 6x_1^2x_3^2 + 6x_2^2x_3^2 + 3x_3^4 - x_1^3x_2 - x_1x_2^3 \\ &\quad -x_1^3x_3 - 3x_1^2x_2x_3 - 3x_1x_2^2x_3 - x_2^3x_3 - 4x_1x_2x_3^2 - x_1x_3^3 - x_2x_3^3 \end{aligned}$$

that satisfies both sos constrains in (4.14). The Gram matrices in these sos decompositions are positive definite. Therefore, W and $\langle \nabla W, \nabla V \rangle$ are positive definite forms. Hence, by Corollary 4.4, we have a proof that V in (4.13) is positive definite. $\triangle$

Interestingly, approaches of this type that use gradient information for proving positivity of polynomials with sum of squares techniques have been studied by Nie, Demmel, and Sturmfels in [113], though the derivation there is not Lyapunov-inspired.

■ 4.3 Non-existence of polynomial Lyapunov functions

As we mentioned at the beginning of this chapter, the question of global asymptotic stability of polynomial vector fields is commonly addressed by seeking a Lyapunov function that is polynomial itself. This approach has become further prevalent over the past decade due to the fact that we can use sum of squares techniques to algorithmically search for such Lyapunov functions. The question therefore naturally arises as to whether existence of polynomial Lyapunov functions is necessary for global stability of polynomial systems. In this section, we give a negative answer to this question by presenting a remarkably simple counterexample. In view of the fact that globally asymptotically stable linear systems always admit quadratic Lyapunov functions, it is quite interesting to observe that the following vector field that is arguably "the next simplest system" to consider does not admit a polynomial Lyapunov function of any degree.

Theorem 4.5. *Consider the polynomial vector field*

$$\begin{aligned} \dot{x} &= -x + xy \\ \dot{y} &= -y. \end{aligned} \tag{4.15}$$

The origin is a globally asymptotically stable equilibrium point, but the system does not admit a polynomial Lyapunov function.

Proof. Let us first show that the system is GAS. Consider the Lyapunov function

$$V(x, y) = \ln(1 + x^2) + y^2,$$

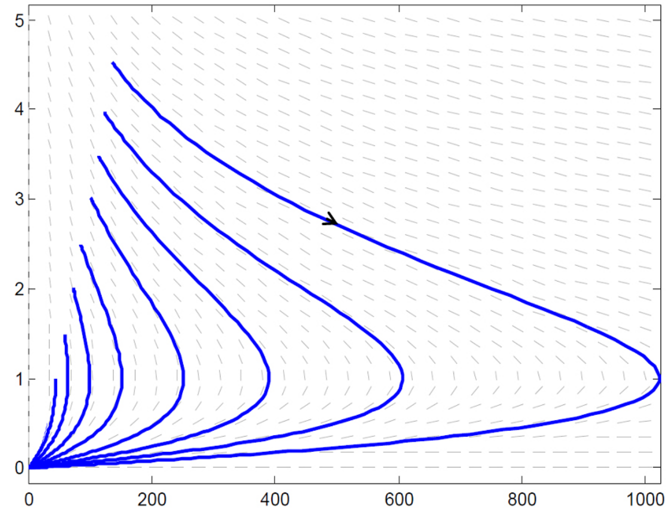


Figure 4.1. Typical trajectories of the vector field in (4.15) starting from initial conditions in the nonnegative orthant.

which clearly vanishes at the origin, is strictly positive for all $(x, y) \neq (0, 0)$, and is radially unbounded. The derivative of $V(x, y)$ along the trajectories of (4.15) is given by

$$\begin{aligned}\dot{V}(x,y) &= \tfrac{\partial V}{\partial x}\dot{x} + \tfrac{\partial V}{\partial y}\dot{y} \\ &= \tfrac{2x^2(y-1)}{1+x^2} - 2y^2 \\ &= -\tfrac{x^2+2y^2+x^2y^2+(x-xy)^2}{1+x^2},\end{aligned}$$

which is obviously strictly negative for all $(x, y) \neq (0, 0)$. In view of Lyapunov's stability theorem (see e.g. [86, p. 124]), this shows that the origin is globally asymptotically stable.

Let us now prove that no positive definite polynomial Lyapunov function (of any degree) can decrease along the trajectories of system (4.15). The proof will be based on simply considering the value of a candidate Lyapunov function at two specific points. We will look at trajectories on the nonnegative orthant, with initial conditions on the line $(k, \alpha k)$ for some constant $\alpha > 0$, and then observe the location of the crossing of the trajectory with the horizontal line $y = \alpha$. We will argue that by taking k large enough, the trajectory will have to travel "too far east" (see Figure 4.1) and this will make it impossible for any polynomial Lyapunov function to decrease.

To do this formally, we start by noting that we can explicitly solve for the solution $(x(t), y(t))$ of the vector field in (4.15) starting from any initial condition $(x(0), y(0))$:

$$\begin{aligned} x(t) &= x(0)e^{[y(0)-y(0)e^{-t}-t]} \\ y(t) &= y(0)e^{-t}. \end{aligned} \tag{4.16}$$

Consider initial conditions

$$(x(0), y(0)) = (k, \alpha k)$$

parameterized by $k > 1$ and for some fixed constant $\alpha > 0$. From the explicit solution in (4.16) we have that the time t^* it takes for the trajectory to cross the line $y = \alpha$ is

$$t^* = \ln(k),$$

and that the location of this crossing is given by

$$(x(t^*), y(t^*)) = (e^{\alpha(k-1)}, \alpha).$$

Consider now any candidate nonnegative polynomial function $V(x, y)$ that depends on both x and y (as any Lyapunov function should). Since $k > 1$ (and thus, $t^* > 0$), for $V(x, y)$ to be a valid Lyapunov function, it must satisfy $V(x(t^*), y(t^*)) < V(x(0), y(0))$, i.e.,

$$V(e^{\alpha(k-1)}, \alpha) < V(k, \alpha k).$$

However, this inequality cannot hold for k large enough, since for a generic fixed α, the left hand side grows exponentially in k whereas the right hand side grows only polynomially in k. The only subtlety arises from the fact that $V(e^{\alpha(k-1)}, \alpha)$ could potentially be a constant for some particular choices of α. However, for any polynomial $V(x, y)$ with nontrivial dependence on y, this may happen for at most finitely many values of α. Therefore, any generic choice of α would make the argument work. □

Example of Bacciotti and Rosier. After our counterexample above was submitted for publication, Christian Ebenbauer brought to our attention an earlier counterexample of Bacciotti and Rosier [22, Prop. 5.2] that achieves the same goal (though by using irrational coefficients). We will explain the differences between the two examples below. At the time of submission of our result, we were under the impression that no such examples were known, partly because of a recent reference in the controls literature that ends its conclusion with the following statement [126], [127]:

> "Still unresolved is the fundamental question of whether *globally* stable vector fields will also admit sum-of-squares Lyapunov functions."

In [126], [127], what is referred to as a sum of squares Lyapunov function (in contrast to our terminology here) is a Lyapunov function that is a sum of squares, with no sos requirements on its derivative. Therefore, the fundamental question referred to above is on existence of a polynomial Lyapunov function. If one were to exist, then we could simply square it to get another polynomial Lyapunov function that is a sum of squares (see Lemma 4.6).

The example of Bacciotti and Rosier is a vector field in 2 variables and degree 5 that is GAS but has no polynomial (and no analytic) Lyapunov function even around the origin. Their very clever construction is complementary to our example in the sense that what creates trouble for existence of polynomial Lyapunov functions in our Theorem 4.5 is growth rates *arbitrarily far* away from the origin, whereas the problem arising in their example is slow decay rates *arbitrarily close* to the origin. The example crucially relies on a parameter that appears as part of the coefficients of the vector field being *irrational*. (Indeed, one easily sees that if that parameter is rational, their vector field does admit a polynomial Lyapunov function.) In practical applications where computational techniques for searching over Lyapunov functions on finite precision machines are used, such issues with irrationality of the input cannot occur. By contrast, the example in (4.15) is much less contrived and demonstrates that non-existence of polynomial Lyapunov functions can happen for extremely simple systems that may very well appear in applications.

In [125], Peet has shown that locally exponentially stable polynomial vector fields admit polynomial Lyapunov functions on compact sets. The example of Bacciotti and Rosier implies that the assumption of exponential stability indeed cannot be dropped.

■ 4.4 (Non)-existence of sum of squares Lyapunov functions

In this section, we suppose that the polynomial vector field at hand admits a polynomial Lyapunov function, and we would like to investigate whether such a Lyapunov function can be found with sos programming. In other words, we would like to see whether the constrains in (4.5) and (4.6) are more conservative than the true Lyapunov inequalities in (4.2) and (4.3). We think of the sos Lyapunov conditions in (4.5) and (4.6) as sufficient conditions for the strict inequalities in (4.2) and (4.3) even though sos decomposition in general merely guarantees non-strict inequalities. The reason for this is that when an sos feasibility problem is strictly feasible, the polynomials returned by interior point algorithms are automatically positive definite (see [1, p. 41] for more discussion).[6]

We shall emphasize that existence of nonnegative polynomials that are not sums of squares does not imply on its own that the sos conditions in (4.5) and (4.6) are more conservative than the Lyapunov inequalities in (4.2) and (4.3). Since Lyapunov functions are not in general unique, it could happen that within the set of valid polynomial Lyapunov functions of a given degree, there is always at least one that satisfies the sos conditions (4.5) and (4.6). Moreover, many of the known examples of nonnegative polynomials that are not sos have multiple zeros and local minima [138] and therefore cannot serve as Lyapunov functions. Indeed, if a function has a local minimum other than the origin, then its value evaluated on a trajectory starting from the local minimum would not be decreasing.

4.4.1 A motivating example

The following example will help motivate the kind of questions that we are addressing in this section.

Example 4.4.1. Consider the dynamical system

$$\begin{array}{rcl} \dot{x_1} & = & -0.15x_1^7 + 200x_1^6x_2 - 10.5x_1^5x_2^2 - 807x_1^4x_2^3 \\ & & +14x_1^3x_2^4 + 600x_1^2x_2^5 - 3.5x_1x_2^6 + 9x_2^7 \\ \\ \dot{x_2} & = & -9x_1^7 - 3.5x_1^6x_2 - 600x_1^5x_2^2 + 14x_1^4x_2^3 \\ & & +807x_1^3x_2^4 - 10.5x_1^2x_2^5 - 200x_1x_2^6 - 0.15x_2^7. \end{array} \tag{4.17}$$

A typical trajectory of the system that starts from the initial condition $x_0 = (2,2)^T$ is plotted in Figure 4.2. Our goal is to establish global asymptotic stability of the origin by searching for a polynomial Lyapunov function. Since the vector field is homogeneous, the search can be restricted to homogeneous Lyapunov functions [72], [146]. To employ the sos technique, we can use the software package SOSTOOLS [132] to search for a Lyapunov function satisfying the sos conditions (4.5) and (4.6). However, if we do this, we will not find any Lyapunov functions of degree 2, 4, or 6. If needed, a certificate from the dual semidefinite program can be obtained, which would prove that no polynomial of degree up to 6 can satisfy the sos requirements (4.5) and (4.6).

At this point we are faced with the following question. Does the system really not admit a Lyapunov function of degree 6 that satisfies the true Lyapunov

[6]We expect the reader to recall the basic definitions and concepts from Subsection 3.2.1 of the previous chapter. Throughout, when we say a Lyapunov function (or the negative of its derivative) is *positive definite*, we mean that it is positive everywhere except possibly at the origin.

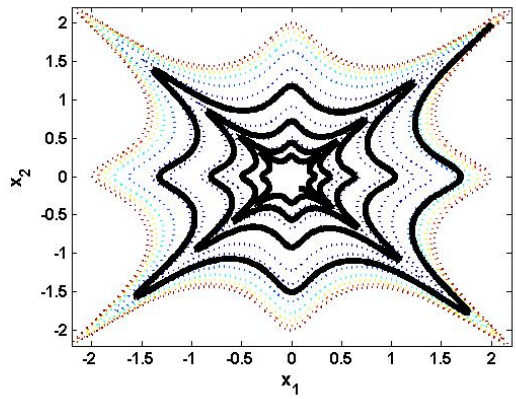


Figure 4.2. A typical trajectory of the vector filed in Example 4.4.1 (solid), level sets of a degree 8 polynomial Lyapunov function (dotted).

inequalities in (4.2), (4.3)? Or is the failure due to the fact that the sos conditions in (4.5), (4.6) are more conservative?

Note that when searching for a degree 6 Lyapunov function, the sos constraint in (4.5) is requiring a homogeneous polynomial in 2 variables and of degree 6 to be a sum of squares. The sos condition (4.6) on the derivative is also a condition on a homogeneous polynomial in 2 variables, but in this case of degree 12. (This is easy to see from $\dot{V} = \langle \nabla V, f \rangle$.) Recall from Theorem 3.1 of the previous chapter that nonnegativity and sum of squares are equivalent notions for homogeneous bivariate polynomials, irrespective of the degree. Hence, we now have a proof that this dynamical system truly does not have a Lyapunov function of degree 6 (or lower).

This fact is perhaps geometrically intuitive. Figure 4.2 shows that the trajectory of this system is stretching out in 8 different directions. So, we would expect the degree of the Lyapunov function to be at least 8. Indeed, when we increase the degree of the candidate function to 8, SOSTOOLS and the SDP solver SeDuMi [157] succeed in finding the following Lyapunov function:

$$\begin{aligned} V(x) \;\; = \;\; & 0.02x_1^8 + 0.015x_1^7x_2 + 1.743x_1^6x_2^2 - 0.106x_1^5x_2^3 \\ & -3.517x_1^4x_2^4 + 0.106x_1^3x_2^5 + 1.743x_1^2x_2^6 \\ & -0.015x_1x_2^7 + 0.02x_2^8. \end{aligned}$$

The level sets of this Lyapunov function are plotted in Figure 4.2 and are clearly invariant under the trajectory. △

4.4.2 A counterexample

Unlike the scenario in the previous example, we now show that a failure in finding a Lyapunov function of a particular degree via sum of squares programming can also be due to the gap between nonnegativity and sum of squares. What will be conservative in the following counterexample is the sos condition on the derivative.[7]

Consider the dynamical system

$$\begin{aligned}\dot{x_1} &= -x_1^3x_2^2 + 2x_1^3x_2 - x_1^3 + 4x_1^2x_2^2 - 8x_1^2x_2 + 4x_1^2 \\ &\quad -x_1x_2^4 + 4x_1x_2^3 - 4x_1 + 10x_2^2 \\ \dot{x_2} &= -9x_1^2x_2 + 10x_1^2 + 2x_1x_2^3 - 8x_1x_2^2 - 4x_1 - x_2^3 \\ &\quad +4x_2^2 - 4x_2.\end{aligned} \tag{4.18}$$

One can verify that the origin is the only equilibrium point for this system, and therefore it makes sense to investigate global asymptotic stability. If we search for a quadratic Lyapunov function for (4.18) using sos programming, we will not find one. It will turn out that the corresponding semidefinite program is infeasible. We will prove shortly why this is the case, i.e, why no quadratic function V can satisfy

$$\begin{array}{rl} V & \text{sos} \\ -\dot{V} & \text{sos.} \end{array} \tag{4.19}$$

Nevertheless, we claim that

$$V(x) = \frac{1}{2}x_1^2 + \frac{1}{2}x_2^2 \tag{4.20}$$

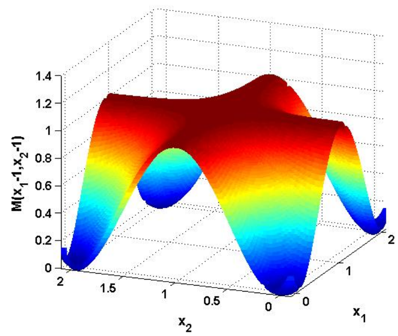
is a valid Lyapunov function. Indeed, one can check that

$$\dot{V}(x) = x_1\dot{x}_1 + x_2\dot{x}_2 = -M(x_1 - 1, x_2 - 1), \tag{4.21}$$

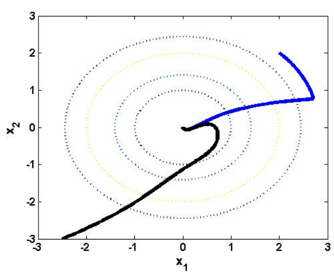
where $M(x_1, x_2)$ is the Motzkin polynomial [107]:

$$M(x_1, x_2) = x_1^4x_2^2 + x_1^2x_2^4 - 3x_1^2x_2^2 + 1.$$

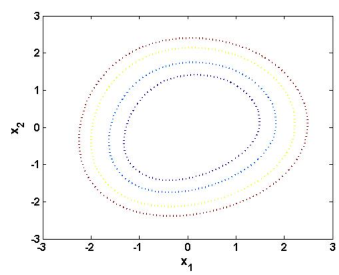
This polynomial is just a dehomogenized version of the Motzkin form presented before, and it has the property of being nonnegative but not a sum of squares. The polynomial $\dot{V}$ is strictly negative everywhere, except for the origin and three other points $(0,2)^T$, $(2,0)^T$, and $(2,2)^T$, where $\dot{V}$ is zero. However, at each of these three points we have $\dot{x} \neq 0$. Once the trajectory reaches any of these three points, it will be kicked out to a region where $\dot{V}$ is strictly negative. Therefore,



(a) Shifted Motzkin polynomial is nonnegative but not sos.



(b) Typical trajectories of (4.18) (solid), level sets of V (dotted).



(c) Level sets of a quartic Lyapunov function found through sos programming.

Figure 4.3. The quadratic polynomial $\frac{1}{2}x_1^2 + \frac{1}{2}x_2^2$ is a valid Lyapunov function for the vector field in (4.18) but it is not detected through sos programming.

by LaSalle's invariance principle (see e.g. [86, p. 128]), the quadratic Lyapunov function in (4.20) proves global asymptotic stability of the origin of (4.18).

The fact that $\dot{V}$ is zero at three points other than the origin is not the reason why sos programming is failing. After all, when we impose the condition that $-\dot{V}$ should be sos, we allow for the possibility of a non-strict inequality. The reason why our sos program does not recognize (4.20) as a Lyapunov function is that the shifted Motzkin polynomial in (4.21) is nonnegative but it is not a sum of squares. This sextic polynomial is plotted in Figure 4.3(a). Trajectories of (4.18) starting at $(2,2)^T$ and $(-2.5,-3)^T$ along with level sets of V are shown in Figure 4.3(b).

So far, we have shown that V in (4.20) is a valid Lyapunov function but does not satisfy the sos conditions in (4.19). We still need to show why no other

[7]This counterexample has appeared in our earlier work [1] but not with a complete proof.

quadratic Lyapunov function

$$U(x) = c_1 x_1^2 + c_2 x_1 x_2 + c_3 x_2^2 \tag{4.22}$$

can satisfy the sos conditions either.[8] We will in fact prove the stronger statement that V in (4.20) is the only valid quadratic Lyapunov function for this system up to scaling, i.e., any quadratic function U that is not a scalar multiple of $\frac{1}{2}x_1^2 + \frac{1}{2}x_2^2$ cannot satisfy $U \geq 0$ and $-\dot{U} \geq 0$. It will even be the case that no such U can satisfy $-\dot{U} \geq 0$ alone. (The latter fact is to be expected since global asymptotic stability of (4.18) together with $-\dot{U} \geq 0$ would automatically imply $U \geq 0$; see [12, Theorem 1.1].)

So, let us show that $-\dot{U} \geq 0$ implies U is a scalar multiple of $\frac{1}{2}x_1^2 + \frac{1}{2}x_2^2$. Because Lyapunov functions are closed under positive scalings, without loss of generality we can take $c_1 = 1$. One can check that

$$-\dot{U}(0,2) = -80c_2,$$

so to have $-\dot{U} \geq 0$, we need $c_2 \leq 0$. Similarly,

$$-\dot{U}(2,2) = -288c_1 + 288c_3,$$

which implies that $c_3 \geq 1$. Let us now look at

$$\begin{array}{rcl} -\dot{U}(x_1,1) & = & -c_2 x_1^3 + 10c_2 x_1^2 + 2c_2 x_1 - 10c_2 - 2c_3 x_1^2 \\ & & +20c_3 x_1 + 2c_3 + 2x_1^2 - 20x_1. \end{array} \tag{4.23}$$

If we let $x_1 \to -\infty$, the term $-c_2 x_1^3$ dominates this polynomial. Since $c_2 \leq 0$ and $-\dot{U} \geq 0$, we conclude that $c_2 = 0$. Once c_2 is set to zero in (4.23), the dominating term for x_1 large will be $(2-2c_3)x_1^2$. Therefore to have $-\dot{U}(x_1,1) \geq 0$ as $x_1 \to \pm\infty$ we must have $c_3 \leq 1$. Hence, we conclude that $c_1 = 1, c_2 = 0, c_3 = 1$, and this finishes the proof.

Even though sos programming failed to prove stability of the system in (4.18) with a quadratic Lyapunov function, if we increase the degree of the candidate Lyapunov function from 2 to 4, then SOSTOOLS succeeds in finding a quartic Lyapunov function

$$\begin{array}{rcl} W(x) & = & 0.08x_1^4 - 0.04x_1^3 + 0.13x_1^2 x_2^2 + 0.03x_1^2 x_2 \\ & & +0.13x_1^2 + 0.04x_1 x_2^2 - 0.15x_1 x_2 \\ & & +0.07x_2^4 - 0.01x_2^3 + 0.12x_2^2, \end{array}$$

[8] Since we can assume that the Lyapunov function U and its gradient vanish at the origin, linear or constant terms are not needed in (4.22).

which satisfies the sos conditions in (4.19). The level sets of this function are close to circles and are plotted in Figure 4.3(c).

Motivated by this example, it is natural to ask whether it is always true that upon increasing the degree of the Lyapunov function one will find Lyapunov functions that satisfy the sum of squares conditions in (4.19). In the next subsection, we will prove that this is indeed the case, at least for planar systems such as the one in this example, and also for systems that are homogeneous.

4.4.3 Converse sos Lyapunov theorems

In [126], [127], it is shown that if a system admits a polynomial Lyapunov function, then it also admits one that is a sum of squares. However, the results there do not lead to any conclusions as to whether the negative of the derivative of the Lyapunov function is sos, i.e, whether condition (4.6) is satisfied. As we remarked before, there is therefore no guarantee that the semidefinite program can find such a Lyapunov function. Indeed, our counterexample in the previous subsection demonstrated this very phenomenon.

The proof technique used in [126],[127] is based on approximating the solution map using the Picard iteration and is interesting in itself, though the actual conclusion that a Lyapunov function that is sos exists has a far simpler proof which we give in the next lemma.

Lemma 4.6. *If a polynomial dynamical system has a positive definite polynomial Lyapunov function V with a negative definite derivative $\dot{V}$, then it also admits a positive definite polynomial Lyapunov function W which is a sum of squares.*

Proof. Take $W = V^2$. The negative of the derivative $-\dot{W} = -2V\dot{V}$ is clearly positive definite (though it may not be sos). $\square$

We will next prove a converse sos Lyapunov theorem that guarantees the derivative of the Lyapunov function will also satisfy the sos condition, though this result is restricted to homogeneous systems. The proof of this theorem relies on the following Positivstellensatz result due to Scheiderer.

Theorem 4.7 (Scheiderer, [151])**.** *Given any two positive definite homogeneous polynomials p and q, there exists an integer k such that pq^k is a sum of squares.*

Theorem 4.8. *Given a homogeneous polynomial vector field, suppose there exists a homogeneous polynomial Lyapunov function V such that V and $-\dot{V}$ are positive definite. Then, there also exists a homogeneous polynomial Lyapunov function W such that W is sos and $-\dot{W}$ is sos.*

Proof. Observe that V^2 and $-2V\dot{V}$ are both positive definite and homogeneous polynomials. Applying Theorem 4.7 to these two polynomials, we conclude the existence of an integer k such that $(-2V\dot{V})(V^2)^k$ is sos. Let

$$W = V^{2k+2}.$$

Then, W is clearly sos since it is a perfect even power. Moreover,

$$-\dot{W} = -(2k+2)V^{2k+1}\dot{V} = -(k+1)2V^{2k}V\dot{V}$$

is also sos by the previous claim.[9] □

Next, we develop a similar theorem that removes the homogeneity assumption from the vector field, but instead is restricted to vector fields on the plane. For this, we need another result of Scheiderer.

Theorem 4.9 (Scheiderer, [150, Cor. 3.12])**.** *Let $p := p(x_1, x_2, x_3)$ and $q := q(x_1, x_2, x_3)$ be two homogeneous polynomials in three variables, with p positive semidefinite and q positive definite. Then, there exists an integer k such that pq^k is a sum of squares.*

Theorem 4.10. *Given a (not necessarily homogeneous) polynomial vector field in two variables, suppose there exists a positive definite polynomial Lyapunov function V, with $-\dot{V}$ positive definite, and such that the highest order term of V has no zeros[10]. Then, there also exists a polynomial Lyapunov function W such that W is sos and $-\dot{W}$ is sos.*

Proof. Let $\tilde{V} = V + 1$. So, $\dot{\tilde{V}} = \dot{V}$. Consider the (non-homogeneous) polynomials $\tilde{V}^2$ and $-2\tilde{V}\dot{\tilde{V}}$ in the variables $x := (x_1, x_2)$. Let us denote the (even) degrees of these polynomials respectively by d_1 and d_2. Note that $\tilde{V}^2$ is nowhere zero and $-2\tilde{V}\dot{\tilde{V}}$ is only zero at the origin. Our first step is to homogenize these polynomials by introducing a new variable y. Observing that the homogenization of products of polynomials equals the product of homogenizations, we obtain the following two trivariate forms:

$$y^{2d_1}\tilde{V}^2(\tfrac{x}{y}), \tag{4.24}$$

$$-2y^{d_1}y^{d_2}\tilde{V}(\tfrac{x}{y})\dot{\tilde{V}}(\tfrac{x}{y}). \tag{4.25}$$

[9] Note that W constructed in this proof proves GAS since $-\dot{W}$ is positive definite and W itself being homogeneous and positive definite is automatically radially unbounded.

[10] This requirement is only slightly stronger than the requirement of radial unboundedness, which is imposed on V by Lyapunov's theorem anyway.

Since by assumption the highest order term of V has no zeros, the form in (4.24) is positive definite . The form in (4.25), however, is only positive semidefinite. In particular, since $\dot{\tilde{V}} = \dot{V}$ has to vanish at the origin, the form in (4.25) has a zero at the point $(x_1, x_2, y) = (0, 0, 1)$. Nevertheless, since Theorem 4.9 allows for positive semidefiniteness of one of the two forms, by applying it to the forms in (4.24) and (4.25), we conclude that there exists an integer k such that

$$-2y^{d_1(2k+1)}y^{d_2}\tilde{V}(\tfrac{x}{y})\dot{\tilde{V}}(\tfrac{x}{y})\tilde{V}^{2k}(\tfrac{x}{y}) \tag{4.26}$$

is sos. Let $W = \tilde{V}^{2k+2}$. Then, W is clearly sos. Moreover,

$$-\dot{W} = -(2k+2)\tilde{V}^{2k+1}\dot{\tilde{V}} = -(k+1)2\tilde{V}^{2k}\tilde{V}\dot{\tilde{V}}$$

is also sos because this polynomial is obtained from (4.26) by setting $y = 1$.[11] $\square$

4.5 Existence of sos Lyapunov functions for switched linear systems

The result of Theorem 4.8 extends in a straightforward manner to Lyapunov analysis of switched systems. In particular, we are interested in the highly-studied problem of stability analysis of arbitrary switched linear systems:

$$\dot{x} = A_i x, \quad i \in \{1, \ldots, m\}, \tag{4.27}$$

$A_i \in \mathbb{R}^{n \times n}$. We assume the minimum dwell time of the system is bounded away from zero. This guarantees that the solutions of (4.27) are well-defined. Existence of a common Lyapunov function is necessary and sufficient for (global) asymptotic stability under arbitrary switching (ASUAS) of system (4.27). The ASUAS of system (4.27) is equivalent to asymptotic stability of the linear differential inclusion

$$\dot{x} \in co\{A_i\}x, \quad i \in \{1, \ldots, m\},$$

where *co* here denotes the convex hull. It is also known that ASUAS of (4.27) is equivalent to exponential stability under arbitrary switching [15]. A common approach for analyzing the stability of these systems is to use the sos technique to search for a common polynomial Lyapunov function [131],[42]. We will prove the following result.

[11] Once again, we note that the function W constructed in this proof is radially unbounded, achieves its global minimum at the origin, and has $-\dot{W}$ positive definite. Therefore, W proves global asymptotic stability.

Theorem 4.11. *The switched linear system in (4.27) is asymptotically stable under arbitrary switching if and only if there exists a common homogeneous polynomial Lyapunov function W such that*

$$\begin{array}{rl} W & \text{sos} \\ -\dot{W}_i = -\langle \nabla W(x), A_i x \rangle & \text{sos}, \end{array}$$

for $i = 1, \ldots, m$, where the polynomials W and $-\dot{W}_i$ are all positive definite.

To prove this result, we will use the following theorem of Mason et al.

Theorem 4.12 (Mason et al., [103])**.** *If the switched linear system in (4.27) is asymptotically stable under arbitrary switching, then there exists a common homogeneous polynomial Lyapunov function V such that*

$$\begin{array}{rcl} V & > & 0 \quad \forall x \neq 0 \\ -\dot{V}_i(x) = -\langle \nabla V(x), A_i x \rangle & > & 0 \quad \forall x \neq 0, \end{array}$$

for $i = 1, \ldots, m$.

The next proposition is an extension of Theorem 4.8 to switched systems (not necessarily linear).

Proposition 4.13. *Consider an arbitrary switched dynamical system*

$$\dot{x} = f_i(x), \quad i \in \{1, \ldots, m\},$$

where $f_i(x)$ is a homogeneous polynomial vector field of degree d_i (the degrees of the different vector fields can be different). Suppose there exists a common positive definite homogeneous polynomial Lyapunov function V such that

$$-\dot{V}_i(x) = -\langle \nabla V(x), f_i(x) \rangle$$

is positive definite for all $i \in \{1, \ldots, m\}$. Then there exists a common homogeneous polynomial Lyapunov function W such that W is sos and the polynomials

$$-\dot{W}_i = -\langle \nabla W(x), f_i(x) \rangle,$$

for all $i \in \{1, \ldots, m\}$, are also sos.

Proof. Observe that for each i, the polynomials V^2 and $-2V\dot{V}_i$ are both positive definite and homogeneous. Applying Theorem 4.7 m times to these pairs of polynomials, we conclude the existence of positive integers k_i such that

$$(-2V\dot{V}_i)(V^2)^{k_i} \text{ is sos,} \tag{4.28}$$

for $i = 1, \ldots, m$. Let

$$k = \max\{k_1, \ldots, k_m\},$$

and let

$$W = V^{2k+2}.$$

Then, W is clearly sos. Moreover, for each i, the polynomial

$$\begin{array}{rcl} -\dot{W}_i & = & -(2k+2)V^{2k+1}\dot{V}_i \\ & = & -(k+1)2V\dot{V}_i V^{2k_i} V^{2(k-k_i)} \end{array}$$

is sos since $(-2V\dot{V}_i)(V^{2k_i})$ is sos by (4.28), $V^{2(k-k_i)}$ is sos as an even power, and products of sos polynomials are sos. $\square$

The proof of Theorem 4.11 now simply follows from Theorem 4.12 and Proposition 4.13 in the special case where $d_i = 1$ for all i.

Analysis of switched linear systems is also of great interest to us in discrete time. In fact, the subject of the next chapter will be on the study of systems of the type

$$x_{k+1} = A_i x_k, \quad i \in \{1, \ldots, m\}, \tag{4.29}$$

where at each time step the update rule can be given by any of the m matrices A_i. The analogue of Theorem 4.11 for these systems has already been proven by Parrilo and Jadbabaie in [122]. It is shown that if (4.29) is asymptotically stable under arbitrary switching, then there exists a homogeneous polynomial Lyapunov function W such that

$$\begin{array}{rl} W(x) & \text{sos} \\ W(x) - W(A_i x) & \text{sos}, \end{array}$$

for $i = 1, \ldots, m$. We will end this section by proving two related propositions of a slightly different flavor. It will be shown that for switched linear systems, both in discrete time and in continuous time, the sos condition on the Lyapunov function itself is never conservative, in the sense that if one of the "decrease inequalities" is sos, then the Lyapunov function is automatically sos. These propositions are really statements about linear systems, so we will present them that way. However, since stable linear systems always admit quadratic Lyapunov functions, the propositions are only interesting in the context where a common polynomial Lyapunov function for a switched linear system is seeked.

Proposition 4.14. *Consider the linear dynamical system $x_{k+1} = Ax_k$ in discrete time. Suppose there exists a positive definite polynomial Lyapunov function V such that $V(x) - V(Ax)$ is positive definite and sos. Then, V is sos.*

Proof. Consider the polynomial $V(x) - V(Ax)$ that is sos by assumption. If we replace x by Ax in this polynomial, we conclude that the polynomial $V(Ax) - V(A^2x)$ is also sos. Hence, by adding these two sos polynomials, we get that $V(x) - V(A^2x)$ is sos. This procedure can obviously be repeated to infer that for any integer $k \geq 1$, the polynomial

$$V(x) - V(A^kx) \tag{4.30}$$

is sos. Since by assumption V and $V(x) - V(Ax)$ are positive definite, the linear system must be GAS, and hence A^k converges to the zero matrix as $k \to \infty$. Observe that for all k, the polynomials in (4.30) have degree equal to the degree of V, and that the coefficients of $V(x) - V(A^kx)$ converge to the coefficients of V as $k \to \infty$. Since for a fixed degree and dimension the cone of sos polynomials is closed [141], it follows that V is sos. □

Similarly, in continuous time, we have the following proposition.

Proposition 4.15. *Consider the linear dynamical system $\dot{x} = Ax$ in continuous time. Suppose there exists a positive definite polynomial Lyapunov function V such that $-\dot{V} = -\langle \nabla V(x), Ax \rangle$ is positive definite and sos. Then, V is sos.*

Proof. The value of the polynomial V along the trajectories of the dynamical system satisfies the relation

$$V(x(t)) = V(x(0)) + \int_o^t \dot{V}(x(\tau))d_\tau.$$

Since the assumptions imply that the system is GAS, $V(x(t)) \to 0$ as t goes to infinity. (Here, we are assuming, without loss of generality, that V vanishes at the origin.) By evaluating the above equation at $t = \infty$, rearranging terms, and substituting $e^{A\tau}x$ for the solution of the linear system at time τ starting at initial condition x, we obtain

$$V(x) = \int_0^\infty -\dot{V}(e^{A\tau}x)d_\tau.$$

By assumption, $-\dot{V}$ is sos and therefore for any value of τ, the integrand $-\dot{V}(e^{A\tau}x)$ is an sos polynomial. Since converging integrals of sos polynomials are sos, it follows that V is sos. □

Remark 4.5.1. The previous proposition does not hold if the system is not linear. For example, consider any positive form V that is not a sum of squares and define a dynamical system by $\dot{x} = -\nabla V(x)$. In this case, both V and $-\dot{V} = ||\nabla V(x)||^2$ are positive definite and $-\dot{V}$ is sos, though V is not sos.

4.6 Some open questions

Some open questions related to the problems studied in this chapter are the following. Regarding complexity, of course the interesting problem is to formally answer the questions of Arnold on undecidability of determining stability for polynomial vector fields. Regarding existence of polynomial Lyapunov functions, Mark Tobenkin asked whether a globally exponentially stable polynomial vector field admits a polynomial Lyapunov function. Our counterexample in Section 4.3, though GAS and locally exponentially stable, is not globally exponentially stable because of exponential growth rates in the large. The counterexample of Bacciotti and Rosier in [22] is not even locally exponentially stable. Another future direction is to prove that GAS homogeneous polynomial vector fields admit homogeneous polynomial Lyapunov functions. This, together with Theorem 4.8, would imply that asymptotic stability of homogeneous polynomial systems can always be decided via sum of squares programming. Also, it is not clear to us whether the assumption of homogeneity and planarity can be removed from Theorems 4.8 and 4.10 on existence of sos Lyapunov functions. Finally, another research direction would be to obtain upper bounds on the degree of polynomial or sos polynomial Lyapunov functions. Some degree bounds are known for Lyapunov analysis of locally exponentially stable systems [127], but they depend on uncomputable properties of the solution such as convergence rate. Degree bounds on Positivstellensatz result of the type in Theorems 4.7 and 4.9 are known, but typically exponential in size and not very encouraging for practical purposes.

Chapter 5

Joint Spectral Radius and Path-Complete Graph Lyapunov Functions

In this chapter, we introduce the framework of path-complete graph Lyapunov functions for analysis of switched systems. The methodology is presented in the context of approximation of the joint spectral radius. The content of this chapter is based on an extended version of the work in [3].

5.1 Introduction

Given a finite set of square matrices $\mathcal{A} := \{A_1, ..., A_m\}$, their *joint spectral radius* $\rho(\mathcal{A})$ is defined as

$$\rho(\mathcal{A}) = \lim_{k \to \infty} \max_{\sigma \in \{1,...,m\}^k} \left\| A_{\sigma_k} ... A_{\sigma_2} A_{\sigma_1} \right\|^{1/k}, \tag{5.1}$$

where the quantity $\rho(\mathcal{A})$ is independent of the norm used in (5.1). The joint spectral radius (JSR) is a natural generalization of the spectral radius of a single square matrix and it characterizes the maximal growth rate that can be obtained by taking products, of arbitrary length, of all possible permutations of $A_1, ..., A_m$. This concept was introduced by Rota and Strang [147] in the early 60s and has since been the subject of extensive research within the engineering and the mathematics communities alike. Aside from a wealth of fascinating mathematical questions that arise from the JSR, the notion emerges in many areas of application such as stability of switched linear dynamical systems, computation of the capacity of codes, continuity of wavelet functions, convergence of consensus algorithms, trackability of graphs, and many others. See [85] and references therein for a recent survey of the theory and applications of the JSR.

Motivated by the abundance of applications, there has been much work on efficient computation of the joint spectral radius; see e.g. [32], [31], [122], and references therein. Unfortunately, the negative results in the literature certainly restrict the horizon of possibilities. In [35], Blondel and Tsitsiklis prove that even when the set $\mathcal{A}$ consists of only two matrices, the question of testing whether $\rho(\mathcal{A}) \leq 1$ is undecidable. They also show that unless P=NP, one cannot compute an approximation $\hat{\rho}$ of ρ that satisfies $|\hat{\rho}-\rho| \leq \epsilon\rho$, in a number of steps polynomial in the bit size of $\mathcal{A}$ and the bit size of ϵ [161]. It is not difficult to show that the spectral radius of any finite product of length k raised to the power of $1/k$ gives a lower bound on ρ [85]. However, for reasons that we explain next, our focus will be on computing upper bounds for ρ.

There is an attractive connection between the joint spectral radius and the stability properties of an arbitrary switched linear system; i.e., dynamical systems of the form

$$x_{k+1} = A_{\sigma(k)} x_k, \tag{5.2}$$

where $\sigma : \mathbb{Z} \to \{1, ..., m\}$ is a map from the set of integers to the set of indices. It is well-known that $\rho < 1$ if and only if system (5.2) is *absolutely asymptotically stable* (AAS), that is, (globally) asymptotically stable for all switching sequences. Moreover, it is known [95] that absolute asymptotic stability of (5.2) is equivalent to absolute asymptotic stability of the linear difference inclusion

$$x_{k+1} \in \mathrm{co}\mathcal{A}\ x_k, \tag{5.3}$$

where co$\mathcal{A}$ here denotes the convex hull of the set $\mathcal{A}$. Therefore, any method for obtaining upper bounds on the joint spectral radius provides sufficient conditions for stability of systems of type (5.2) or (5.3). Conversely, if we can prove absolute asymptotic stability of (5.2) or (5.3) for the set $\mathcal{A}_\gamma := \{\gamma A_1, \ldots, \gamma A_m\}$ for some positive scalar γ, then we get an upper bound of $\frac{1}{\gamma}$ on $\rho(\mathcal{A})$. (This follows from the scaling property of the JSR: $\rho(\mathcal{A}_\gamma) = \gamma\rho(\mathcal{A})$.) One advantage of working with the notion of the joint spectral radius is that it gives a way of rigorously quantifying the performance guarantee of different techniques for stability analysis of systems (5.2) or (5.3).

Perhaps the most well-established technique for proving stability of switched systems is the use of a *common (or simultaneous) Lyapunov function*. The idea here is that if there is a continuous, positive, and homogeneous (Lyapunov) function $V(x) : \mathbb{R}^n \to \mathbb{R}$ that for some $\gamma > 1$ satisfies

$$V(\gamma A_i x) \leq V(x) \quad \forall i = 1, \ldots, m, \ \ \forall x \in \mathbb{R}^n, \tag{5.4}$$

(i.e., $V(x)$ decreases no matter which matrix is applied), then the system in (5.2) (or in (5.3)) is AAS. Conversely, it is known that if the system is AAS, then there exists a *convex* common Lyapunov function (in fact a norm); see e.g. [85,

p. 24]. However, this function is not in general finitely constructable. A popular approach has been to try to approximate this function by a class of functions that we can efficiently search for using convex optimization and in particular semidefinite programming. As we mentioned in our introductory chapters, semidefinite programs (SDPs) can be solved with arbitrary accuracy in polynomial time and lead to efficient computational methods for approximation of the JSR. As an example, if we take the Lyapunov function to be quadratic (i.e., $V(x) = x^T P x$), then the search for such a Lyapunov function can be formulated as the following SDP:

$$\begin{array}{rcl} P & \succ & 0 \\ \gamma^2 A_i^T P A_i & \preceq & P \quad \forall i = 1, \ldots, m. \end{array} \tag{5.5}$$

The quality of approximation of common quadratic Lyapunov functions is a well-studied topic. In particular, it is known [32] that the estimate $\hat{\rho}_{\mathcal{V}^2}$ obtained by this method[1] satisfies

$$\frac{1}{\sqrt{n}} \hat{\rho}_{\mathcal{V}^2}(\mathcal{A}) \leq \rho(\mathcal{A}) \leq \hat{\rho}_{\mathcal{V}^2}(\mathcal{A}), \tag{5.6}$$

where n is the dimension of the matrices. This bound is a direct consequence of John's ellipsoid theorem and is known to be tight [13].

In [122], the use of sum of squares (sos) polynomial Lyapunov functions of degree $2d$ was proposed as a common Lyapunov function for the switched system in (5.2). As we know, the search for such a Lyapunov function can again be formulated as a semidefinite program. This method does considerably better than a common quadratic Lyapunov function in practice and its estimate $\hat{\rho}_{\mathcal{V}^{SOS,2d}}$ satisfies the bound

$$\frac{1}{\sqrt[2d]{\eta}} \hat{\rho}_{\mathcal{V}^{SOS,2d}}(\mathcal{A}) \leq \rho(\mathcal{A}) \leq \hat{\rho}_{\mathcal{V}^{SOS,2d}}(\mathcal{A}), \tag{5.7}$$

where $\eta = \min\{m, \binom{n+d-1}{d}\}$. Furthermore, as the degree $2d$ goes to infinity, the estimate $\hat{\rho}_{\mathcal{V}^{SOS,2d}}$ converges to the true value of ρ [122]. The semidefinite programming based methods for approximation of the JSR have been recently generalized and put in the framework of conic programming [134].

5.1.1 Contributions and organization of this chapter

It is natural to ask whether one can develop better approximation schemes for the joint spectral radius by using multiple Lyapunov functions as opposed to requiring

[1] The estimate $\hat{\rho}_{\mathcal{V}^2}$ is the reciprocal of the largest γ that satisfies (5.5) and can be found by bisection.

simultaneous contractibility of a single Lyapunov function with respect to all the matrices. More concretely, our goal is to understand how we can write inequalities among, say, k different Lyapunov functions $V_1(x), \ldots, V_k(x)$ that imply absolute asymptotic stability of (5.2) and can be checked via semidefinite programming.

The general idea of using several Lyapunov functions for analysis of switched systems is a very natural one and has already appeared in the literature (although to our knowledge not in the context of the approximation of the JSR); see e.g. [136], [39], [81], [80], [64]. Perhaps one of the earliest references is the work on "piecewise quadratic Lyapunov functions" in [136]. However, this work is in the different framework of state dependent switching, where the dynamics switches depending on which region of the space the trajectory is traversing (as opposed to arbitrary switching). In this setting, there is a natural way of using several Lyapunov functions: assign one Lyapunov function per region and "glue them together". Closer to our setting, there is a body of work in the literature that gives sufficient conditions for existence of piecewise Lyapunov functions of the type $\max\{x^TP_1x, \ldots, x^TP_kx\}$, $\min\{x^TP_1x, \ldots, x^TP_kx\}$, and $\text{conv}\{x^TP_1x, \ldots, x^TP_kx\}$, i.e, the pointwise maximum, the pointwise minimum, and the convex envelope of a set of quadratic functions [81], [80], [64], [82]. These works are mostly concerned with analysis of linear differential inclusions in continuous time, but they have obvious discrete time counterparts. The main drawback of these methods is that in their greatest generality, they involve solving bilinear matrix inequalities, which are non-convex and in general NP-hard. One therefore has to turn to heuristics, which have no performance guarantees and their computation time quickly becomes prohibitive when the dimension of the system increases. Moreover, all of these methods solely provide sufficient conditions for stability with no performance guarantees.

There are several unanswered questions that in our view deserve a more thorough study: (i) With a focus on conditions that are amenable to convex optimization, what are the different ways to write a set of inequalities among k Lyapunov functions that imply absolute asymptotic stability of (5.2)? Can we give a unifying framework that includes the previously proposed Lyapunov functions and perhaps also introduces new ones? (ii) Among the different sets of inequalities that imply stability, can we identify some that are less conservative than some other? (iii) The available methods on piecewise Lyapunov functions solely provide sufficient conditions for stability with no guarantee on their performance. Can we give converse theorems that guarantee the existence of a feasible solution to our search for a given accuracy?

The contributions of this chapter to these questions are as follows. We propose a unifying framework based on a representation of Lyapunov inequalities with labeled graphs and by making some connections with basic concepts in automata theory. This is done in Section 5.2, where we define the notion of a path-complete graph (Definition 5.2) and prove that any such graph provides an approximation scheme for the JSR (Theorem 5.4). In Section 5.3, we give examples of families of path-complete graphs and show that many of the previously proposed techniques come from particular classes of simple path-complete graphs (e.g., Corollary 5.8, Corollary 5.9, and Remark 5.3.2). In Section 5.4, we characterize all the path-complete graphs with two nodes for the analysis of the JSR of two matrices. We determine how the approximations obtained from all of these graphs compare (Proposition 5.12). In Section 5.5, we study in more depth the approximation properties of a particular pair of "dual" path-complete graphs that seem to perform very well in practice. Subsection 5.5.1 contains more general results about duality within path-complete graphs and its connection to transposition of matrices (Theorem 5.13). Subsection 5.5.2 gives an approximation guarantee for the graphs studied in Section 5.5 (Theorem 5.16), and Subsection 5.5.3 contains some numerical examples. In Section 5.6, we prove a converse theorem for the method of max-of-quadratics Lyapunov functions (Theorem 5.17) and an approximation guarantee for a new class of methods for proving stability of switched systems (Theorem 5.18). Finally, some concluding remarks and future directions are presented in Section 5.7.

■ 5.2 Path-complete graphs and the joint spectral radius

In what follows, we will think of the set of matrices $\mathcal{A} := \{A_1, ..., A_m\}$ as a finite alphabet and we will often refer to a finite product of matrices from this set as a *word*. We denote the set of all words $A_{i_t} \dots A_{i_1}$ of length t by $\mathcal{A}^t$. Contrary to the standard convention in automata theory, our convention is to read a word from right to left. This is in accordance with the order of matrix multiplication. The set of all finite words is denoted by $\mathcal{A}^*$; i.e., $\mathcal{A}^* = \bigcup_{t \in \mathbb{Z}^+} \mathcal{A}^t$.

The basic idea behind our framework is to represent through a graph all the possible occurrences of products that can appear in a run of the dynamical system in (5.2), and assert via some Lyapunov inequalities that no matter what occurrence appears, the product must remain stable. A convenient way of representing these Lyapunov inequalities is via a directed labeled graph $G(N, E)$. Each node of this graph is associated with a (continuous, positive definite, and homogeneous) Lyapunov function $V_i(x) : \mathbb{R}^n \to \mathbb{R}$, and each edge is labeled by a finite product of matrices, i.e., by a word from the set $\mathcal{A}^*$. As illustrated in Figure 5.1, given two nodes with Lyapunov functions $V_i(x)$ and $V_j(x)$ and an edge going from node i to node j labeled with the matrix A_l, we write the Lyapunov inequality:

$$V_j(A_l x) \leq V_i(x) \quad \forall x \in \mathbb{R}^n. \tag{5.8}$$

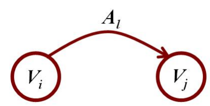


Figure 5.1. Graphical representation of Lyapunov inequalities. The edge in the graph above corresponds to the Lyapunov inequality $V_j(A_l x) \leq V_i(x)$. Here, A_l can be a single matrix from $\mathcal{A}$ or a finite product of matrices from $\mathcal{A}$.

The problem that we are interested in is to understand which sets of Lyapunov inequalities imply stability of the switched system in (5.2). We will answer this question based on the corresponding graph.

For reasons that will become clear shortly, we would like to reduce graphs whose edges have arbitrary labels from the set $\mathcal{A}^*$ to graphs whose edges have labels from the set $\mathcal{A}$, i.e, labels of length one. This is explained next.

Definition 5.1. *Given a labeled directed graph $G(N, E)$, we define its* expanded graph *$G^e(N^e, E^e)$ as the outcome of the following procedure. For every edge $(i, j) \in E$ with label $A_{ik} \ldots A_{i1} \in \mathcal{A}^k$, where $k > 1$, we remove the edge (i, j) and replace it with k new edges $(s_q, s_{q+1}) \in E^e \setminus E : \; q \in \{0, \ldots, k-1\}$, where $s_0 = i$ and $s_k = j$.*[2] *(These new edges go from node i through $k - 1$ newly added nodes $s_1, \ldots, s_{k-1}$ and then to node j.) We then label the new edges $(i, s_1), \ldots, (s_q, s_{q+1}), \ldots, (s_{k-1}, j)$ with $A_{i1}, \ldots, A_{ik}$ respectively.*

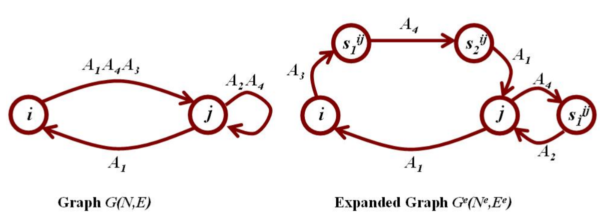


Figure 5.2. Graph expansion: edges with labels of length more than one are broken into new edges with labels of length one.

An example of a graph and its expansion is given in Figure 5.2. Note that if a graph has only labels of length one, then its expanded graph equals itself. The next definition is central to our development.

Definition 5.2. *Given a directed graph $G(N, E)$ whose edges are labeled with words from the set $\mathcal{A}^*$, we say that the graph is* path-complete*, if for all finite*

[2] It is understood that the node index s_q depends on the original nodes i and j. To keep the notation simple we write s_q instead of s_q^{ij}.

words $A_{\sigma_k} \dots A_{\sigma_1}$ of any length k (i.e., for all words in $\mathcal{A}^*$), there is a directed path in its expanded graph $G^e(N^e, E^e)$ such that the labels on the edges of this path are the labels A_{σ_1} up to A_{σ_k}.

In Figure 5.3, we present seven path-complete graphs on the alphabet $\mathcal{A} = \{A_1, A_2\}$. The fact that these graphs are path-complete is easy to see for graphs H_1, H_2, G_3, and G_4, but perhaps not so obvious for graphs H_3, G_1, and G_2. One way to check if a graph is path-complete is to think of it as a finite automaton by introducing an auxiliary start node (state) with free transitions to every node and by making all the other nodes be accepting states. Then, there are well-known algorithms (see e.g. [78, Chap. 4]) that check whether the language accepted by an automaton is $\mathcal{A}^*$, which is equivalent to the graph being path-complete. At least for the cases where the automata are deterministic (i.e., when all outgoing edges from any node have different labels), these algorithms are very efficient and have running time of only $O(|N|^2)$. Similar algorithms exist in the symbolic dynamics literature; see e.g. [96, Chap. 3]. Our interest in path-complete graphs stems from the Theorem 5.4 below that establishes that any such graph gives a method for approximation of the JSR. We introduce one last definition before we state this theorem.

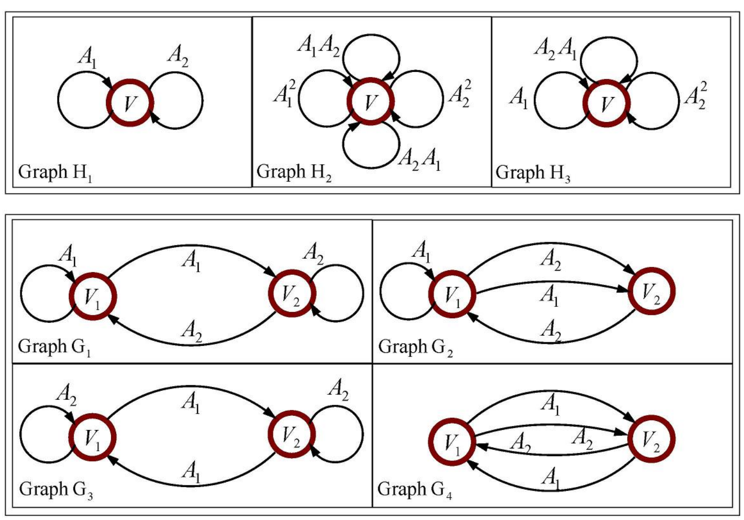


Figure 5.3. Examples of path-complete graphs for the alphabet $\{A_1, A_2\}$. If Lyapunov functions satisfying the inequalities associated with any of these graphs are found, then we get an upper bound of unity on $\rho(A_1, A_2)$.

Definition 5.3. *Let $\mathcal{A} = \{A_1, \dots, A_m\}$ be a set of matrices. Given a path-complete graph $G(N, E)$ and $|N|$ functions $V_i(x)$, we say that $\{V_i(x) | i = 1, \dots, |N|\}$*

is a graph Lyapunov function (GLF) associated with $G\,(N,E)$ *if*

$$V_j\,(L\,((i,j))\,x) \leq V_i\,(x) \qquad \forall x \in \mathbb{R}^n, \;\; \forall\;(i,j) \in E,$$

where $L\,((i,j)) \in \mathcal{A}^*$ *is the label associated with edge* $(i,j) \in E$ *going from node* i *to node* j.

Theorem 5.4. *Consider a finite set of matrices* $\mathcal{A} = \{A_1, \ldots, A_m\}$. *For a scalar* $\gamma > 0$, *let* $\mathcal{A}_\gamma := \{\gamma A_1, \ldots, \gamma A_m\}$. *Let* $G(N,E)$ *be a path-complete graph whose edges are labeled with words from* $\mathcal{A}_\gamma^*$. *If there exist positive, continuous, and homogeneous*[3] *functions* $V_i(x)$, *one per node of the graph, such that* $\{V_i(x) \mid i = 1, \ldots, |N|\}$ *is a graph Lyapunov function associated with* $G(N,E)$, *then* $\rho(\mathcal{A}) \leq \frac{1}{\gamma}$.

Proof. We will first prove the claim for the special case where the edge labels of $G(N,E)$ belong to $\mathcal{A}_\gamma$ and therefore $G(N,E) = G^e(N^e, E^e)$. The general case will be reduced to this case afterwards. Let d be the degree of homogeneity of the Lyapunov functions $V_i(x)$, i.e., $V_i(\lambda x) = \lambda^d V_i(x)$ for all $\lambda \in \mathbb{R}$. (The actual value of d is irrelevant.) By positivity, continuity, and homogeneity of $V_i(x)$, there exist scalars α_i and β_i with $0 < \alpha_i \leq \beta_i$ for $i = 1, \ldots, |N|$, such that

$$\alpha_i ||x||^d \leq V_i(x) \leq \beta_i ||x||^d, \tag{5.9}$$

for all $x \in \mathbb{R}^n$ and for all $i = 1, \ldots, |N|$, where $||x||$ here denotes the Euclidean norm of x. Let

$$\xi = \max_{i,j \in \{1,\ldots,|N|\}^2} \frac{\beta_i}{\alpha_j}. \tag{5.10}$$

Now consider an arbitrary product $A_{\sigma_k} \ldots A_{\sigma_1}$ of length k. Because the graph is path-complete, there will be a directed path corresponding to this product that consists of k edges, and goes from some node i to some node j. If we write the chain of k Lyapunov inequalities associated with these edges (cf. Figure 5.1), then we get

$$V_j(\gamma^k A_{\sigma_k} \ldots A_{\sigma_1} x) \leq V_i(x),$$

which by homogeneity of the Lyapunov functions can be rearranged to

$$\left(\frac{V_j(A_{\sigma_k} \ldots A_{\sigma_1} x)}{V_i(x)} \right)^{\frac{1}{d}} \leq \frac{1}{\gamma^k}. \tag{5.11}$$

[3] The requirement of homogeneity can be replaced by radial unboundedness which is implied by homogeneity and positivity. However, since the dynamical system in (5.2) is homogeneous, there is no conservatism in asking $V_i(x)$ to be homogeneous.

We can now bound the spectral norm of $A_{\sigma_k}\dots A_{\sigma_1}$ as follows:

$$\begin{aligned}||A_{\sigma_k}\dots A_{\sigma_1}|| &\leq \max_x \frac{||A_{\sigma_k}\dots A_{\sigma_1}x||}{||x||}\\ &\leq \left(\frac{\beta_i}{\alpha_j}\right)^{\frac{1}{d}} \max_x \frac{V_j^{\frac{1}{d}}(A_{\sigma_k}\dots A_{\sigma_1}x)}{V_i^{\frac{1}{d}}(x)}\\ &\leq \left(\frac{\beta_i}{\alpha_j}\right)^{\frac{1}{d}} \frac{1}{\gamma^k}\\ &\leq \xi^{\frac{1}{d}}\frac{1}{\gamma^k},\end{aligned}$$

where the last three inequalities follow from (5.9), (5.11), and (5.10) respectively. From the definition of the JSR in (5.1), after taking the k-th root and the limit $k\to\infty$, we get that $\rho(\mathcal{A})\leq\frac{1}{\gamma}$ and the claim is established.

Now consider the case where at least one edge of $G(N,E)$ has a label of length more than one and hence $G^e(N^e,E^e)\neq G(N,E)$. We will start with the Lyapunov functions $V_i(x)$ assigned to the nodes of $G(N,E)$ and from them we will explicitly construct $|N^e|$ Lyapunov functions for the nodes of $G^e(N^e,E^e)$ that satisfy the Lyapunov inequalities associated to the edges in E^e. Once this is done, in view of our preceding argument and the fact that the edges of $G^e(N^e,E^e)$ have labels of length one by definition, the proof will be completed.

For $j\in N^e$, let us denote the new Lyapunov functions by $V_j^e(x)$. We give the construction for the case where $|N^e|=|N|+1$. The result for the general case follows by iterating this simple construction. Let $s\in N^e\backslash N$ be the added node in the expanded graph, and $q,r\in N$ be such that $(s,q)\in E^e$ and $(r,s)\in E^e$ with A_{sq} and A_{rs} as the corresponding labels respectively. Define

$$V_j^e(x)=\begin{cases} V_j(x), & \text{if } \ j\in N\\ V_q(A_{sq}x), & \text{if } \ j=s.\end{cases} \tag{5.12}$$

By construction, r and q, and subsequently, A_{sq} and A_{rs} are uniquely defined and hence, $\left\{V_j^e(x)\ |\ j\in N^e\right\}$ is well defined. We only need to show that

$$V_q(A_{sq}x)\leq V_s^e(x) \tag{5.13}$$
$$V_s^e(A_{rs}x)\leq V_r(x). \tag{5.14}$$

Inequality (5.13) follows trivially from (5.12). Furthermore, it follows from (5.12) that

$$\begin{aligned}V_s^e(A_{rs}x) &= V_q(A_{sq}A_{rs}x)\\ &\leq V_r(x),\end{aligned}$$

where the inequality follows from the fact that for $i \in N$, the functions $V_i(x)$ satisfy the Lyapunov inequalities of the edges of $G(N, E)$. □

Remark 5.2.1. If the matrix A_{sq} is not invertible, the extended function $V_j^e(x)$ as defined in (5.12) will only be positive semidefinite. However, since our goal is to approximate the JSR, we will never be concerned with invertibility of the matrices in $\mathcal{A}$. Indeed, since the JSR is continuous in the entries of the matrices [85], we can always perturb the matrices slightly to make them invertible without changing the JSR by much. In particular, for any $\alpha > 0$, there exist $0 < \varepsilon, \delta < \alpha$ such that

$$\hat{A}_{sq} = \frac{A_{sq} + \delta I}{1 + \varepsilon}$$

is invertible and $(5.12)-(5.14)$ are satisfied with $A_{sq} = \hat{A}_{sq}$.

To understand the generality of the framework of "path-complete graph Lyapunov funcitons" more clearly, let us revisit the path-complete graphs in Figure 5.3 for the study of the case where the set $\mathcal{A} = \{A_1, A_2\}$ consists of only two matrices. For all of these graphs if our choice for the Lyapunov functions $V(x)$ or $V_1(x)$ and $V_2(x)$ are quadratic functions or sum of squares polynomial functions, then we can formulate the well-established semidefinite programs that search for these candidate Lyapunov functions.

Graph H_1, which is clearly the simplest possible one, corresponds to the well-known common Lyapunov function approach. Graph H_2 is a common Lyapunov function applied to all products of length two. This graph also obviously implies stability.[4] But graph H_3 tells us that if we find a Lyapunov function that decreases whenever A_1, A_2^2, and A_2A_1 are applied (but with no requirement when A_1A_2 is applied), then we still get stability. This is a priori not obvious and we believe this approach has not appeared in the literature before. Graph H_3 is also an example that explains why we needed the expansion process. Note that for the unexpanded graph, there is no path for any word of the form $(A_1A_2)^k$ or of the form A_2^{2k-1}, for any $k \in \mathbb{N}$. However, one can check that in the expanded graph of graph H_3, there is a path for every finite word, and this in turn allows us to conclude stability from the Lyapunov inequalities of graph H_3.

The remaining graphs in Figure 5.3 which all have two nodes and four edges with labels of length one have a connection to the method of min-of-quadratics or max-of-quadratics Lyapunov functions [81], [80], [64], [82]. If Lyapunov inequalities associated with any of these four graphs are satisfied, then either $\min\{V_1(x), V_2(x)\}$ or $\max\{V_1(x), V_2(x)\}$ or both serve as a common Lyapunov

[4]By slight abuse of terminology, we say that a graph implies stability meaning that the associated Lyapunov inequalities imply stability.

function for the switched system. In the next section, we assert these facts in a more general setting (Corollaries 5.8 and 5.9) and show that these graphs in some sense belong to "simplest" families of path-complete graphs.

■ 5.3 Duality and examples of families of path-complete graphs

Now that we have shown that *any* path-complete graph introduces a method for proving stability of switched systems, our next focus is naturally on showing how one can produce graphs that are path-complete. Before we proceed to some basic constructions of such graphs, let us define a notion of *duality* among graphs which essentially doubles the number of path-complete graphs that we can generate.

Definition 5.5. *Given a directed graph $G(N,E)$ whose edges are labeled from the words in $\mathcal{A}^*$, we define its* dual graph *$G'(N,E')$ to be the graph obtained by reversing the direction of the edges of G, and changing the labels $A_{\sigma_k}\dots A_{\sigma_1}$ of every edge of G to its reversed version $A_{\sigma_1}\dots A_{\sigma_k}$.*

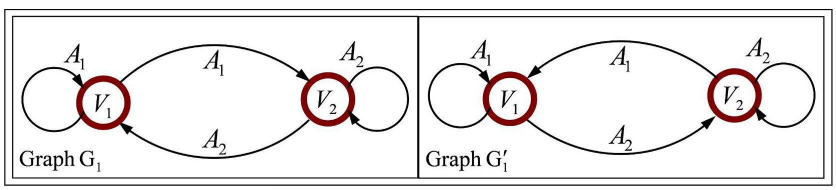


Figure 5.4. An example of a pair of dual graphs.

An example of a pair of dual graphs with labels of length one is given in Figure 5.4. The following theorem relates dual graphs and path-completeness.

Theorem 5.6. *If a graph $G(N,E)$ is path-complete, then its dual graph $G'(N,E')$ is also path-complete.*

Proof. Consider an arbitrary finite word $A_{i_k}\dots A_{i_1}$. By definition of what it means for a graph to be path-complete, our task is to show that there exists a path corresponding to this word in the expanded graph of the dual graph G'. It is easy to see that the expanded graph of the dual graph of G is the same as the dual graph of the expanded graph of G; i.e, $G'^e(N^e,E'^e) = G^{e^{'}}(N^e,E^{e^{'}})$. Therefore, we show a path for $A_{i_k}\dots A_{i_1}$ in $G^{e^{'}}$. Consider the reversed word $A_{i_i}\dots A_{i_k}$. Since G is path-complete, there is a path corresponding to this reversed word in G^e. Now if we just trace this path backwards, we get exactly a path for the original word $A_{i_k}\dots A_{i_1}$ in $G^{e^{'}}$. This completes the proof. □

The next proposition offers a very simple construction for obtaining a large family of path-complete graphs with labels of length one.

Proposition 5.7. *A graph having any of the two properties below is path-complete.*
Property (i): every node has outgoing edges with all the labels in $\mathcal{A}$.
Property (ii): every node has incoming edges with all the labels in $\mathcal{A}$.

Proof. If a graph has Property (i), then it is obviously path-complete. If a graph has Property (ii), then its dual has Property (i) and therefore by Theorem 5.6 it is path-complete. □

Examples of path-complete graphs that fall in the category of this proposition include graphs G_1, G_2, G_3, and G_4 in Figure 5.3 and all of their dual graphs. By combining the previous proposition with Theorem 5.4, we obtain the following two simple corollaries which unify several linear matrix inequalities (LMIs) that have been previously proposed in the literature. These corollaries also provide a link to min/max-of-quadratics Lyapunov functions. Different special cases of these LMIs have appeared in [81], [80], [64], [82], [93], [53]. Note that the framework of path-complete graph Lyapunov functions makes the proof of the fact that these LMIs imply stability immediate.

Corollary 5.8. *Consider a set of m matrices and the switched linear system in (5.2) or (5.3). If there exist k positive definite matrices P_j such that*

$$\begin{aligned} &\forall (i,k) \in \{1,\ldots,m\}^2,\ \exists j \in \{1,\ldots,m\} \\ &\qquad \textit{such that} \qquad \gamma^2 A_i^T P_j A_i \preceq P_k, \end{aligned} \tag{5.15}$$

for some $\gamma > 1$, then the system is absolutely asymptotically stable. Moreover, the pointwise minimum

$$\min\{x^T P_1 x, \ldots, x^T P_k x\}$$

of the quadratic functions serves as a common Lyapunov function.

Proof. The inequalities in (5.15) imply that every node of the associated graph has outgoing edges labeled with all the different m matrices. Therefore, by Proposition 5.7 the graph is path-complete, and by Theorem 5.4 this implies absolute asymptotic stability. The proof that the pointwise minimum of the quadratics is a common Lyapunov function is easy and left to the reader. □

Corollary 5.9. *Consider a set of m matrices and the switched linear system in (5.2) or (5.3). If there exist k positive definite matrices P_j such that*

$$\begin{aligned} &\forall (i,j) \in \{1,\ldots,m\}^2,\ \exists k \in \{1,\ldots,m\} \\ &\qquad \textit{such that} \qquad \gamma^2 A_i^T P_j A_i \preceq P_k, \end{aligned} \tag{5.16}$$

for some $\gamma > 1$*, then the system is absolutely asymptotically stable. Moreover, the pointwise maximum*

$$\max\{x^T P_1 x, \ldots, x^T P_k x\}$$

of the quadratic functions serves as a common Lyapunov function.

Proof. The inequalities in (5.16) imply that every node of the associated graph has incoming edges labeled with all the different m matrices. Therefore, by Proposition 5.7 the graph is path-complete and the proof of absolute asymptotic stability then follows. The proof that the pointwise maximum of the quadratics is a common Lyapunov function is again left to the reader. □

Remark 5.3.1. The linear matrix inequalities in (5.15) and (5.16) are (convex) sufficient conditions for existence of min-of-quadratics or max-of-quadratics Lyapunov functions. The converse is not true. The works in [81], [80], [64], [82] have additional multipliers in (5.15) and (5.16) that make the inequalities non-convex but when solved with a heuristic method contain a larger family of min-of-quadratics and max-of-quadratics Lyapunov functions. Even if the non-convex inequalities with multipliers could be solved exactly, except for special cases where the $\mathcal{S}$-procedure is exact (e.g., the case of two quadratic functions), these methods still do not completely characterize min-of-quadratics and max-of-quadratics functions.

Remark 5.3.2. The work in [93] on "path-dependent quadratic Lyapunov functions" and the work in [53] on "parameter dependent Lyapunov functions"–when specialized to the analysis of arbitrary switched linear systems–are special cases of Corollary 5.8 and 5.9 respectively. This observation makes a connection between these techniques and min/max-of-quadratics Lyapunov functions which is not established in [93], [53]. It is also interesting to note that the path-complete graph corresponding to the LMIs proposed in [93] (see Theorem 9 there) is the well-known De Bruijn graph [67].

The set of path-complete graphs is much broader than the set of simple family of graphs constructed in Proposition 5.7. Indeed, there are many graphs that are path-complete without having outgoing (or incoming) edges with all the labels on every node; see e.g. graph H_4^e in Figure 5.5. This in turn means that there are several more sophisticated Lyapunov inequalities that we can explore for proving stability of switched systems. Below, we give one particular example of such "non-obvious" inequalities for the case of switching between two matrices.

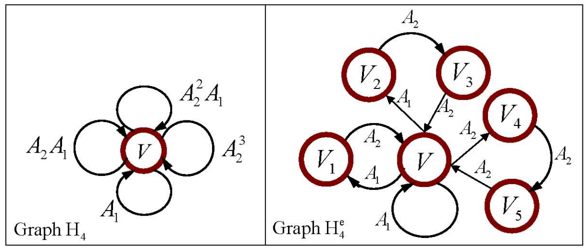


Figure 5.5. The path-complete graphs corresponding to Proposition 5.10.

Proposition 5.10. *Consider the set* $\mathcal{A} = \{A_1, A_2\}$ *and the switched linear system in (5.2) or (5.3). If there exist a positive definite matrix* P *such that*

$$\begin{aligned} \gamma^2 A_1^T P A_1 &\preceq P, \\ \gamma^4 (A_2A_1)^T P (A_2A_1) &\preceq P, \\ \gamma^6 (A_2^2A_1)^T P (A_2^2A_1) &\preceq P, \\ \gamma^6 A_2^{3^T} P A_2^3 &\preceq P, \end{aligned}$$

for some $\gamma > 1$*, then the system is absolutely asymptotically stable.*

Proof. The graph H_4 associated with the LMIs above and its expanded version H_4^e are drawn in Figure 5.5. We leave it as an exercise for the reader to show (e.g. by induction on the length of the word) that there is path for every finite word in H_4^e. Therefore, H_4 is path-complete and in view of Theorem 5.4 the claim is established. □

Remark 5.3.3. Proposition 5.10 can be generalized as follows: If a single Lyapunov function decreases with respect to the matrix products

$$\{A_1, A_2A_1, A_2^2A_1, \ldots, A_2^{k-1}A_1, A_2^k\}$$

for some integer $k \geq 1$, then the arbitrary switched system consisting of the two matrices A_1 and A_2 is absolutely asymptotically stable. We omit the proof of this generalization due to space limitations. We will later prove (Theorem 5.18) a bound for the quality of approximation of path-complete graphs of this type, where a common Lyapunov function is required to decrease with respect to products of different lengths.

When we have so many different ways of imposing conditions for stability, it is natural to ask which ones are better. The answer clearly depends on the combinatorial structure of the graphs and does not seem to be easy in general. Nevertheless, in the next section, we compare the performance of all path-complete graphs with two nodes for analysis of switched systems with two matrices. The connections between the bounds obtained from these graphs are not always obvious. For example, we will see that the graphs H_1, G_3, and G_4 always give the same bound on the joint spectral radius; i.e, one graph will succeed in proving stability if and only if the other will. So, there is no point in increasing the number of decision variables and the number of constraints and impose G_3 or G_4 in place of H_1. The same is true for the graphs in H_3 and G_2, which makes graph H_3 preferable to graph G_2. (See Proposition 5.12.)

5.4 Path-complete graphs with two nodes

In this section, we characterize the set of all path-complete graphs consisting of two nodes, an alphabet set $\mathcal{A} = \{A_1, A_2\}$, and edge labels of unit length. We will elaborate on the set of all admissible topologies arising in this setup and compare the performance—in the sense of conservatism of the ensuing analysis—of different path-complete graph topologies.

5.4.1 The set of path-complete graphs

The next lemma establishes that for thorough analysis of the case of two matrices and two nodes, we only need to examine graphs with four or fewer edges.

Lemma 5.11. *Let $G(\{1,2\}, E)$ be a path-complete graph with labels of length one for $\mathcal{A} = \{A_1, A_2\}$. Let $\{V_1, V_2\}$ be a graph Lyapunov function for G. If $|E| > 4$, then, either*

(i) there exists $\hat{e} \in E$ such that $G(\{1,2\}, E \backslash \hat{e})$ is a path-complete graph,
or
(ii) either V_1 or V_2 or both are common Lyapunov functions for $\mathcal{A}$.

Proof. If $|E| > 4$, then at least one node has three or more outgoing edges. Without loss of generality let node 1 be a node with exactly three outgoing edges e_1, e_2, e_3, and let $L(e_1) = L(e_2) = A_1$. Let $\mathcal{D}(e)$ denote the destination node of an edge $e \in E$. If $\mathcal{D}(e_1) = \mathcal{D}(e_2)$, then e_1 (or e_2) can be removed without changing the output set of words. If $\mathcal{D}(e_1) \neq \mathcal{D}(e_2)$, assume, without loss of generality, that $\mathcal{D}(e_1) = 1$ and $\mathcal{D}(e_2) = 2$. Now, if $L(e_3) = A_1$, then regardless of its destination node, e_3 can be removed. If $L(e_3) = A_2$ and $\mathcal{D}(e_3) = 1$, then V_1 is a common Lyapunov function for $\mathcal{A}$. The only remaining possibility is that $L(e_3) = A_2$ and $\mathcal{D}(e_3) = 2$. Note that there must be an edge $e_4 \in E$ from node 2 to node 1, otherwise either node 2 would have two self-edges with the same label or V_2 would be a common Lyapunov function for $\mathcal{A}$. If $L(e_4) = A_2$ then it can be verified that $G(\{1,2\}, \{e_1, e_2, e_3, e_4\})$ is path-complete and thus all other edge can be removed. If there is no edge from node 2 to node 1 with label A_2 then $L(e_4) = A_1$ and node 2 must have a self-edge $e_5 \in E$ with label $L(e_5) = A_2$, otherwise the graph would not be path-complete. In this case, it can be verified that e_2 can be removed without affecting the output set of words. $\square$

It can be verified that a path-complete graph with two nodes and less than four edges must necessarily place two self-loops with different labels on one node, which necessitates existence of a common Lyapunov function for the underlying switched system. Since we are interested in exploiting the favorable properties of graph Lyapunov functions in approximation of the JSR, we will focus on graphs with four edges.

Before we proceed, for convenience we introduce the following notation: Given a labeled graph $G(N, E)$ associated with two matrices A_1 and A_2, we denote by $\overline{G}(N, E)$, the graph obtained by swapping of A_1 and A_2 in all the labels on every edge.

5.4.2 Comparison of performance

It can be verified that for path-complete graphs with two nodes, four edges, and two matrices, and without multiple self-loops on a single node, there are a total of nine distinct graph topologies to consider. Of the nine graphs, six have the property that every node has two incoming edges with different labels. These are graphs G_1, G_2, $\overline{G}_2$, G_3, $\overline{G}_3$, and G_4 (Figure 5.3). Note that $\overline{G}_1 = G_1$ and $\overline{G}_4 = G_4$. The duals of these six graphs, i.e., G_1', G_2', $\overline{G}_2'$, $G_3' = G_3$, $\overline{G}_3' = \overline{G}_3$, and $G_4' = G_4$ have the property that every node has two outgoing edges with different labels. Evidently, G_3, $\overline{G}_3$, and G_4 are *self-dual graphs*, i.e., they are isomorphic to their dual graphs. The self-dual graphs are least interesting to us since, as we will show, they necessitate existence of a common Lyapunov function for $\mathcal{A}$ (cf. Proposition 5.12, equation (5.18)).

Note that all of these graphs perform at least as well as a common Lyapunov function because we can always take $V_1(x) = V_2(x)$. Furthermore, we know from Corollaries 5.9 and 5.8 that if Lyapunov inequalities associated with G_1, G_2, $\overline{G}_2$, G_3, $\overline{G}_3$, and G_4 are satisfied, then $\max\{V_1(x), V_2(x)\}$ is a common Lyapunov function, whereas, in the case of graphs G_1', G_2', $\overline{G}_2'$, G_3', $\overline{G}_3'$, and G_4', the function $\min\{V_1(x), V_2(x)\}$ would serve as a common Lyapunov function. Clearly, for the self-dual graphs G_3, $\overline{G}_3$, and G_4 both $\max\{V_1(x), V_2(x)\}$ and $\min\{V_1(x), V_2(x)\}$ are common Lyapunov functions.

Notation: Given a set of matrices $\mathcal{A} = \{A_1, \cdots, A_m\}$, a path-complete graph $G(N, E)$, and a class of functions $\mathcal{V}$, we denote by $\hat{\rho}_{\mathcal{V},G}(\mathcal{A})$, the upper bound on the JSR of $\mathcal{A}$ that can be obtained by numerical optimization of GLFs

$V_i \in \mathcal{V},\ i \in N$, defined over G. With a slight abuse of notation, we denote by $\hat{\rho}_{\mathcal{V}}(\mathcal{A})$, the upper bound that is obtained by using a common Lyapunov function $V \in \mathcal{V}$.

Proposition 5.12. *Consider the set $\mathcal{A} = \{A_1, A_2\}$, and let $G_1,\ G_2,\ G_3,\ G_4$, and H_3 be the path-complete graphs shown in Figure 5.3. Then, the upper bounds on the JSR of $\mathcal{A}$ obtained by analysis via the associated GLFs satisfy the following relations:*

$$\hat{\rho}_{\mathcal{V},G_1}(\mathcal{A}) = \hat{\rho}_{\mathcal{V},G_1'}(\mathcal{A}) \tag{5.17}$$

and

$$\hat{\rho}_{\mathcal{V}}(\mathcal{A}) = \hat{\rho}_{\mathcal{V},G_3}(\mathcal{A}) = \hat{\rho}_{\mathcal{V},\overline{G}_3}(\mathcal{A}) = \hat{\rho}_{\mathcal{V},G_4}(\mathcal{A}) \tag{5.18}$$

and

$$\hat{\rho}_{\mathcal{V},G_2}(\mathcal{A}) = \hat{\rho}_{\mathcal{V},H_3}(\mathcal{A}), \qquad \hat{\rho}_{\mathcal{V},\overline{G}_2}(\mathcal{A}) = \hat{\rho}_{\mathcal{V},\overline{H}_3}(\mathcal{A}) \tag{5.19}$$

and

$$\hat{\rho}_{\mathcal{V},G_2'}(\mathcal{A}) = \hat{\rho}_{\mathcal{V},H_3'}(\mathcal{A}), \qquad \hat{\rho}_{\mathcal{V},\overline{G}_2'}(\mathcal{A}) = \hat{\rho}_{\mathcal{V},\overline{H}_3'}(\mathcal{A}). \tag{5.20}$$

Proof. A proof of (5.17) in more generality is provided in Section 5.5 (cf. Corollary 5.15). The proof of (5.18) is based on symmetry arguments. Let $\{V_1, V_2\}$ be a GLF associated with G_3 (V_1 is associated with node 1 and V_2 is associated with node 2). Then, by symmetry, $\{V_2, V_1\}$ is also a GLF for G_3 (where V_1 is associated with node 2 and V_2 is associated with node 1). Therefore, letting $V = V_1 + V_2$, we have that $\{V, V\}$ is a GLF for G_3 and thus, $V = V_1 + V_2$ is also a common Lyapunov function for $\mathcal{A}$, which implies that $\hat{\rho}_{\mathcal{V},G_3}(\mathcal{A}) \geq \hat{\rho}_{\mathcal{V}}(\mathcal{A})$. The other direction is trivial: If $V \in \mathcal{V}$ is a common Lyapunov function for $\mathcal{A}$, then $\{V_1, V_2 \mid V_1 = V_2 = V\}$ is a GLF associated with G_3, and hence, $\hat{\rho}_{\mathcal{V},G_3}(\mathcal{A}) \leq \hat{\rho}_{\mathcal{V}}(\mathcal{A})$. Identical arguments based on symmetry hold for $\overline{G}_3$ and G_4. We now prove the left equality in (5.19), the proofs for the remaining equalities in (5.19) and (5.20) are analogous. The equivalence between G_2 and H_3 is a special case of the relation between a graph and its *reduced* model, obtained by removing a node without any self-loops, adding a new edge per each pair of incoming and outgoing edges to that node, and then labeling the new edges by taking the composition of the labels of the corresponding incoming and outgoing edges in the original graph; see [145], [144, Chap. 5]. Note that H_3 is an offspring of G_2 in this sense. This intuition helps construct a proof. Let $\{V_1, V_2\}$ be a GLF associated with G_2. It can be verified that V_1 is a Lyapunov function associated with H_3, and therefore, $\hat{\rho}_{\mathcal{V},H_3}(\mathcal{A}) \leq \hat{\rho}_{\mathcal{V},G_2}(\mathcal{A})$. Similarly, if $V \in \mathcal{V}$ is a Lyapunov function associated with H_3, then one can check that $\{V_1, V_2 \mid V_1(x) = V(x), V_2(x) = V(A_2 x)\}$ is a GLF associated with G_2, and hence, $\hat{\rho}_{\mathcal{V},H_3}(\mathcal{A}) \geq \hat{\rho}_{\mathcal{V},G_2}(\mathcal{A})$. □

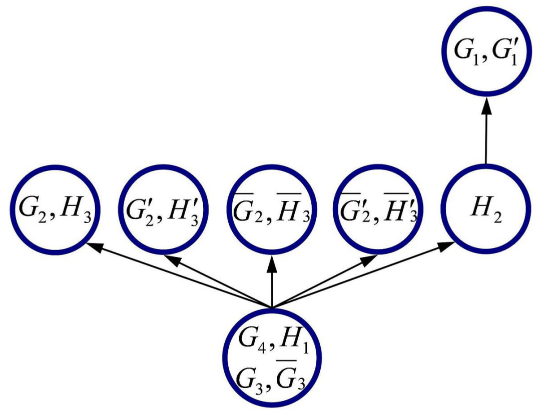


Figure 5.6. A diagram describing the relative performance of the path-complete graphs of Figure 5.3 together with their duals and label permutations. The graphs placed in the same circle always give the same approximation of the JSR. A graph at the end of an arrow results in an approximation of the JSR that is always at least as good as that of the graph at the start of the arrow. When there is no directed path between two graphs in this diagram, either graph can outperform the other depending on the set of matrices $\mathcal{A}$.

Remark 5.4.1. Proposition 5.12 (equation 5.17) establishes the equivalence of the bounds obtained from the pair of dual graphs G_1 and G'_1. This, however, is not true for graphs G_2 and $\overline{G}_2$ as there exist examples for which

$$\hat{\rho}_{\mathcal{V},G_2}(\mathcal{A}) \neq \hat{\rho}_{\mathcal{V},G'_2}(\mathcal{A}),$$
$$\hat{\rho}_{\mathcal{V},\overline{G}_2}(\mathcal{A}) \neq \hat{\rho}_{\mathcal{V},\overline{G}'_2}(\mathcal{A}).$$

The diagram in Figure 5.6 summarizes the results of this section. We remark that no relations other than the ones given in Figure 5.6 can be made among these path-complete graphs. Indeed, whenever there are no relations between two graphs in Figure 5.6, we have examples of matrices A_1, A_2 (not presented here) for which one graph can outperform the other.

The graphs G_1 and G'_1 seem to statistically perform better than all other graphs in Figure 5.6. For example, we ran experiments on a set of 100 random 5×5 matrices $\{A_1, A_2\}$ with elements uniformly distributed in $[-1, 1]$ to compare the performance of graphs G_1, G_2 and $\overline{G}_2$. If in each case we also consider the relabeled matrices (i.e., $\{A_2, A_1\}$) as our input, then, out of the total 200 instances, graph G_1 produced strictly better bounds on the JSR 58 times, whereas graphs G_2 and $\overline{G}_2$ each produced the best bound of the three graphs only 23 times. (The numbers do not add up to 200 due to ties.) In addition to this superior performance, the bound $\hat{\rho}_{\mathcal{V},G_1}(\{A_1, A_2\})$ obtained by analysis via the graph G_1 is invariant under (i) permutation of the labels A_1 and A_2 (obvious), and (ii) transposing of A_1 and A_2 (Corollary 5.15). These are desirable properties which fail to hold for G_2 and

$\overline{G}_2$ or their duals. Motivated by these observations, we generalize G_1 and its dual G_1' in the next section to the case of m matrices and m Lyapunov functions and establish that they have certain appealing properties. We will prove (cf. Theorem 5.16) that these graphs always perform better than a common Lyapunov function in 2 steps (i.e., the graph H_2 in Figure 5.3), whereas, this is not the case for G_2 and $\overline{G}_2$ or their duals.

5.5 Further analysis of a particular family of path-complete graphs

The framework of path-complete graphs provides a multitude of semidefinite programming based techniques for the approximation of the JSR whose performance vary with computational cost. For instance, as we increase the number of nodes of the graph, or the degree of the polynomial Lyapunov functions assigned to the nodes, or the number of edges of the graph that instead of labels of length one have labels of higher length, we obtain better results but at a higher computational cost. Many of these approximation techniques are asymptotically tight, so in theory they can be used to achieve any desired accuracy of approximation. For example,

$$\hat{\rho}_{\mathcal{V}^{SOS,2d}}(\mathcal{A}) \rightarrow \rho(\mathcal{A}) \text{ as } 2d \rightarrow \infty,$$

where $\mathcal{V}^{SOS,2d}$ denotes the class of sum of squares homogeneous polynomial Lyapunov functions of degree $2d$. (Recall our notation for bounds from Section 5.4.2.) It is also true that a common quadratic Lyapunov function for products of higher length achieves the true JSR asymptotically [85]; i.e.[5],

$$\sqrt[t]{\hat{\rho}_{\mathcal{V}^2}(\mathcal{A}^t)} \rightarrow \rho(\mathcal{A}) \text{ as } t \rightarrow \infty.$$

Nevertheless, it is desirable for practical purposes to identify a class of path-complete graphs that provide a good tradeoff between quality of approximation and computational cost. Towards this objective, we propose the use of m quadratic Lyapunov functions assigned to the nodes of the De Bruijn graph of order 1 on m symbols for the approximation of the JSR of a set of m matrices. This graph and its dual are particular path-complete graphs with m nodes and m^2 edges and will be the subject of study in this section. If we denote the quadratic Lyapunov functions by $x^T P_i x$, then we are proposing the use of linear matrix inequalities

$$\begin{array}{rcll} P_i & \succ & 0 & \forall i = 1, \ldots, m, \\ \gamma^2 A_i^T P_j A_i & \preceq & P_i & \forall i, j = \{1, \ldots, m\}^2 \end{array} \tag{5.21}$$

[5]By $\mathcal{V}^2$ we denote the class of quadratic homogeneous polynomials. We drop the superscript “SOS” because nonnegative quadratic polynomials are always sums of squares.

or the set of LMIs

$$\begin{array}{rcl} P_i & \succ & 0 \quad \forall i = 1, \ldots, m, \\ \gamma^2 A_i^T P_i A_i & \preceq & P_j \quad \forall i, j = \{1, \ldots, m\}^2 \end{array} \tag{5.22}$$

for the approximation of the JSR of m matrices. Throughout this section, we denote the path-complete graphs associated with (5.21) and (5.22) with G_1 and G_1' respectively. (The De Bruijn graph of order 1, by standard convention, is actually the graph G_1'.) Observe that G_1 and G_1' are indeed dual graphs as they can be obtained from each other by reversing the direction of the edges. For the case $m = 2$, our notation is consistent with the previous section and these graphs are illustrated in Figure 5.4. Also observe from Corollary 5.8 and Corollary 5.9 that the LMIs in (5.21) give rise to max-of-quadratics Lyapunov functions, whereas the LMIs in (5.22) lead to min-of-quadratics Lyapunov functions. We will prove in this section that the approximation bound obtained by these LMIs (i.e., the reciprocal of the largest γ for which the LMIs (5.21) or (5.22) hold) is always the same and lies within a multiplicative factor of $\frac{1}{\sqrt[4]{n}}$ of the true JSR, where n is the dimension of the matrices. The relation between the bound obtained by a pair of dual path-complete graphs has a connection to transposition of the matrices in the set $\mathcal{A}$. We explain this next.

5.5.1 Duality and invariance under transposition

In [63], [64], it is shown that absolute asymptotic stability of the linear difference inclusion in (5.3) defined by the matrices $\mathcal{A} = \{A_1, \ldots, A_m\}$ is equivalent to absolute asymptotic stability of (5.3) for the transposed matrices $\mathcal{A}^T := \{A_1^T, \ldots, A_m^T\}$. Note that this fact is immediately seen from the definition of the JSR in (5.1), since $\rho(\mathcal{A}) = \rho(\mathcal{A}^T)$. It is also well-known that

$$\hat{\rho}_{\mathcal{V}^2}(\mathcal{A}) = \hat{\rho}_{\mathcal{V}^2}(\mathcal{A}^T).$$

Indeed, if $x^T P x$ is a common quadratic Lyapunov function for the set $\mathcal{A}$, then it is easy to show that $x^T P^{-1} x$ is a common quadratic Lyapunov function for the set $\mathcal{A}^T$. However, this nice property is not true for the bound obtained from some other techniques. For instance, the next example shows that

$$\hat{\rho}_{\mathcal{V}^{SOS,4}}(\mathcal{A}) \neq \hat{\rho}_{\mathcal{V}^{SOS,4}}(\mathcal{A}^T), \tag{5.23}$$

i.e, the upper bound obtained by searching for a common quartic sos polynomial is not invariant under transposition.

Example 5.5.1. Consider the set of matrices $\mathcal{A} = \{A_1, A_2, A_3, A_4\}$, with

$$A_1 = \begin{bmatrix} 10 & -6 & -1 \\ 8 & 1 & -16 \\ -8 & 0 & 17 \end{bmatrix}, A_2 = \begin{bmatrix} -5 & 9 & -14 \\ 1 & 5 & 10 \\ 3 & 2 & 16 \end{bmatrix}, A_3 = \begin{bmatrix} -14 & 1 & 0 \\ -15 & -8 & -12 \\ -1 & -6 & 7 \end{bmatrix}, A_4 = \begin{bmatrix} 1 & -8 & -2 \\ 1 & 16 & 3 \\ 16 & 11 & 14 \end{bmatrix}.$$

We have $\hat{\rho}_{\mathcal{V}^{SOS,4}}(\mathcal{A}) = 21.411$, but $\hat{\rho}_{\mathcal{V}^{SOS,4}}(\mathcal{A}^T) = 21.214$ (up to three significant digits). $\triangle$

Similarly, the bound obtained by non-convex inequalities proposed in [63] is not invariant under transposing the matrices. For such methods, one would have to run the numerical optimization twice—once for the set $\mathcal{A}$ and once for the set $\mathcal{A}^T$—and then pick the better bound of the two. We will show that by contrast, the bound obtained from the LMIs in (5.21) and (5.22) are invariant under transposing the matrices. Before we do that, let us prove a general result which states that for path-complete graphs with quadratic Lyapunov functions as nodes, transposing the matrices has the same effect as dualizing the graph.

Theorem 5.13. *Let $G(N,E)$ be a path-complete graph, and let $G'(N,E')$ be its dual graph. Then,*

$$\hat{\rho}_{\mathcal{V}^2,G}(\mathcal{A}^T) = \hat{\rho}_{\mathcal{V}^2,G'}(\mathcal{A}). \tag{5.24}$$

Proof. For ease of notation, we prove the claim for the case where the edge labels of $G(N,E)$ have length one. The proof of the general case is identical. Pick an arbitrary edge $(i,j) \in E$ going from node i to node j and labeled with some matrix $A_l \in \mathcal{A}$. By the application of the Schur complement we have

$$A_l P_j A_l^T \preceq P_i \;\Leftrightarrow\; \begin{bmatrix} P_i & A_l \\ A_l^T & P_j^{-1} \end{bmatrix} \succeq 0 \;\Leftrightarrow\; A_l^T P_i^{-1} A_l \preceq P_j^{-1}.$$

But this already establishes the claim since we see that P_i and P_j satisfy the LMI associated with edge $(i,j) \in E$ when the matrix A_l is transposed if and only if P_j^{-1} and P_i^{-1} satisfy the LMI associated with edge $(j,i) \in E'$. $\square$

Corollary 5.14. $\hat{\rho}_{\mathcal{V}^2,G}(\mathcal{A}) = \hat{\rho}_{\mathcal{V}^2,G}(\mathcal{A}^T)$ *if and only if* $\hat{\rho}_{\mathcal{V}^2,G}(\mathcal{A}) = \hat{\rho}_{\mathcal{V}^2,G'}(\mathcal{A})$.

Proof. This is an immediate consequence of the equality in (5.24). $\square$

It is an interesting question for future research to characterize the topologies of path-complete graphs for which one has $\hat{\rho}_{\mathcal{V}^2,G}(\mathcal{A}) = \hat{\rho}_{\mathcal{V}^2,G}(\mathcal{A}^T)$. For example, the above corollary shows that this is obviously the case for any path-complete graph that is self-dual. Let us show next that this is also the case for graphs G_1 and G_1' despite the fact that they are not self-dual.

Corollary 5.15. *For the path-complete graphs G_1 and G_1' associated with the inequalities in (5.21) and (5.22), and for any class of continuous, homogeneous, and positive definite functions $\mathcal{V}$, we have*

$$\hat{\rho}_{\mathcal{V},G_1}(\mathcal{A}) = \hat{\rho}_{\mathcal{V},G_1'}(\mathcal{A}). \tag{5.25}$$

Moreover, if quadratic Lyapunov functions are assigned to the nodes of G_1 and G_1', then we have

$$\hat{\rho}_{\mathcal{V}^2,G_1}(\mathcal{A}) = \hat{\rho}_{\mathcal{V}^2,G_1}(\mathcal{A}^T) = \hat{\rho}_{\mathcal{V}^2,G_1'}(\mathcal{A}) = \hat{\rho}_{\mathcal{V}^2,G_1'}(\mathcal{A}^T). \tag{5.26}$$

Proof. The proof of (5.25) is established by observing that the GLFs associated with G_1 and G_1' can be derived from one another via $V_i'(A_i x) = V_i(x)$. (Note that we are relying here on the assumption that the matrices A_i are invertible, which as we noted in Remark 5.2.1, is not a limiting assumption.) Since (5.25) in particular implies that $\hat{\rho}_{\mathcal{V}^2,G_1}(\mathcal{A}) = \hat{\rho}_{\mathcal{V}^2,G_1'}(\mathcal{A})$, we get the rest of the equalities in (5.26) immediately from Corollary 5.14 and this finishes the proof. For concreteness, let us also prove the leftmost equality in (5.26) directly. Let P_i, $i = 1, \dots, m$, satisfy the LMIs in (5.21) for the set of matrices $\mathcal{A}$. Then, the reader can check that

$$\tilde{P}_i = A_i P_i^{-1} A_i^T, \quad i = 1, \dots, m,$$

satisfy the LMIs in (5.21) for the set of matrices $\mathcal{A}^T$. $\square$

5.5.2 An approximation guarantee

The next theorem gives a bound on the quality of approximation of the estimate resulting from the LMIs in (5.21) and (5.22). Since we have already shown that $\hat{\rho}_{\mathcal{V}^2,G_1}(\mathcal{A}) = \hat{\rho}_{\mathcal{V}^2,G_1'}(\mathcal{A})$, it is enough to prove this bound for the LMIs in (5.21).

Theorem 5.16. *Let $\mathcal{A}$ be a set of m matrices in $\mathbb{R}^{n\times n}$ with JSR $\rho(\mathcal{A})$. Let $\hat{\rho}_{\mathcal{V}^2,G_1}(\mathcal{A})$ be the bound on the JSR obtained from the LMIs in (5.21). Then,*

$$\frac{1}{\sqrt[4]{n}}\hat{\rho}_{\mathcal{V}^2,G_1}(\mathcal{A}) \leq \rho(\mathcal{A}) \leq \hat{\rho}_{\mathcal{V}^2,G_1}(\mathcal{A}). \tag{5.27}$$

Proof. The right inequality is just a consequence of G_1 being a path-complete graph (Theorem 5.4). To prove the left inequality, consider the set $\mathcal{A}^2$ consisting of all m^2 products of length two. In view of (5.6), a common quadratic Lyapunov function for this set satisfies the bound

$$\frac{1}{\sqrt{n}}\hat{\rho}_{\mathcal{V}^2}(\mathcal{A}^2) \leq \rho(\mathcal{A}^2).$$

It is easy to show that

$$\rho(\mathcal{A}^2) = \rho^2(\mathcal{A}).$$

See e.g. [85]. Therefore,

$$\frac{1}{\sqrt[4]{n}}\hat{\rho}_{\mathcal{V}^2}^{\frac{1}{2}}(\mathcal{A}^2) \leq \rho(\mathcal{A}). \tag{5.28}$$

Now suppose for some $\gamma > 0$, $x^T Q x$ is a common quadratic Lyapunov function for the matrices in $\mathcal{A}_\gamma^2$; i.e., it satisfies

$$\begin{array}{rcl} Q & \succ & 0 \\ \gamma^4 (A_i A_j)^T Q A_i A_j & \preceq & Q \quad \forall i,j = \{1,\ldots,m\}^2. \end{array}$$

Then, we leave it to the reader to check that

$$P_i = Q + A_i^T Q A_i, \quad i = 1,\ldots,m$$

satisfy (5.21). Hence,

$$\hat{\rho}_{\mathcal{V}^2,G_1}(\mathcal{A}) \leq \hat{\rho}_{\mathcal{V}^2}^{\frac{1}{2}}(\mathcal{A}^2),$$

and in view of (5.28) the claim is established. □

Note that the bound in (5.27) is independent of the number of matrices. Moreover, we remark that this bound is tighter, in terms of its dependence on n, than the known bounds for $\hat{\rho}_{\mathcal{V}^{SOS,2d}}$ for any finite degree $2d$ of the sum of squares polynomials. The reader can check that the bound in (5.7) goes asymptotically as $\frac{1}{\sqrt{n}}$. Numerical evidence suggests that the performance of both the bound obtained by sum of squares polynomials and the bound obtained by the LMIs in (5.21) and (5.22) is much better than the provable bounds in (5.7) and in Theorem 5.16. The problem of improving these bounds or establishing their tightness is open. It goes without saying that instead of quadratic functions, we can associate sum of squares polynomials to the nodes of G_1 and obtain a more powerful technique for which we can also prove better bounds with the exact same arguments.

■ 5.5.3 Numerical examples

In the proof of Theorem 5.16, we essentially showed that the bound obtained from LMIs in (5.21) is tighter than the bound obtained from a common quadratic applied to products of length two. Our first example shows that the LMIs in (5.21) can in fact do better than a common quadratic applied to products of *any* finite length.

Example 5.5.2. Consider the set of matrices $\mathcal{A} = \{A_1, A_2\}$, with

$$A_1 = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}, \; A_2 = \begin{bmatrix} 0 & 1 \\ 0 & -1 \end{bmatrix}.$$

This is a benchmark set of matrices that has been studied in [13], [122], [6] because it gives the worst case approximation ratio of a common quadratic Lyapunov function. Indeed, it is easy to show that $\rho(\mathcal{A}) = 1$, but $\hat{\rho}_{\mathcal{V}^2}(\mathcal{A}) = \sqrt{2}$. Moreover, the bound obtained by a common quadratic function applied to the set $\mathcal{A}^t$ is

$$\hat{\rho}_{\mathcal{V}^2}^{\frac{1}{t}}(\mathcal{A}^t) = 2^{\frac{1}{2t}},$$

which for no finite value of t is exact. On the other hand, we show that the LMIs in (5.21) give the exact bound; i.e., $\hat{\rho}_{\mathcal{V}^2,G_1}(\mathcal{A}) = 1$. Due to the simple structure of A_1 and A_2, we can even give an analytical expression for our Lyapunov functions. Given any $\varepsilon > 0$, the LMIs in (5.21) with $\gamma = 1/(1+\varepsilon)$ are feasible with

$$P_1 = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix}, \qquad P_2 = \begin{bmatrix} b & 0 \\ 0 & a \end{bmatrix},$$

for any $b > 0$ and $a > b/2\varepsilon$. △

Example 5.5.3. Consider the set of randomly generated matrices $\mathcal{A} = \{A_1, A_2, A_3\}$, with

$$A_1 = \begin{bmatrix} 0 & -2 & 2 & 2 & 4 \\ 0 & 0 & -4 & -1 & -6 \\ 2 & 6 & 0 & -8 & 0 \\ -2 & -2 & -3 & 1 & -3 \\ -1 & -5 & 2 & 6 & -4 \end{bmatrix}, A_2 = \begin{bmatrix} -5 & -2 & -4 & 6 & -1 \\ 1 & 1 & 4 & 3 & -5 \\ -2 & 3 & -2 & 8 & -1 \\ 0 & 8 & -6 & 2 & 5 \\ -1 & -5 & 1 & 7 & -4 \end{bmatrix}, A_3 = \begin{bmatrix} 3 & -8 & -3 & 2 & -4 \\ -2 & -2 & -9 & 4 & -1 \\ 2 & 2 & -5 & -8 & 6 \\ -4 & -1 & 4 & -3 & 0 \\ 0 & 5 & 0 & -3 & 5 \end{bmatrix}.$$

A lower bound on $\rho(\mathcal{A})$ is $\rho(A_1A_2A_2)^{1/3} = 11.8015$. The upper approximations for $\rho(\mathcal{A})$ that we computed for this example are as follows:

$$\begin{array}{rcl} \hat{\rho}_{\mathcal{V}^2}(\mathcal{A}) & = & 12.5683 \\ \hat{\rho}_{\mathcal{V}^2}^{\frac{1}{2}}(\mathcal{A}^2) & = & 11.9575 \\ \hat{\rho}_{\mathcal{V}^2,G_1}(\mathcal{A}) & = & 11.8097 \\ \hat{\rho}_{\mathcal{V}^{SOS,4}}(\mathcal{A}) & = & 11.8015. \end{array} \tag{5.29}$$

The bound $\hat{\rho}_{\mathcal{V}^{SOS,4}}$ matches the lower bound numerically and is most likely exact for this example. This bound is slightly better than $\hat{\rho}_{\mathcal{V}^2,G_1}$. However, a simple calculation shows that the semidefinite program resulting in $\hat{\rho}_{\mathcal{V}^{SOS,4}}$ has 25 more decision variables than the one for $\hat{\rho}_{\mathcal{V}^2,G_1}$. Also, the running time of the algorithm leading to $\hat{\rho}_{\mathcal{V}^{SOS,4}}$ is noticeably larger than the one leading to $\hat{\rho}_{\mathcal{V}^2,G_1}$. In general, when the dimension of the matrices is large, it can often be cost-effective to increase the number of the nodes of our path-complete graphs but keep the degree of the polynomial Lyapunov functions assigned to its nodes relatively low. △

■ 5.6 Converse Lyapunov theorems and approximation with arbitrary accuracy

It is well-known that existence of a Lyapunov function which is the pointwise maximum of quadratics is not only sufficient but also necessary for absolute asymptotic stability of (5.2) or (5.3); see e.g. [105]. This is perhaps an intuitive fact if we recall that switched systems of type (5.2) and (5.3) always admit a convex Lyapunov function. Indeed, if we take "enough" quadratics, the convex and compact unit sublevel set of a convex Lyapunov function can be approximated arbitrarily well with sublevel sets of max-of-quadratics Lyapunov functions, which are intersections of ellipsoids. This of course implies that the bound obtained from max-of-quadratics Lyapunov functions is asymptotically tight for the approximation of the JSR. However, this converse Lyapunov theorem does not answer two natural questions of importance in practice: (i) How many quadratic functions do we need to achieve a desired quality of approximation? (ii) Can we search for these quadratic functions via semidefinite programming or do we need to resort to non-convex formulations? Our next theorem provides an answer to these questions.

Theorem 5.17. *Let $\mathcal{A}$ be a set of m matrices in $\mathbb{R}^{n\times n}$. Given any positive integer l, there exists an explicit path-complete graph G consisting of m^{l-1} nodes assigned to quadratic Lyapunov functions and m^l edges with labels of length one such that the linear matrix inequalities associated with G imply existence of a max-of-quadratics Lyapunov function and the resulting bound obtained from the LMIs satisfies*

$$\frac{1}{\sqrt[2l]{n}}\hat{\rho}_{\mathcal{V}^2,G}(\mathcal{A}) \leq \rho(\mathcal{A}) \leq \hat{\rho}_{\mathcal{V}^2,G}(\mathcal{A}). \tag{5.30}$$

Proof. Let us denote the m^{l-1} quadratic Lyapunov functions by $x^T P_{i_1\ldots i_{l-1}} x$, where $i_1 \ldots i_{l-1} \in \{1,\ldots,m\}^{l-1}$ is a multi-index used for ease of reference to our Lyapunov functions. We claim that we can let G be the graph dual to the De Bruijn graph of order $l-1$ on m symbols. The LMIs associated to this graph are given by

$$\begin{array}{rcl} P_{i_1 i_2\ldots i_{l-2} i_{l-1}} & \succ & 0 \quad \forall i_1 \ldots i_{l-1} \in \{1,\ldots,m\}^{l-1} \\ A_j^T P_{i_1 i_2 \ldots i_{l-2} i_{l-1}} A_j & \preceq & P_{i_2 i_3 \ldots i_{l-1} j} \\ & & \forall i_1 \ldots i_{l-1} \in \{1,\ldots,m\}^{l-1}, \\ & & \forall j \in \{1,\ldots,m\}. \end{array} \tag{5.31}$$

The fact that G is path-complete and that the LMIs imply existence of a max-of-quadratics Lyapunov function follows from Corollary 5.9. The proof that these LMIs satisfy the bound in (5.30) is a straightforward generalization of the proof of Theorem 5.16. By the same arguments we have

$$\frac{1}{\sqrt[2l]{n}}\hat{\rho}_{\mathcal{V}^2}^{\frac{1}{l}}(\mathcal{A}^l) \leq \rho(\mathcal{A}). \tag{5.32}$$

Suppose x^TQx is a common quadratic Lyapunov function for the matrices in $\mathcal{A}^l$; i.e., it satisfies

$$\begin{array}{rcl} Q & \succ & 0 \\ (A_{i_1}\dots A_{i_l})^T Q A_{i_1}\dots A_{i_l} & \preceq & Q \quad \forall i_1\dots i_l \in \{1,\dots,m\}^l. \end{array}$$

Then, it is easy to check that[6]

$$\begin{array}{l} P_{i_1 i_2 \dots i_{l-2} i_{l-1}} = Q + A_{i_{l-1}}^T Q A_{i_{l-1}} \\ +(A_{i_{l-2}}A_{i_{l-1}})^T Q (A_{i_{l-2}}A_{i_{l-1}}) + \cdots \\ +(A_{i_1}A_{i_2}\dots A_{i_{l-2}}A_{i_{l-1}})^T Q (A_{i_1}A_{i_2}\dots A_{i_{l-2}}A_{i_{l-1}}), \\ i_1\dots i_{l-1} \in \{1,\dots,m\}^{l-1}, \end{array}$$

satisfy (5.31). Hence,

$$\hat{\rho}_{\mathcal{V}^2,G}(\mathcal{A}) \leq \hat{\rho}_{\mathcal{V}^2}^{\frac{1}{l}}(\mathcal{A}^l),$$

and in view of (5.32) the claim is established. □

Remark 5.6.1. A converse Lyapunov theorem identical to Theorem 5.17 can be proven for the min-of-quadratics Lyapunov functions. The only difference is that the LMIs in (5.31) would get replaced by the ones corresponding to the dual graph of G.

Our last theorem establishes approximation bounds for a family of path-complete graphs with one single node but several edges labeled with words of different lengths. Examples of such path-complete graphs include graph H_3 in Figure 5.3 and graph H_4 in Figure 5.5.

Theorem 5.18. *Let $\mathcal{A}$ be a set of matrices in $\mathbb{R}^{n\times n}$. Let $\tilde{G}(\{1\},E)$ be a path-complete graph, and l be the length of the shortest word in $\tilde{\mathcal{A}} = \{L(e) : e \in E\}$. Then $\hat{\rho}_{\mathcal{V}^2,\tilde{G}}(\mathcal{A})$ provides an estimate of $\rho(\mathcal{A})$ that satisfies*

$$\frac{1}{\sqrt[2l]{n}}\hat{\rho}_{\mathcal{V}^2,\tilde{G}}(\mathcal{A}) \leq \rho(\mathcal{A}) \leq \hat{\rho}_{\mathcal{V}^2,\tilde{G}}(\mathcal{A}).$$

Proof. The right inequality is obvious, we prove the left one. Since both $\hat{\rho}_{\mathcal{V}^2,\tilde{G}}(\mathcal{A})$ and ρ are homogeneous in $\mathcal{A}$, we may assume, without loss of generality, that $\hat{\rho}_{\mathcal{V}^2,\tilde{G}}(\mathcal{A}) = 1$. Suppose for the sake of contradiction that

$$\rho(\mathcal{A}) < 1/\sqrt[2l]{n}. \tag{5.33}$$

[6]The construction of the Lyapunov function here is a special case of a general scheme for constructing Lyapunov functions that are monotonically decreasing from those that decrease only every few steps; see [1, p. 58].

We will show that this implies that $\hat{\rho}_{\mathcal{V}^2,\tilde{G}}(\mathcal{A}) < 1$. Towards this goal, let us first prove that $\rho(\tilde{\mathcal{A}}) \leq \rho^l(\mathcal{A})$. Indeed, if we had $\rho(\tilde{\mathcal{A}}) > \rho^l(\mathcal{A})$, then there would exist[7] an integer i and a product $A_\sigma \in \tilde{\mathcal{A}}^i$ such that

$$\rho^{\frac{1}{i}}(A_\sigma) > \rho^l(\mathcal{A}). \tag{5.34}$$

Since we also have $A_\sigma \in \mathcal{A}^j$ (for some $j \geq il$), it follows that

$$\rho^{\frac{1}{j}}(A_\sigma) \leq \rho(\mathcal{A}). \tag{5.35}$$

The inequality in (5.34) together with $\rho(\mathcal{A}) \leq 1$ gives

$$\rho^{\frac{1}{j}}(A_\sigma) > \rho^{\frac{il}{j}}(\mathcal{A}) \geq \rho(\mathcal{A}).$$

But this contradicts (5.35). Hence we have shown

$$\rho(\tilde{\mathcal{A}}) \leq \rho^l(\mathcal{A}).$$

Now, by our hypothesis (5.33) above, we have that $\rho(\tilde{\mathcal{A}}) < 1/\sqrt{n}$. Therefore, there exists $\epsilon > 0$ such that $\rho((1+\epsilon)\tilde{\mathcal{A}}) < 1/\sqrt{n}$. It then follows from (5.6) that there exists a common quadratic Lyapunov function for $(1+\epsilon)\tilde{\mathcal{A}}$. Hence, $\hat{\rho}_{\mathcal{V}^2}((1+\epsilon)\tilde{\mathcal{A}}) \leq 1$, which immediately implies that $\hat{\rho}_{\mathcal{V}^2,\tilde{G}}(\mathcal{A}) < 1$, a contradiction. □

A noteworthy immediate corollary of Theorem 5.18 (obtained by setting $\tilde{\mathcal{A}} = \bigcup_{t=r}^{k} \mathcal{A}^t$) is the following: If $\rho(\mathcal{A}) < \frac{1}{\sqrt[2r]{n}}$, then there exists a quadratic Lyapunov function that decreases simultaneously for all products of lengths $r, r+1, \ldots, r+k$, for any desired value of k. Note that this fact is obvious for $r = 1$, but nonobvious for $r \geq 2$.

5.7 Conclusions and future directions

We introduced the framework of path-complete graph Lyapunov functions for the formulation of semidefinite programming based algorithms for approximating the joint spectral radius (or equivalently establishing absolute asymptotic stability of an arbitrary switched linear system). We defined the notion of a path-complete graph, which was inspired by concepts in automata theory. We showed that every path-complete graph gives rise to a technique for the approximation of the JSR. This provided a unifying framework that includes many of the previously proposed techniques and also introduces new ones. (In fact, all families of LMIs

[7] Here, we are appealing to the well-known fact about the JSR of a general set of matrices $\mathcal{B}$: $\rho(\mathcal{B}) = \limsup_{k\to\infty} \max_{B\in\mathcal{B}^k} \rho^{\frac{1}{k}}(B)$. See e.g. [85, Chap. 1].

that we are aware of are particular cases of our method.) We shall also emphasize that although we focused on switched *linear* systems because of our interest in the JSR, the analysis technique of multiple Lyapunov functions on path-complete graphs is clearly valid for switched *nonlinear* systems as well.

We compared the quality of the bound obtained from certain classes of path-complete graphs, including all path-complete graphs with two nodes on an alphabet of two matrices, and also a certain family of dual path-complete graphs. We proposed a specific class of such graphs that appear to work particularly well in practice and proved that the bound obtained from these graphs is invariant under transposition of the matrices and is always within a multiplicative factor of $1/\sqrt[4]{n}$ from the true JSR. Finally, we presented two converse Lyapunov theorems, one for the well-known methods of minimum and maximum-of-quadratics Lyapunov functions, and the other for a new class of methods that propose the use of a common quadratic Lyapunov function for a set of words of possibly different lengths.

We believe the methodology proposed in this chapter should straightforwardly extend to the case of *constrained switching* by requiring the graphs to have a path not for all the words, but only the words allowed by the constraints on the switching. A rigorous treatment of this idea is left for future work.

Vincent Blondel showed that when the underlying automaton is not deterministic, checking path-completeness of a labeled directed graph is an NP-hard problem (personal communication). In general, the problem of deciding whether a non-deterministic finite automaton accepts all finite words is known to be PSPACE-complete [61, p. 265]. However, we are yet to investigate whether the same is true for automata arising from path-complete graphs which have a little more structure. At the moment, the NP-hardness proof of Blondel remains as the strongest negative result we have on this problem. Of course, the step of checking path-completeness of a graph is done offline and prior to the run of our algorithms for approximating the JSR. Therefore, while checking path-completeness is in general difficult, the approximation algorithms that we presented indeed run in polynomial time since they work with a fixed (a priori chosen) path-complete graph. Nevertheless, the question on complexity of checking path-completeness is interesting in many other settings, e.g., when deciding whether a given set of Lyapunov inequalities imply stability of an arbitrary switched system.

Some other interesting questions that can be explored in the future are the following. What are some other classes of path-complete graphs that lead to new techniques for proving stability of switched systems? How can we compare the performance of different path-complete graphs in a systematic way? Given a set of matrices, a class of Lyapunov functions, and a fixed size for the graph, can we efficiently come up with the least conservative topology of a path-complete graph? Within the framework that we proposed, do all the Lyapunov inequalities that prove stability come from path-complete graphs? What are the analogues of the results of this chapter for continuous time switched systems? To what extent do the results carry over to the synthesis (controller design) problem for switched systems? These questions and several others show potential for much follow-up work on path-complete graph Lyapunov functions.

References

[1] A. A. Ahmadi. Non-monotonic Lyapunov functions for stability of nonlinear and switched systems: theory and computation. Master's thesis, Massachusetts Institute of Technology, June 2008. Available from `http://dspace.mit.edu/handle/1721.1/44206`.

[2] A. A. Ahmadi, G. Blekherman, and P. A. Parrilo. Convex ternary quartics are sos-convex. In preparation, 2011.

[3] A. A. Ahmadi, R. Jungers, P. A. Parrilo, and M. Roozbehani. Analysis of the joint spectral radius via Lyapunov functions on path-complete graphs. In *Hybrid Systems: Computation and Control 2011*, Lecture Notes in Computer Science. Springer, 2011.

[4] A. A. Ahmadi, M. Krstic, and P. A. Parrilo. A globally asymptotically stable polynomial vector field with no polynomial Lyapunov function. In *Proceedings of the 50^{th} IEEE Conference on Decision and Control*, 2011.

[5] A. A. Ahmadi, A. Olshevsky, P. A. Parrilo, and J. N. Tsitsiklis. NP-hardness of deciding convexity of quartic polynomials and related problems. *Mathematical Programming*, 2011. Accepted for publication. Online version available at arXiv:.1012.1908.

[6] A. A. Ahmadi and P. A. Parrilo. Non-monotonic Lyapunov functions for stability of discrete time nonlinear and switched systems. In *Proceedings of the 47^{th} IEEE Conference on Decision and Control*, 2008.

[7] A. A. Ahmadi and P. A. Parrilo. A positive definite polynomial Hessian that does not factor. In *Proceedings of the 48^{th} IEEE Conference on Decision and Control*, 2009.

[8] A. A. Ahmadi and P. A. Parrilo. On the equivalence of algebraic conditions

for convexity and quasiconvexity of polynomials. In *Proceedings of the 49^{th} IEEE Conference on Decision and Control*, 2010.

[9] A. A. Ahmadi and P. A. Parrilo. A complete characterization of the gap between convexity and sos-convexity. In preparation, 2011.

[10] A. A. Ahmadi and P. A. Parrilo. Converse results on existence of sum of squares Lyapunov functions. In *Proceedings of the 50^{th} IEEE Conference on Decision and Control*, 2011.

[11] A. A. Ahmadi and P. A. Parrilo. A convex polynomial that is not sos-convex. *Mathematical Programming*, 2011. DOI: 10.1007/s10107-011-0457-z.

[12] A. A. Ahmadi and P. A. Parrilo. On higher order derivatives of Lyapunov functions. In *Proceedings of the 2011 American Control Conference*, 2011.

[13] T. Ando and M.-H. Shih. Simultaneous contractibility. *SIAM Journal on Matrix Analysis and Applications*, 19:487–498, 1998.

[14] A. Andreini, A. Bacciotti, and G. Stefani. Global stabilizability of homogeneous vector fields of odd degree. *Systems and Control Letters*, 10(4):251–256, 1988.

[15] D. Angeli. A note on stability of arbitrarily switched homogeneous systems. 1999. Preprint.

[16] V. I. Arnold. Algebraic unsolvability of the problem of Lyapunov stability and the problem of topological classification of singular points of an analytic system of differential equations. *Functional Analysis and its Applications*, 4(3):173–180. Translated from Funktsional'nyi Analiz i Ego Prilozheniya (1970).

[17] V. I. Arnold. Problems of present day mathematics, XVII (Dynamical systems and differential equations). *Proc. Symp. Pure Math.*, 28(59), 1976.

[18] K. J. Arrow and A. C. Enthoven. Quasi-concave programming. *Econometrica*, 29(4):779–800, 1961.

[19] E. Artin. Über die Zerlegung Definiter Funktionen in Quadrate. *Hamb. Abh.*, 5:100–115, 1927.

[20] E. M. Aylward, S. M. Itani, and P. A. Parrilo. Explicit SOS decomposition of univariate polynomial matrices and the Kalman-Yakubovich-Popov lemma. In *Proceedings of the 46^{th} IEEE Conference on Decision and Control*, 2007.

[21] E. M. Aylward, P. A. Parrilo, and J. J. E. Slotine. Stability and robustness analysis of nonlinear systems via contraction metrics and SOS programming. *Automatica*, 44(8):2163–2170, 2008.

[22] A. Bacciotti and L. Rosier. *Liapunov Functions and Stability in Control Theory*. Springer, 2005.

[23] J. Baillieul. The geometry of homogeneous polynomial dynamical systems. *Nonlinear analysis, Theory, Methods and Applications*, 4(5):879–900, 1980.

[24] S. Basu, R. Pollack, and M. F. Roy. *Algorithms in Real Algebraic Geometry*, volume 10 of *Algorithms and Computation in Mathematics*. Springer-Verlag, Berlin, second edition, 2006.

[25] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty. *Nonlinear Programming*. Wiley-Interscience, 2006. Third edition.

[26] G. Blekherman. Convex forms that are not sums of squares. arXiv:0910.0656, 2009.

[27] G. Blekherman. Nonnegative polynomials and sums of squares. arXiv:1010.3465, 2010.

[28] G. Blekherman and C. B. Riener. Symmetric sums of squares in degree four. In preparation, 2011.

[29] V. D. Blondel, O. Bournez, P. Koiran, C. H. Papadimitriou, and J. N. Tsitsiklis. Deciding stability and mortality of piecewise affine systems. *Theoretical Computer Science*, 255(1-2):687–696, 2001.

[30] V. D. Blondel, O. Bournez, P. Koiran, and J. N. Tsitsiklis. The stability of saturated linear dynamical systems is undecidable. *J. Comput. System Sci.*, 62(3):442–462, 2001.

[31] V. D. Blondel and Y. Nesterov. Computationally efficient approximations of the joint spectral radius. *SIAM J. Matrix Anal. Appl.*, 27(1):256–272, 2005.

[32] V. D. Blondel, Y. Nesterov, and J. Theys. On the accuracy of the ellipsoidal norm approximation of the joint spectral radius. *Linear Algebra Appl.*, 394:91–107, 2005.

[33] V. D. Blondel and J. N. Tsitsiklis. Overview of complexity and decidability results for three classes of elementary nonlinear systems. In *Learning, Control and Hybrid Systems*, pages 46–58. Springer, 1998.

[34] V. D. Blondel and J. N. Tsitsiklis. Complexity of stability and controllability of elementary hybrid system. *Automatica*, 35:479–489, 1999.

[35] V. D. Blondel and J. N. Tsitsiklis. The boundedness of all products of a pair of matrices is undecidable. *Systems and Control Letters*, 41:135–140, 2000.

[36] V. D. Blondel and J. N. Tsitsiklis. A survey of computational complexity results in systems and control. *Automatica*, 36(9):1249–1274, 2000.

[37] O. Bournez and M. L. Campagnolo. A survey on continuous time computations. *New Computational Paradigms*, 4:383–423, 2008.

[38] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[39] M. S. Branicky. Multiple Lyapunov functions and other analysis tools for switched and hybrid systems. *IEEE Transactions on Automatic Control*, 43(4):475–482, 1998.

[40] J. Canny. Some algebraic and geometric computations in PSPACE. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, pages 460–469, New York, 1988. ACM.

[41] G. Chesi and D. Henrion (editors). Special issue on positive polynomials in control. *IEEE Trans. Automat. Control*, 54(5), 2009.

[42] G. Chesi, A. Garulli, A. Tesi, and A. Vicino. Polynomially parameter-dependent Lyapunov functions for robust stability of polytopic systems: an LMI approach. *IEEE Trans. Automat. Control*, 50(3):365–370, 2005.

[43] G. Chesi, A. Garulli, A. Tesi, and A. Vicino. *Homogeneous polynomial forms for robustness analysis of uncertain systems*. Number 390 in Lecture Notes in Control and Information Sciences. Springer, 2009.

[44] G. Chesi and Y. S. Hung. Establishing convexity of polynomial Lyapunov functions and their sublevel sets. *IEEE Trans. Automat. Control*, 53(10):2431–2436, 2008.

[45] M. D. Choi. Positive semidefinite biquadratic forms. *Linear Algebra and its Applications*, 12:95–100, 1975.

[46] M. D. Choi and T. Y. Lam. Extremal positive semidefinite forms. *Math. Ann.*, 231:1–18, 1977.

[47] M. D. Choi, T.-Y. Lam, and B. Reznick. Real zeros of positive semidefinite forms. I. *Math. Z.*, 171(1):1–26, 1980.

[48] R. W. Cottle and J. A. Ferland. On pseudo-convex functions of nonnegative variables. *Math. Programming*, 1(1):95–101, 1971.

[49] J. P. Crouzeix and J. A. Ferland. Criteria for quasiconvexity and pseudoconvexity: relationships and comparisons. *Math. Programming*, 23(2):193–205, 1982.

[50] C. A. R. Crusius. *Automated analysis of convexity properties of nonlinear programs.* PhD thesis, Department of Electrical Engineering, Stanford University, 2002.

[51] N. C. A. da Costa and F. A. Doria. On Arnold's Hilbert symposium problems. In *Computational Logic and Proof Theory*, volume 713 of *Lecture Notes in Computer Science*, pages 152–158. Springer, 1993.

[52] N. C. A. da Costa and F. A. Doria. Undecidable Hopf bifurcation with undecidable fixed point. *International Journal of Theoretical Physics,*, 33(9), 1994.

[53] J. Daafouz and J. Bernussou. Parameter dependent Lyapunov functions for discrete time systems with time varying parametric uncertainties. *Systems and Control Letters*, 43(5):355–359, 2001.

[54] E. de Klerk. The complexity of optimizing over a simplex, hypercube or sphere: a short survey. *CEJOR Cent. Eur. J. Oper. Res.*, 16(2):111–125, 2008.

[55] E. de Klerk and M. Laurent. On the Lasserre hierarchy of semidefinite programming relaxations of convex polynomial optimization problems. Available at `http://www.optimization-online.org/DB-FILE/2010/11/2800.pdf`, 2010.

[56] E. de Klerk and D. V. Pasechnik. Approximation of the stability number of a graph via copositive programming. *SIAM Journal on Optimization*, 12(4):875–892, 2002.

[57] P. H. Diananda. On non-negative forms in real variables some or all of which are non-negative. *Proceedings of the Cambridge philosophical society*, 58:17–25, 1962.

[58] P. Dickinson and L. Gijben. On the computational complexity of membership problems for the completely positive cone and its dual. Available at `http://www.optimization-online.org/DB-FILE/2011/05/3041.pdf`, 2011.

[59] A. C. Doherty, P. A. Parrilo, and F. M. Spedalieri. Distinguishing separable and entangled states. *Physical Review Letters*, 88(18), 2002.

[60] J. A. Ferland. Matrix-theoretic criteria for the quasiconvexity of twice continuously differentiable functions. *Linear Algebra Appl.*, 38:51–63, 1981.

[61] M. R. Garey and D. S. Johnson. *Computers and Intractability.* W. H. Freeman and Co., San Francisco, Calif., 1979.

[62] K. Gatermann and P. A. Parrilo. Symmetry groups, semidefinite programs, and sums of squares. *Journal of Pure and Applied Algebra*, 192:95–128, 2004.

[63] R. Goebel, T. Hu, and A. R. Teel. Dual matrix inequalities in stability and performance analysis of linear differential/difference inclusions. In *Current Trends in Nonlinear Systems and Control*, pages 103–122. 2006.

[64] R. Goebel, A. R. Teel, T. Hu, and Z. Lin. Conjugate convex Lyapunov functions for dual linear differential inclusions. *IEEE Transactions on Automatic Control*, 51(4):661–666, 2006.

[65] D. S. Graca, J. Buescu, and M. L. Campagnolo. Boundedness of the domain of definition is undecidable for polynomial ODEs. In *Proceedings of the Fourth International Conference of Computability and Complexity in Analysis*, 2007.

[66] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 1.21. `http://cvxr.com/cvx`, May 2010.

[67] J. L. Gross and J. Yellen. *Handbook of Graph Theory (Discrete Mathematics and Its Applications).* CRC Press, 2003.

[68] L. Grüne. Homogeneous state feedback stabilization of homogeneous systems. In *Proceedings of the 39^{th} IEEE Conference on Decision and Control*, 2000.

[69] B. Guo. On the difficulty of deciding the convexity of polynomials over simplexes. *International Journal of Computational Geometry and Applications*, 6(2):227–229, 1996.

[70] L. Gurvits. Classical deterministic complexity of Edmonds' problem and quantum entanglement. In *Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing*, pages 10–19 (electronic), New York, 2003. ACM.

[71] N. Gvozdenović and M. Laurent. Semidefinite bounds for the stability number of a graph via sums of squares of polynomials. *Mathematical Programming*, 110(1):145–173, 2007.

[72] W. Hahn. *Stability of Motion*. Springer-Verlag, New York, 1967.

[73] E. Hainry. Decidability and undecidability in dynamical systems. Research report. Available at `http://hal.inria.fr/inria-00429965/PDF/dynsys.pdf`, 2009.

[74] M. A. Hammamia and H. Jerbia. The stabilization of homogeneous cubic vector fields in the plane. *Applied Mathematics Letters*, 7(4):95–99, 1994.

[75] J. W. Helton and J. Nie. Semidefinite representation of convex sets. *Mathematical Programming*, 122(1, Ser. A):21–64, 2010.

[76] D. Henrion and A. Garulli, editors. *Positive polynomials in control*, volume 312 of *Lecture Notes in Control and Information Sciences*. Springer, 2005.

[77] D. Hilbert. Über die Darstellung Definiter Formen als Summe von Formenquadraten. *Math. Ann.*, 32, 1888.

[78] J. E. Hopcroft, R. Motwani, and J. D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison Wesley, 2001.

[79] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 1995.

[80] T. Hu and Z. Lin. Absolute stability analysis of discrete-time systems with composite quadratic Lyapunov functions. *IEEE Transactions on Automatic Control*, 50(6):781–797, 2005.

[81] T. Hu, L. Ma, and Z. Li. On several composite quadratic Lyapunov functions for switched systems. In *Proceedings of the 45^{th} IEEE Conference on Decision and Control*, 2006.

[82] T. Hu, L. Ma, and Z. Lin. Stabilization of switched systems via composite quadratic functions. *IEEE Transactions on Automatic Control*, 53(11):2571 – 2585, 2008.

[83] Z. Jarvis-Wloszek, R. Feeley, W. Tan, K. Sun, and A. Packard. Some controls applications of sum of squares programming. In *Proceedings of the 42^{th} IEEE Conference on Decision and Control*, pages 4676–4681, 2003.

[84] R. G. Jeroslow. There cannot be an algorithm for integer programming with quadratic constraints. *Operations Research*, 21(1):221–224, 1973.

[85] R. Jungers. *The joint spectral radius: theory and applications*, volume 385 of *Lecture Notes in Control and Information Sciences*. Springer, 2009.

[86] H. Khalil. *Nonlinear systems*. Prentice Hall, 2002. Third edition.

[87] V. L. Kharitonov. Asymptotic stability of an equilibrium position of a family of systems of linear differential equations. *Differential Equations*, 14:1483–1485, 1979.

[88] M. Kojima. Sums of squares relaxations of polynomial semidefinite programs. *Research report B-397, Dept. of Mathematical and Computing Sciences. Tokyo Institute of Technology*, 2003.

[89] J. B. Lasserre. Convexity in semialgebraic geometry and polynomial optimization. *SIAM Journal on Optimization*, 19(4):1995–2014, 2008.

[90] J. B. Lasserre. Representation of nonnegative convex polynomials. *Archiv der Mathematik*, 91(2):126–130, 2008.

[91] J. B. Lasserre. Certificates of convexity for basic semi-algebraic sets. *Applied Mathematics Letters*, 23(8):912–916, 2010.

[92] L. J. Lau. Testing and imposing monotonicity, convexity and quasiconvexity constraints. In M. A. Fuss and D. L. McFadden, editors, *Production Economics: A Dual Approach to Theory and Applications*, pages 409–453. North-Holland Publishing Company, 1978.

[93] J. W. Lee and G. E. Dullerud. Uniform stabilization of discrete-time switched and Markovian jump linear systems. *Automatica*, 42(2):205–218, 2006.

[94] H. W. Lenstra. Integer programming with a fixed number of variables. *Mathematics of Operations Research*, 8(4):538–548, 1983.

[95] H. Lin and P. J. Antsaklis. Stability and stabilizability of switched linear systems: a short survey of recent results. In *Proceedings of IEEE International Symposium on Intelligent Control*, 2005.

[96] D. Lind and B. Marcus. *An Introduction to Symbolic Dynamics and Coding.* Cambridge University Press, 1995.

[97] C. Ling, J. Nie, L. Qi, and Y. Ye. Biquadratic optimization over unit spheres and semidefinite programming relaxations. *SIAM Journal on Optimization*, 20(3):1286–1310, 2009.

[98] J. Löfberg. Yalmip : A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, 2004. Available from `http://control.ee.ethz.ch/~joloef/yalmip.php`.

[99] A. M. Lyapunov. *General problem of the stability of motion.* PhD thesis, Kharkov Mathematical Society, 1892. In Russian.

[100] A. Magnani, S. Lall, and S. Boyd. Tractable fitting with convex polynomials via sum of squares. In *Proceedings of the 44^{th} IEEE Conference on Decision and Control*, 2005.

[101] O. L. Mangasarian. Pseudo-convex functions. *J. Soc. Indust. Appl. Math. Ser. A Control*, 3:281–290, 1965.

[102] O. L. Mangasarian. *Nonlinear Programming.* SIAM, 1994. First published in 1969 by the McGraw-Hill Book Company, New York.

[103] P. Mason, U. Boscain, and Y. Chitour. Common polynomial Lyapunov functions for linear switched systems. *SIAM Journal on Optimization and Control*, 45(1), 2006.

[104] P. Mereau and J. G. Paquet. Second order conditions for pseudo-convex functions. *SIAM J. Appl. Math.*, 27:131–137, 1974.

[105] A. Molchanov and Y. Pyatnitskiy. Criteria of asymptotic stability of differential and difference inclusions encountered in control theory. *Systems and Control Letters*, 13:59–64, 1989.

[106] L. Moreau, D. Aeyels, J. Peuteman, and R. Sepulchre. Homogeneous systems: stability, boundedness and duality. In *Proceedings of the 14th Symposium on Mathematical Theory of Networks and Systems*, 2000.

[107] T. S. Motzkin. The arithmetic-geometric inequality. In *Inequalities (Proc. Sympos. Wright-Patterson Air Force Base, Ohio, 1965)*, pages 205–224. Academic Press, New York, 1967.

[108] T. S. Motzkin and E. G. Straus. Maxima for graphs and a new proof of a theorem of Turán. *Canadian Journal of Mathematics*, 17:533–540, 1965.

[109] K. G. Murty and S. N. Kabadi. Some NP-complete problems in quadratic and nonlinear programming. *Mathematical Programming*, 39:117–129, 1987.

[110] A. Nemirovskii. Several NP-hard problems arising in robust stability analysis. *Mathematics of Control, Signals, and Systems*, 6:99–105, 1993.

[111] Y. E. Nesterov and A. Nemirovski. *Interior point polynomial methods in convex programming*, volume 13 of *Studies in Applied Mathematics*. SIAM, Philadelphia, PA, 1994.

[112] J. Nie. Polynomial matrix inequality and semidefinite representation. arXiv:0908.0364., 2009.

[113] J. Nie, J. Demmel, and B. Sturmfels. Minimizing polynomials via sum of squares over the gradient ideal. *Mathematical Programming*, 106(3, Ser. A):587–606, 2006.

[114] A. Papachristodoulou and S. Prajna. On the construction of Lyapunov functions using the sum of squares decomposition. In *IEEE Conference on Decision and Control*, 2002.

[115] C. H. Papadimitriou. On the complexity of integer programming. *Journal of the ACM*, 28:765–768, 1981.

[116] P. M. Pardalos, editor. *Complexity in Numerical Optimization*. World Scientific Publishing Co. Inc., River Edge, NJ, 1993.

[117] P. M. Pardalos and S. A. Vavasis. Open questions in complexity theory for numerical optimization. *Mathematical Programming*, 57(2):337–339, 1992.

[118] P. A. Parrilo. *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization.* PhD thesis, California Institute of Technology, May 2000.

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

[120] P. A. Parrilo. Polynomial games and sum of squares optimization. In *Proceedings of the 45^{th} IEEE Conference on Decision and Control*, 2006.

[121] P. A. Parrilo. On a decomposition of multivariable forms via LMI methods. *American Control Conference, 2000.*, 1(6):322–326 vol.1, Sep 2000.

[122] P. A. Parrilo and A. Jadbabaie. Approximation of the joint spectral radius using sum of squares. *Linear Algebra Appl.*, 428(10):2385–2402, 2008.

[123] P. A. Parrilo and R. Peretz. An inequality for circle packings proved by semidefinite programming. *Discrete and Computational Geometry*, 31(3):357–367, 2004.

[124] P. A. Parrilo and B. Sturmfels. Minimizing polynomial functions. *Algorithmic and Quantitative Real Algebraic Geometry, DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 60:83–99, 2003.

[125] M. M. Peet. Exponentially stable nonlinear systems have polynomial Lyapunov functions on bounded regions. *IEEE Trans. Automat. Control*, 54(5):979–987, 2009.

[126] M. M. Peet and A. Papachristodoulou. A converse sum of squares Lyapunov result: an existence proof based on the Picard iteration. In *Proceedings of the 49^{th} IEEE Conference on Decision and Control*, 2010.

[127] M. M. Peet and A. Papachristodoulou. A converse sum of squares Lyapunov result with a degree bound. *IEEE Trans. Automat. Control*, 2011. To appear.

[128] A. Pfister and C. Scheiderer. An elementary proof of Hilbert's theorem on ternary quartics. arXiv:1009.3144, 2010.

[129] B. Poonen. Undecidability in number theory. *Notices of the Amer. Math. Soc.*, 55(3):344–350, 2008.

[130] V. Powers, B. Reznick, C. Scheiderer, and F. Sottile. A new approach to Hilbert's theorem on ternary quartics. *Comptes Rendus Mathematique*, 339(9):617 – 620, 2004.

[131] S. Prajna and A. Papachristodoulou. Analysis of switched and hybrid systems – beyond piecewise quadratic methods. In *Proceedings of the American Control Conference*, 2003.

[132] S. Prajna, A. Papachristodoulou, and P. A. Parrilo. *SOSTOOLS: Sum of squares optimization toolbox for MATLAB*, 2002-05. Available from `http://www.cds.caltech.edu/sostools` and `http://www.mit.edu/~parrilo/sostools`.

[133] S. Prajna, P. A. Parrilo, and A. Rantzer. Nonlinear control synthesis by convex optimization. *IEEE Trans. Automat. Control*, 49(2):310–314, 2004.

[134] V. Y. Protasov, R. M. Jungers, and V. D. Blondel. Joint spectral characteristics of matrices: a conic programming approach. *SIAM Journal on Matrix Analysis and Applications*, 31(4):2146–2162, 2010.

[135] L. Rademacher and S. Vempala. Testing geometric convexity. In *FSTTCS 2004: Foundations of Software Technology and Theoretical Computer Science*, volume 3328 of *Lecture Notes in Comput. Sci.*, pages 469–480. Springer, Berlin, 2004.

[136] M. J. A. Rantzer. Computation of piecewise quadratic Lyapunov functions for hybrid systems. *IEEE Trans. Automat. Control*, 43(4):555–559, 1998.

[137] B. Reznick. Uniform denominators in Hilbert's 17th problem. *Math Z.*, 220(1):75–97, 1995.

[138] B. Reznick. Some concrete aspects of Hilbert's 17th problem. In *Contemporary Mathematics*, volume 253, pages 251–272. American Mathematical Society, 2000.

[139] B. Reznick. On Hilbert's construction of positive polynomials. arXiv:0707.2156., 2007.

[140] B. Reznick. Blenders. arXiv:1008.4533, 2010.

[141] R. M. Robinson. Some definite polynomials which are not sums of squares of real polynomials. In *Selected questions of algebra and logic (collection dedicated to the memory of A. I. Mal′cev) (Russian)*, pages 264–282. Izdat. "Nauka" Sibirsk. Otdel., Novosibirsk, 1973.

[142] R. T. Rockafellar. *Convex Analysis.* Princeton University Press, Princeton, New Jersey, 1970.

[143] R. T. Rockafellar. Lagrange multipliers and optimality. *SIAM Review*, 35:183–238, 1993.

[144] M. Roozbehani. *Optimization of Lyapunov invariants in analysis and implementation of safety-critical software systems.* PhD thesis, Massachusetts Institute of Technology, 2008.

[145] M. Roozbehani, A. Megretski, E. Frazzoli, and E. Feron. Distributed Lyapunov functions in analysis of graph models of software. *Springer Lecture Notes in Computer Science*, 4981:443–456, 2008.

[146] L. Rosier. Homogeneous Lyapunov function for homogeneous continuous vector fields. *Systems and Control Letters*, 19(6):467–473, 1992.

[147] G. C. Rota and W. G. Strang. A note on the joint spectral radius. *Indag. Math.*, 22:379–381, 1960.

[148] W. Rudin. *Real and complex analysis.* McGraw-Hill series in higher mathematics, 1987. Third edition.

[149] N. Samardzija. Stability properties of autonomous homogeneous polynomial differential systems. *Journal of Differential Equations*, 48(1):60–70, 1983.

[150] C. Scheiderer. Sums of squares on real algebraic surfaces. *Manuscr. Math.*, 119(4):395–410, 2006.

[151] C. Scheiderer. A Positivstellensatz for projective real varieties. arXiv:1104.1772, 2011.

[152] C. W. Scherer and C. W. J. Hol. Matrix sum of squares relaxations for robust semidefinite programs. *Mathematical Programming*, 107:189–211, 2006.

[153] A. Schrijver. *Theory of Linear and Integer Programming.* John Wiley & sons, 1998.

[154] A. Seidenberg. A new decision method for elementary algebra. *Ann. of Math. (2)*, 60:365–374, 1954.

[155] N. Z. Shor. Class of global minimum bounds of polynomial functions. *Cybernetics*, 23(6):731–734, 1987. (Russian orig.: Kibernetika, No. 6, (1987), 9–11).

[156] E. D. Sontag. From linear to nonlinear: some complexity comparisons. In *Proceedings of the 34^{th} IEEE Conference on Decision and Control*, 1995.

[157] J. Sturm. *SeDuMi version 1.05*, Oct. 2001. Latest version available at `http://sedumi.ie.lehigh.edu/`.

[158] A. Tarski. *A decision method for elementary algebra and geometry*. University of California Press, Berkeley and Los Angeles, Calif., 1951. Second edition.

[159] R. Tedrake, I. R. Manchester, M. M. Tobenkin, and J. W. Roberts. LQR-Trees: Feedback motion planning via sums of squares verification. *International Journal of Robotics Research*, 29:1038–1052, 2010.

[160] A. L. Tits. Lecture notes on optimal control. Available from `http://www.isr.umd.edu/~andre/664.pdf`, 2008.

[161] J. N. Tsitsiklis and V. Blondel. The Lyapunov exponent and joint spectral radius of pairs of matrices are hard- when not impossible- to compute and to approximate. *Mathematics of Control, Signals, and Systems*, 10:31–40, 1997.

[162] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM Review*, 38(1):49–95, Mar. 1996.