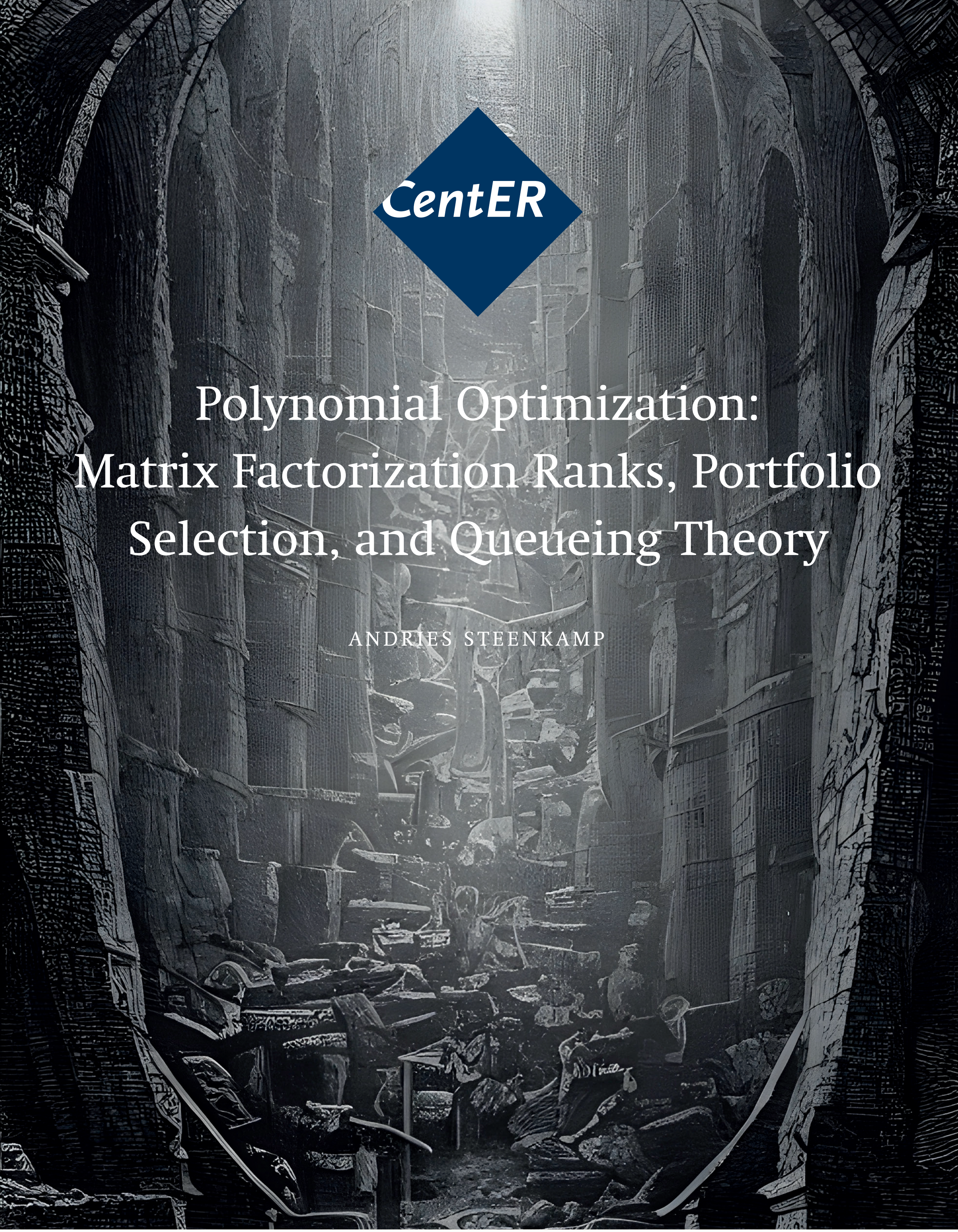


Polynomial Optimization: Matrix Factorization Ranks, Portfolio Selection, and Queueing Theory

ANDRIES STEENKAMP

Polynomial Optimization: Matrix Factorization Ranks, Portfolio Selection, and Queueing Theory

Proefschrift ter verkrijging van de graad van doctor aan Tilburg University op gezag van de rector magnificus, prof. dr. W.B.H.J. van de Donk, in het openbaar te verdedigen ten overstaan van een door het college voor promoties aangewezen commissie in de Aula van de Universiteit op

vrijdag 27 oktober 2023 om 10.00 uur

door

Johannes Andries Jacobus Steenkamp,

geboren te Pretoria, Zuid-Afrika.

PROMOTORES: prof. dr. M. Laurent (Tilburg University)
prof. dr. E. de Klerk (Tilburg University)

PROMOTIECOMMISSIE: prof. dr. D. Henrion (LAAS-CNRS Toulouse and
Czech Technical University in Prague)
dr. D. de Laat (Delft University of Technology)
prof. dr. J.S.H. van Leeuwaarden (Tilburg University)
prof. dr. ir. R. Sotirov (Tilburg University)
dr. J.C. Vera Lizcano (Tilburg University)

This research was carried out at the Centrum Wiskunde & Informatica (CWI) in Amsterdam.
This research has received funding from the European Union's Framework Programme for Research and Innovation Horizon 2020 under the Marie Skłodowska-Curie Actions Grant Agreement No. 813211 (POEMA).

Acknowledgments

Just as a single hoplite does not make a phalanx, a single student does not make research. I owe much of my achievements in these past four years to the patient support of others. I now give thanks to them in no particular order.

To my promotors, mentors, and advisors, Monique Laurent and Etienne de Klerk, I am grateful for your unwavering support and for setting an uncompromising standard of excellence.

I thank my Ph.D. committee members, Didier Henrion, David de Laat, Johan van Leeuwaarden, Renata Sotirov, and Juan Vera Lizcano, for their valuable comments, insights, and questions about my work.

Next, I want to thank my co-authors Victor Magron, Milan Korda, Sander Gribling, and Daniel Brosch, without whom research would have been much harder and dull. In particular, I thank Victor for hosting my stay in France and requisitioning extra rations for me at the canteen.

I give thanks to my predecessors, Lucas and Sven: You have saved me a lot of effort by walking the path before me and warning me of the pitfalls that lay and wait for the unwary Ph.D. student.

To my POEMA cohorts: Though the pandemic and the passage of time have all but dissolved our bonds, I look fondly upon the comradery we forged in Florence. I give particular thanks to my office mate Luis; knowing that you must endure the same administrations as me bolstered my resolve. I thank my academic half-brother Felix for being a kindred spirit in the search for strength.

A special thanks is given to my colleagues at CWI for breaking bread with me and humoring my tall tales of the dark continent. In particular, I mention Konstantinos for adding several tomes on history, war, and atrocities to my growing pile of yet-to-read books. Simon, for making me more cultured with his excellent taste in Belgian beer and authentic Italian pizza. Samarth, for showing me proper squatting form and the *pranayama* of bending iron.

Finally, I give my greatest gratitude to my mother for courageously sending me off to Europe and starting my strange quest.

Contents

Introduction

According to Leonhard Euler: "*In der Welt geschieht nichts, worin man nicht den Sinn eines bestimmten Maximums oder Minimums erkennen könnte.*"[1] In this spirit, I have written my Ph.D., with optimization as the golden thread that links the four major parts of my thesis.

Mathematical optimization is an umbrella term for the study of maximizing (resp., minimizing) a quantity of interest subject to constraints. On the most basic level, mathematical optimization problems have two parts: an objective function that quantifies improvement due to the choice of input, and a set of valid inputs. The type of an optimization problem is determined by the nature of its objective function and feasible region. For example, polynomial optimization problems optimize a polynomial objective over a domain defined by polynomial constraints. Often the objective and domain of optimization are determined by some underlying model or real-world problem of interest.

Matrix factorization ranks

As an example of a mathematical optimization problem, consider the completely positive rank of a completely positive matrix. A symmetric nonnegative matrix $A \in \mathbb{R}_+^{n\times n}$ is called *completely positive* (CP) if there exists nonnegative vectors $\mathbf{a}_1, ..., \mathbf{a}_r \in \mathbb{R}^n_+$ with the property that $A = \sum_{\ell\in[r]} \mathbf{a}_\ell \mathbf{a}_\ell^T$. The *completely positive rank* of a CP matrix A, denoted $\mathrm{rank}_{\mathrm{cp}}(A)$, is the smallest positive integer $r \in \mathbb{N}$ for which there exist nonnegative vectors $\mathbf{a}_1, ..., \mathbf{a}_r \in \mathbb{R}^n_+$ with the property that $A = \sum_{\ell\in[r]} \mathbf{a}_\ell \mathbf{a}_\ell^T$. Formulated as an optimization problem, the CP rank reads as follows:

$$\mathrm{rank}_{\mathrm{cp}}(A) := \min\Big\{ r \in \mathbb{N} : \mathbf{a}_1, ..., \mathbf{a}_r \in \mathbb{R}^n_+,\ A = \sum_{i\in[r]} \mathbf{a}_\ell \mathbf{a}_\ell^T \Big\}.$$

If A is not CP, we set $\mathrm{rank}_{\mathrm{cp}}(A) = \infty$. The set of all $n \times n$ CP matrices form a convex cone, denoted by $\mathcal{CP}^n$.

Building on an earlier result from 1965 by Motzkin and Straus [**43**], de Klerk and Pasechnik [**43**, Theorem 2.2] showed in 2002 that the problem of

[1] Nothing happens in this world in which one could not recognize the meaning of a certain maximum or minimum.

computing the stability number $\alpha(G)$ of a graph $G := ([n], E)$ could be recast as an optimization problem over $\mathcal{CP}^n$. Burer [**28**] expanded on this result in 2009 by showing that any nonconvex quadratic program with binary and continuous variables could be reformulated as a linear program over the cone of CP matrices. This effectively meant that many NP-hard problems could now be viewed as linear programs with CP membership constraints.

In statistics, CP matrices are linked to the theory of *block designs*; see [**145**]. Block designs are used in many applications where systematic comparisons are being made. For example, when a researcher wishes to test the efficacy of different treatments (like the protection offered by different sunscreen lotions) on a group of test subjects (like a random sample of the human population), the researcher could group test subjects by the treatment they receive (assigning them to a "block") in such a way that the inherent differences between the test subjects (like age and ethnicity) do not skew the results of the different treatments; see [**83**]. Hence, finding a "minimal" block design would correspond to one that requires the fewest test subjects.

We have dedicated Part 2 of this thesis to studying the CP rank and several other matrix factorization ranks, such as the nonnegative rank and separable rank.

Polynomial optimization

As a sub-topic of optimization, we consider polynomial optimization, where the objective function is a polynomial, and the set of valid inputs is a semialgebraic set characterized by a finite system of polynomial constraints, i.e.,

$$\begin{aligned}&\inf f(\mathbf{x})\\ &\text{s.t. } \mathbf{x} \in K := \{\mathbf{x} \in \mathbb{R}^n : g_i(\mathbf{x}) \geq 0 \ (i \in [N_g]),\ h_j(\mathbf{x}) = 0 \ (j \in [N_h])\}.\end{aligned}$$

Here, $f, g_1, ..., g_{N_g}, h_1, ..., h_{N_h} \in \mathbb{R}[\mathbf{x}] := \mathbb{R}[x_1, x_2, ..., x_n]$ are polynomials.

Polynomial optimization already provides a rich enough framework to capture many pertinent problems like matrix factorization rank, portfolio selection, and some optimization problems arising in queueing theory. Another important industrial application of polynomial optimization is to the optimal power flow problem; see, e.g., [**9**] and the references therein. The primary problems in Parts 3 and 4 of this thesis boil down to dealing with particular classes of polynomial optimization problems.

Because of its expressive power, polynomial optimization contains many NP-hard problems. Hence, polynomial optimization problems are often difficult to solve. A notable exception is when the problem is convex, i.e., when the objective function is convex, and the domain of integration is convex. In this case, provided we have some additional information, like the gradient of the objective and an efficient means of projecting onto the feasible region, we can use powerful and well-studied first-order methods to optimize the problem. We leverage these techniques in Part 3, where the crux of our solution lies in showing that a multi-objective problem of interest can be partially solved by solving a collection of (scalar-objective) problems, many of which are convex. In Part 4, the core result is again showing convexity, though this time requiring tools from matrix algebra theory.

The mean-variance-skewness-kurtosis problem. As an example of a polynomial optimization problem, consider the problem

$$\begin{aligned} &\min \lambda_1 f_1(w) + \lambda_2 f_2(w) + \lambda_3 f_3(w) + \lambda_4 f_4(w) \\ &\text{s.t. } w \in \Delta^n := \{w \in \mathbb{R}^n : w_i \geq 0, \sum_{i \in [n]} w_i = 1\}, \end{aligned} \tag{0.1}$$

for some polynomials $f_1, f_2, f_3, f_4 \in \mathbb{R}[w]$ and a fixed parameter $\lambda \in \mathbb{R}^4$. This problem emerges in Part 3, where it is used to recover Pareto optimal solutions to the mean-variance-skewness-kurtosis (MVSK) problem in finance. The MVSK problem is a particular model for the problem of portfolio optimization, where one is tasked with selecting a subset of assets (called a portfolio) from a pool of available assets in such a way as to maximize the appreciation of the selection's value while minimizing the risk of losing the initial capital investment.

In Part 3, the polynomials f_1 and f_3 will model the expected returns on investment, the polynomials f_2 and f_4 will model the risk of monetary loss, and the parameter $\lambda \in \mathbb{R}^4$ will be chosen to represent the investor's preferences in balancing these conflicting objectives. For $k \in [4]$ we have $\deg(f_k) = k$, hence (0.1) is a quartic optimization problem over the simplex.

Our core contribution to this topic is the characterization of a large class of $\lambda \in \Delta^4$ for which (0.1) becomes a convex optimization problem. This seemingly simple result is either not mentioned or assumed not to hold; see, e.g., [**92, 95, 117, 120, 147, 174**]. To the best of our knowledge, this convexity result does not appear to be known in the literature on the MVSK problem. We also provide peripheral results on finding sparse solutions.

The minimum of a graph-based polynomial from queueing theory. In Part 4, we consider two classes of polynomials that have significance in queueing theory, in particular, with regard to the asymptotic behavior of a parallel-server system's job occupancy with redundancy scheduling. Let

$$E := \{e \subseteq [n] : |e| = L\}$$

be the set of all edges of a complete L-uniform hypergraph on n elements. We are tasked with proving that the optimal value of

$$\begin{aligned} \min\, p_d(x) &:= \sum_{(e_1,\dots,e_d)\in E^d} \frac{1}{|e_1\cup\ldots\cup e_d|} x_{e_1}\cdots x_{e_d} \\ \text{s.t. } x\in\Delta_m &:= \Big\{x=(x_e)_{e\in E}\in\mathbb{R}^m : x\geq 0, \sum_{e\in E} x_e = 1\Big\}, \end{aligned} \tag{0.2}$$

is attained at the *barycenter* $x^* := \frac{1}{m}(1,\dots,1)$ of Δ_m for all $d\in\mathbb{N}$ and L. This is done by exploiting the symmetry properties of $p_d(x)$ and the fact that p_d is convex over the simplex Δ_m. We prove that p_d is convex by showing that the Hessian $H(p_d)$ is positive semidefinite over Δ_m. Proving the PSDness of the Hessian $H(p_d)$ is the main result of Part 4, to which we dedicate Chapter 12. The proof proceeds with several PSDness preserving reductions of $H(p_d)$ into smaller matrices, which are then shown to belong to the Terwilliger algebra of the binary Hamming cube. We prove these final matrices are PSD using classical results from Artin, Wedderburn [**166**], and Schrijver [**144**]. For a more modern treatment of matrix algebras, we refer the reader to the thesis of Dion Gijswit [**73**] and the references therein.

The other (more important) class of polynomials f_d (which we do not define here) is then investigated in Chapter 13, where it is observed that f_d also obtains its global optimum at the barycenter provided f_d is convex over Δ_m. Except for a few special cases, we fail to prove that f_d is convex in general. Still, we show that $H(p_d)$ appears in the Hessian $H(f_d)$ in some intricate way, laying the foundation for future research (like that of [**131**]) into this problem.

Generalized moment problems

A way to circumvent the computational difficulty of NP-hard problems is to consider related or relaxed problems, which are easier to solve than the original problem, and whose optimal values approximate that of the original problem. A prime example is the moment method applied to generalized moment problems (GMPs). First, we briefly describe moment problems and then give the moment method's gist.

Moment problems have been actively studied for at least a century, and as such, the field is very rich and broad in applications; see, e.g., Akhiezer [**4**], Schmüdgen [**142**], and Lasserre [**106**]. For a recent survey, see [**42**].

We focus on generalized moment problems from the perspective of linear optimization problems over measures, which contains polynomial optimization as a special case. The study of GMPs should be understood in contradistinction to "the moment problem", which is a related classical topic where one seeks a representing measure for a given (partial) set of moments.

Building on the example of the CP rank, we consider the following generalized moment problem, first defined by Fawzi and Parilo in [**67**]:

$$\tau_{\mathrm{cp}}(A) := \inf_{\mu \in \mathscr{M}(K_A)} \Big\{ \int_{K_A} 1 d\mu : \int_{K_A} x_i x_j d\mu = A_{ij} \ (i,j \in [n]) \Big\}. \tag{0.3}$$

Here, one optimizes over positive Borel measures μ supported on the semialgebraic set K_A, which is defined as

$$K_A = \Big\{ x \in \mathbb{R}^n : \begin{array}{l} \sqrt{A_{ii}} x_i - x_i^2 \geq 0 \ (i \in [n]), \\ A_{ij} - x_i x_j \geq 0 \ (i < j \in [n]), \\ A - \mathbf{x}\mathbf{x}^T \succeq 0 \Big\}. \end{array}$$

The GMP (0.3) has an optimal value that lower bounds the CP rank of A. To see this, take any optimal CP factorization $A = \sum_{\ell \in [r]} \mathbf{a}_\ell \mathbf{a}_\ell^T$ of matrix A and define the measure $\mu := \sum_{\ell \in [r]} \delta_{\mathbf{a}_\ell}$, where $\delta_{\mathbf{a}_\ell}$ is the Dirac delta measure centered at $\mathbf{a}_\ell \in K_A$. Some inspection will show that μ is a feasible solution to (0.3) with $\int_{K_A} 1 d\mu = r = \mathrm{rank}_{\mathrm{cp}}(A)$. Thus we have

$$\tau_{\mathrm{cp}}(A) \leq \mathrm{rank}_{\mathrm{cp}}(A).$$

Since GMPs capture polynomial optimization, they are generally hard to solve. So, we must look at approximate solutions by considering relaxed problems related to the GMP.

The moment method. The core idea of the moment method is to recast a GMP (like the one in (0.3)) in terms of a linear functional L and then to impose positivity conditions on L that are necessary for L to have a representing measure. The classical result of Putinar (see Theorem 2.8) provides such necessary conditions. It states that a linear functional L has a representing measure μ supported on the semialgebraic set

$$\mathscr{D}(H) := \Big\{ \mathbf{x} \in \mathbb{R}^n : g(\mathbf{x}) \geq 0, \text{ for all } g \in H \Big\},$$

where $H \subseteq \mathbb{R}[\mathbf{x}]$ is some set of polynomials, provided the associated quadratic module

$$\mathcal{M}(H) := \mathrm{cone} \Big\{ g p \overline{p} : p \in \mathbb{R}[\mathbf{x}], \ g \in H \cup \{1\} \Big\}$$

is Archimedean (i.e., $R - \sum_{i=1}^n x_i^2 \in \mathcal{M}(H)$ for some $R > 0$) and L is positive on $\mathcal{M}(H)$. By relaxing the positivity of L on the quadratic module $\mathcal{M}(H)$ to only positivity on the truncated quadratic module

$$\mathcal{M}_{2t}(H) := \mathrm{cone} \Big\{ g p \overline{p} : p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}], \ g \in H \cup \{1\}, \ \deg(g p \overline{p}) \leq 2t \Big\},$$

for some $t \in \mathbb{N}$, one obtains lower bounds on the optimal value for the GMP.

These relaxations form a hierarchy of semidefinite programs, with each level t in the hierarchy corresponding to a different order of truncation. Applying the moment method, for every $t \in \mathbb{N} \cup \{\infty\}$, to the GMP in (0.3) we get the hierarchy of SDPs

$$\begin{aligned}\xi_t^{\mathrm{cp}}(A) := \min\Big\{L(1) :\, & L \in \mathbb{R}[\mathbf{x}]_{2t}^*,\\ & L(\mathbf{x}\mathbf{x}^T) = A,\\ & L([\mathbf{x}]_t[\mathbf{x}]_t^T) \succeq 0,\\ & L((\sqrt{A_{ii}}x_i - x_i^2)[\mathbf{x}]_{t-1}[\mathbf{x}]_{t-1}^T) \succeq 0 \ (i \in [n]),\\ & L((A_{ij} - x_ix_j)[\mathbf{x}]_{t-1}[\mathbf{x}]_{t-1}^T) \succeq 0 \ (1 \le i < j \le n),\\ & L((A - \mathbf{x}\mathbf{x}^T) \otimes [\mathbf{x}]_{t-1}[\mathbf{x}]_{t-1}^T) \succeq 0\Big\}.\end{aligned}$$

Using the general results we develop in Sections 3.1.1 and 3.1.2, we show in Section 6.1.2 that this particular hierarchy satisfies

$$\xi_1^{\mathrm{cp}}(A) \le \xi_2^{\mathrm{cp}}(A) \le \cdots \le \xi_\infty^{\mathrm{cp}}(A) = \tau_{\mathrm{cp}}(A).$$

These parameters $\xi_t^{\mathrm{cp}}(A)$ can often be computed for reasonably sized matrices A and values t. Constructing these hierarchies and exploring the bounds they provide in the context of matrix factorization ranks is our core contribution in Part 2.

The Achilles' heel of the moment method is the exponential growth of the moment matrices defining the SDPs of the hierarchy. To combat this, we developed a technique we call *ideal sparsity*, that exploits a special structure in the GMP.

Ideal sparsity. Consider a GMP of the particular form

$$\mathbf{val} := \inf_{\mu \in \mathscr{M}(K)} \Big\{ \int f_0 d\mu : \int f_i d\mu = a_i \ (i \in [N_f])\Big\},$$
$$K = \Big\{\mathbf{x} \in \mathbb{R}^n : g_j(\mathbf{x}) \ge 0 \ (j \in [N_g]),\ \prod_{i \in S} x_i = 0 \ (S \in \mathcal{S})\Big\},$$

where $\mathscr{M}(K)$ is the set of positive Borel measures supported on K, f_0,f_1,...,f_{N_f}, $g_1, ..., g_{N_g} \in \mathbb{R}[\mathbf{x}]$ are polynomials, $a_1, ..., a_{N_f} \in \mathbb{R}$ are scalars, and $\mathcal{S} \subseteq \mathcal{P}([n])$. The distinguishing feature of this GMP is the presence of the particular ideal constraint requiring $\mathrm{supp}(\mu) \subseteq \{\mathbf{x} \in \mathbb{R}^n : \prod_{i \in S} x_i = 0 \ (S \in \mathcal{S})\}$ in the definition of K.

In Chapter 2, we show that this GMP has an equivalent sparse reformulation, where the single (high-dimensional) measure variable is replaced by several (lower-dimensional) measure variables. Even though the resulting ideal sparse GMP is equivalent, its associate ideal sparse hierarchy is both more economical in terms of the involved matrix sizes and in terms of the quality of bounds it provides.

This result stands in contradistinction to other sparsity techniques, where one often sacrifices the bound strength in exchange for a computational speed-up. We demonstrate the significant improvement due to ideal sparsity with applications to the CP rank (Chapter 6) and nonnegative rank (Chapter 5).

Similarly, we consider a “block-diagonal reduction” in Section 7.2 to make the SDP hierarchy associated with the separable rank (Chapter 7) more efficient.

Societal and scientific relevance

Matrix factorization is a powerful mathematical tool that significantly impacts a wide range of applications in today’s society. From recommendation systems and data mining to image processing and natural language processing, matrix factorization techniques play a crucial role in various algorithms. It is prized for its ability to compress data, reduce dimensionality, and its ease of interpretation. Interpretability is becoming ever more appreciated as several prominent machine learning algorithms essentially function as “black boxes,” obfuscating their inner workings and casting suspicion on their conclusions. Matrix factorization ranks can be seen as quantifying the complexity of the data represented as a matrix. However, some of these matrix factorization ranks are difficult to compute. Hence, there is a need to find accurate lower bounds that are easier to compute.

Moments problems are an extremely rich field of study with several applications like global optimization, Markov chains, optimal control, and multivariate integration, to name a few. Generalized moment problems, in particular, are often used to attack optimization problems in many diverse contexts. One example that stands out is the estimation of Lipschitz constants for ReLU networks, thereby gauging the robustness of the network. This has been done using semialgebraic optimization and could potentially benefit from ideal sparsity.

Portfolio selection is at the heart of wealth management, both on the individual investor’s level and the level of large institutions. To wisely allocate investments based on the deluge of information available today, it is essential to have data-driven models that do not rely solely on human judgment. Portfolio optimization codifies the portfolio selection task as a mathematical optimization problem, thereby quantifying risks and rewards while balancing them against each other.

Queueing theory is an imminently practical subtopic of operations research that studies the allocation of incoming jobs (a queue of jobs) to a collection of servers (things that complete jobs). The goal is often one of optimization, where one, for example, wants to maximize the number of jobs completed in a fixed time frame or minimize the time a server spends idling. The goals are often set in accordance with some business decisions. Historically, queueing theory was used in project management and industrial engineering. Due to the popularity of the Internet, online services and website traffic have become popular use cases. Today, queueing theory also has applications in (but not limited to) logistics, health care, service operations management, revenue management, theoretical economics, and pricing.

Organization

This thesis is organized into four parts.

Part 1 contains preliminaries (Chapter 1) followed by an introduction to generalized moment problems (Chapter 2). Ideal sparsity is a subtopic of GMPs (Section 2.2.1), which we developed in [**100**]. The moment method and associated fundamental results from the literature are provided in Chapter 3, where we adapt them for their use in Part 2 of the thesis, which is on matrix factorization ranks (MFR). In the moment method, we look, in particular, at polynomial matrix localizing constraints, which have a natural application in MFR. Part 1 is mostly written for complex polynomials as this will be required in Chapter 7 for the separable rank. However, real analogs are also provided where applicable, and the core results are stated in both variants for clarity and convenience.

Part 2 is dedicated to MFR. We give a general overview of matrix factorization in Chapter 4, which is based on our book chapter [**151**]. Special focus is given to the nonnegative rank (Chapter 5), the completely positive rank (Chapter 6), and the separable rank (Chapter 7), where we apply the moment method described in Chapter 3. In particular, we apply ideal sparsity to the hierarchies associated with nonnegative rank and completely positive rank, and we apply a block-diagonalization technique to the separable rank.

Part 3 is devoted to the MVSK problem in portfolio optimization. This part of the thesis stands on its own and is based on our work in [**150**]. Chapter 8 gives some finance theory background on the MVSK problem and some preliminaries on multi-objective optimization. We follow up in Chapter 9 with the mathematical formulation of MVSK as a multi-objective optimization problem. We attack the MVSK problem by linearly scalarizing it for a given hyper-parameter $\lambda \in \Delta^4$, resulting in a problem that looks similar to the one in (0.1). These scalarized problems are then shown to be convex for a large class of λ's. By grid sampling Δ^4, and solving the associated scalarized problem (for different λ's) using first-order methods, we partially recover the

Pareto front of the MVSK problem. Chapter 10 considers the results of numerical experiments on real-world data and visualizes them.

In Part 4, we introduce two hypergraph-based classes of polynomials, which we denote (but do not define here) as f_d and p_d, respectively. We give a brief motivation from queueing theory for our interest in these polynomials and introduce some necessary preliminaries on the Terwilliger algebra of the binary Hamming cube (Chapter 11). In Chapter 12, we give the main result of Part 4; namely, we show that the polynomials p_d are convex over the standard simplex and that this implies that they attain their global minimum at the barycenter of the simplex. We do this by exploiting symmetry properties of p_d (also present in f_d) and showing that its Hessian is PSD using several PSDness preserving reductions and a classical result by Schrijver. In Chapter 13, we relate the polynomials p_d to the polynomials f_d and show some partial results for f_d in the same spirit as that of Chapter 12.

Publications

This thesis is based on the following research papers:

[**27**] D. Brosch, M. Laurent, and A. Steenkamp.
Optimizing hypergraph-based polynomials modeling job-occupancy in queueing with redundancy scheduling.
SIAM Journal on Optimization, 31(3):2227–2254, 2021.
https://doi.org/10.1137/20M1369592

[**81**] S. Gribling, M. Laurent, and A. Steenkamp.
Bounding the separable rank via polynomial optimization.
Linear Algebra and its Applications, 648:1–55, 2022.
https://doi.org/10.1016/j.laa.2022.04.010

[**100**] M. Korda, M. Laurent, V. Magron, and A. Steenkamp.
Exploiting ideal-sparsity in the generalized moment problem with application to matrix factorization ranks.
Mathematical Programming, 2023.
https://doi.org/10.1007/s10107-023-01993-x

[**151**] A. Steenkamp.
Matrix factorization ranks via polynomial optimization.
Chapter of *Polynomial Optimisation, Moments, and Applications,* to appear in Springer series “Optimization and Its Applications”, 2023.
http://arxiv.org/abs/2302.09994

[**150**] A. Steenkamp.
Convex scalarizations of the mean-variance-skewness-kurtosis problem in portfolio selection.
Submitted to INFORMS Journal on Computing, 2023.
https://arxiv.org/abs/2302.10573

Part 1

Polynomial optimization techniques

Part 1 of the thesis contains three chapters. Chapter 1 introduces most of the notation, basic definitions, and nomenclature for chapters 2, 3, and the rest of the thesis. Chapter 2 introduces the generalized moment problem (GMP), which describes a rich class of optimization problems. Our primary interest and motivation are to solve GMPs. To this end, we dedicate the third and final chapter, Chapter 3, where we describe the moment method. This is a classical method and our primary technique for approximating GMPs with a sequence of semidefinite programs.

We present our new research after stating classical results for contextualization. In particular, our results on ideal sparsity are presented following the formal definition of a GMP in Chapter 2. Similarly, the ideal-sparse hierarchy is only introduced once the classical moment method is described and the resulting hierarchy is defined in Chapter 3. Our most significant motivation for using ideal sparsity comes from its efficacy in bounding matrix factorization ranks (MFR). However, we only properly introduce MFR in Part 2, where we dedicate several chapters to the topic. We must stress that ideal sparsity has applications beyond MFR, and as such, it is treated here in Part 1 in the abstract setting of GMPs.

The contribution to ideal sparsity is based on our joint work with Milan Korda, Monique Laurent, and Victor Magron in [**100**].

CHAPTER 1

Polynomials

We introduce some fundamental objects, definitions, and results that will be used throughout the thesis. In particular, we introduce polynomials (Section 1.1), their dual, linear functionals (Section 1.2), and moment matrices (Section 1.3). We explain the fundamental relationships and properties between these three topics as we introduce them.

These first three sections are all stated for complex variables. This is a rather technical but necessary requirement for the forthcoming Chapter 7, where the objects of interest, separable states, are described as matrices with complex entries. Many classical results are phrased in the real setting; as such, we need to extend them to the complex setting. We do this in the final Section 1.4.

For the other topics of this thesis, we will work exclusively in the real setting. To this end, Section 1.4 contains many useful conversions between complex objects and their real analogs.

1.1. Basic definitions and objects

1.1.1. Basic notation. We start with some basic mathematical objects and notation. Let $\mathbb{Z}$ denote the set of integers, and let $\mathbb{N} := \{0, 1, 2, 3, ...\}$ denote the set of nonnegative integers. For any integer n, with $n \geq 1$, let $[n] := \{1, 2, ..., n\}$. Similarly, for any two distinct integers $k < n$ define the set $[k, n] := \{k, k+1, \dots, n-1, n\}$. For $n, k \in \mathbb{N}$ with $k \leq n$ denote by

$$\binom{n}{k} := \frac{n!}{(n-k)!k!}$$

the combinatorial parameter representing the number of ways one can choose k objects from a set of size n. Consider now a vector of nonnegative integers $\alpha \in \mathbb{N}^n$ and define its size by $|\alpha| := \sum_{i=1}^{n} \alpha_i$. The set of nonnegative integer vectors with size at most $k \in \mathbb{N}$ is denoted $\mathbb{N}^n_k := \{\alpha \in \mathbb{N}^n : |\alpha| \leq k\}$; it has cardinality $\binom{n+k}{k}$. For any real number $a \in \mathbb{R}$, we denote

- its *floor* by $\lfloor a \rfloor := \max\{b \in \mathbb{Z} : b \leq a\}$ and,
- its *ceiling* by $\lceil a \rceil := \min\{b \in \mathbb{Z} : b \geq a\}$.

For any set V we denote by $\mathcal{P}(V) := \{S : S \subseteq V\}$ the *power set* of V. We denote the *cardinality or size* of a finite set S by $|S| \in \mathbb{N}$.

Complex objects. For a complex scalar $a \in \mathbb{C}$, we denote its conjugate by $\overline{a}$ and its modulus by $|a| := \sqrt{\overline{a}a}$. For a complex vector $\mathbf{a} := (a_1, a_2, ..., a_n) \in \mathbb{C}^n$ we denote its conjugate transpose by $\mathbf{a}^* := (\overline{a}_1, ..., \overline{a}_n)^T$. Similarly, for a complex matrix $X \in \mathbb{C}^{n \times m}$, we denote its transpose by X^T and its conjugate transpose by X^*.

Vectors and matrices. Let the vector space $\mathbb{C}^n$ be equipped with the scalar product $\langle \mathbf{x}, \mathbf{y} \rangle := \mathbf{x}^* \mathbf{y} = \sum_{i=1}^n x_i^* y_j$ for $\mathbf{x}, \mathbf{y} \in \mathbb{C}^n$. Our convention is to use bold lower-case Roman letters for vectors, e.g., $\mathbf{a}, \mathbf{b}, \mathbf{x}, \mathbf{y}, \mathbf{z}, \mathbf{u}, \mathbf{v}, \mathbf{w}$. This inner product induces the Euclidean norm: $\|\mathbf{x}\| := \sqrt{\mathbf{x}^* \mathbf{x}}$. Analogously, the vector space $\mathbb{C}^{n \times n}$ is equipped with the trace inner product $\langle X, Y \rangle = \mathrm{Tr}(X^* Y) = \sum_{i,j=1}^n \overline{X}_{ij} Y_{ij}$ and the Frobenius norm $\|X\| := \sqrt{\langle X, X \rangle}$, where $X, Y \in \mathbb{C}^{n \times n}$. The *support of a vector* $\mathbf{x} \in \mathbb{R}^n$ is the set of indices

$$\mathrm{supp}(\mathbf{x}) := \{i \in [n] : x_i \neq 0\}.$$

For a set S in a vector space, we let $\mathrm{cone}(S)$ and $\mathrm{conv}(S)$ denote its conic hull and its convex hull.

We let I_n and J_n denote the *identity matrix* and the *all-ones matrix* of size n, which we sometimes also denote as I and J when the dimension is clear from the context.

A matrix $X \in \mathbb{C}^{n \times n}$ is called Hermitian if $X^* = X$, and we denote the space of complex Hermitian $n \times n$ matrices by $\mathcal{H}^n$. A matrix $X \in \mathcal{H}^n$ is said to be (Hermitian) positive semidefinite (PSD), denoted $X \succeq 0$, if $\mathbf{v}^* X \mathbf{v} \geq 0$ for all $\mathbf{v} \in \mathbb{C}^n$. Let $\mathcal{H}^n_+$ denote the cone of Hermitian positive semidefinite matrices; it is self-dual in the sense that,

$$X \in \mathcal{H}^n_+ \iff \langle X, Y \rangle \geq 0 \text{ for all } Y \in \mathcal{H}^n_+.$$

Furthermore, every Hermitian PSD matrix $X \in \mathcal{H}^n_+$ has a *Cholesky factorization*, i.e., $X = VV^*$ for some $V \in \mathbb{C}^{n \times r}$, where $r := \mathrm{rank}(X)$.

Semidefinite program. For Hermitian matrices $C, A_1, ..., A_m \in \mathcal{H}^n$, and real numbers $b_1, ..., b_m \in \mathbb{R}$ we call the following optimization problem a *semidefinite program* (SDP):

$$\begin{aligned} &\inf_{X \in \mathcal{H}^n_+} \langle C, X \rangle \\ &\text{s.t. } \langle A_i, X \rangle = b_i \ (i \in [m]). \end{aligned}$$

SDPs like the above can be solved (up to some ε-accuracy) efficiently by interior point methods (under some technical assumptions like rational data, knowledge of a feasible point, and well-behavedness of the feasible region); see, e.g., [**125, 45**]. As such, a running theme throughout this thesis will be reformulating various optimization problems as SDPs, and then solving them.

Restriction notation. Consider a set $U \subseteq V := [n]$. Given a vector $\mathbf{y} \in \mathbb{R}^{|U|}$, we let $(\mathbf{y}, 0_{V \setminus U}) \in \mathbb{R}^n$ denote the vector obtained by padding $\mathbf{y}$ with zeros at the entries indexed by $[n] \setminus U$. For an n-variate function $f : \mathbb{R}^{|V|} \to \mathbb{R}$, we let $f_{|U} : \mathbb{R}^{|U|} \to \mathbb{R}$ denote the function in the variables $\mathbf{x}(U) := \{x_i : i \in U\}$, which is obtained from f by setting to zero all the variables x_i indexed by $i \in V \setminus U$. That is, $f_{|U}(\mathbf{y}) = f(\mathbf{y}, 0_{V \setminus U})$ for $\mathbf{y} \in \mathbb{R}^{|U|}$. So, if f is an n-variate polynomial, then $f_{|U}$ is a $|U|$-variate polynomial in the variables $\mathbf{x}(U)$.

Basic graph theory. Often we use $G := (V, E)$ to denote a *graph* with set of vertices V and set of edges E. The set of *non-edges* of G is defined to be the set

$$\overline{E} = \Big\{\{i,j\} : i \in V,\ j \in V,\ i \neq j,\ \{i,j\} \notin E\Big\},$$

and similarly, the complement of G is defined as $\overline{G} := (V, \overline{E})$. For any subset of vertices $S \subseteq V$ we denote the set of *induced edges* as

$$E(S) := \Big\{\{i,j\} \in E : \{i,j\} \subseteq S\Big\}.$$

A *clique* $C \subseteq V$ of G is a set of vertices such that $\{i,j\} \in E$ for all $i, j \in C$ with $i \neq j$. An *edge covering* $\mathcal{E} \subseteq \mathcal{P}(V)$ of G is a collection of vertex sets with the property that every edge $e \in E$ is contained in some $S \in \mathcal{E}$.

For a symmetric matrix $A \in \mathbb{R}^{n \times n}$ we define its *support graph*

$$G_A := \Big([n], E_A := \big\{\{i,j\} \subseteq [n] : i \neq j,\ A_{i,j} \neq 0\big\}\Big).$$

For an example of the above, we refer to Fig. 1 in Section 2.2.

A sequence of distinct vertices $v_1, v_2, ..., v_k \in V$ with $k \geq 3$ is called a *cycle C (of length k)* of G if $\{v_1, v_2\}, \{v_2, v_3\}, ..., \{v_k, v_1\} \in E$. Then, an edge $\{v_i, v_j\} \in E$ with $|i - j| \geq 2$ is called a *chord* of C. A graph G is said to be chordal if any cycle C of length 4 or more has a chord.

1.1.2. Polynomials. We consider polynomials in n complex variables $x_1, ..., x_n$ and their conjugates $\overline{x}_1, ..., \overline{x}_n$. For $\alpha, \beta \in \mathbb{N}^n$ we use the short-hand $\mathbf{x}^\alpha \overline{\mathbf{x}}^\beta$ to denote the monomial

$$\mathbf{x}^\alpha \overline{\mathbf{x}}^\beta := \prod_{i \in [n]} x_i^{\alpha_i} \prod_{j \in [n]} \overline{x_j}^{\beta_j}.$$

The degree of this monomial is defined to be the following nonnegative integer:

$$\deg(\mathbf{x}^\alpha \overline{\mathbf{x}}^\beta) := |\alpha| + |\beta| = \sum_{i \in [n]} \alpha_i + \beta_i \in \mathbb{N}.$$

It is often convenient to refer to the maximal degree in a set $H \subseteq \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$ of polynomials; in such cases, we write $d_H := \max_{g \in H}\{\deg(g)\}$. We collect the set of all monomials of degree at most $t \in \mathbb{N} \cup \{\infty\}$ in the vector $[\mathbf{x}, \overline{\mathbf{x}}]_t$ (using some given ordering of the monomials); for ease of notation, we set $[\mathbf{x}, \overline{\mathbf{x}}] := [\mathbf{x}, \overline{\mathbf{x}}]_\infty$ consisting of all monomials (i.e., with no degree upper bound).

Note that $\overline{[\mathbf{x},\overline{\mathbf{x}}]}=[\overline{\mathbf{x}},\mathbf{x}]$. It is often convenient to use $[\mathbf{x},\overline{\mathbf{x}}]_t$ as a set, ignoring its ordering. In such cases, we write $\mathbf{x}^\alpha\overline{\mathbf{x}}^\beta \in [\mathbf{x},\overline{\mathbf{x}}]_t$. Taking the complex linear span of all monomials in $[\mathbf{x},\overline{\mathbf{x}}]_t$ gives

$$\mathbb{C}[\mathbf{x},\overline{\mathbf{x}}]_t := \operatorname{span}\{m : m \in [\mathbf{x},\overline{\mathbf{x}}]_t\} = \Big\{ \sum_{m\in[\mathbf{x},\overline{\mathbf{x}}]_t} a_m m : a_m \in \mathbb{C}\Big\},$$

the space of polynomials with complex coefficients and degrees at most t. If $t=\infty$ we write $\mathbb{C}[\mathbf{x},\overline{\mathbf{x}}]$ for the full polynomial ring in $\mathbf{x},\overline{\mathbf{x}}$ over $\mathbb{C}$. Observe that any polynomial $p \in \mathbb{C}[\mathbf{x},\overline{\mathbf{x}}]$ is of the form

$$p = \sum_{\alpha,\beta} p_{\alpha,\beta}\mathbf{x}^\alpha\overline{\mathbf{x}}^\beta,$$

where only finitely many coefficients $p_{\alpha,\beta}$ are nonzero. The *degree* of p is the maximum degree of its constituent monomials, i.e.,

$$\deg(p) := \max_{p_{\alpha,\beta}\neq 0} \deg(\mathbf{x}^\alpha\overline{\mathbf{x}}^\beta).$$

Let $\mathbb{C}_0^{\mathbb{N}^n\times\mathbb{N}^n} := \{\mathbf{a} = (a_{\alpha,\beta})_{(\alpha,\beta)\in\mathbb{N}^n\times\mathbb{N}^n} : \|\mathbf{a}\|_0 < \infty\}$ denote the set of vectors in $\mathbb{C}^{\mathbb{N}^n\times\mathbb{N}^n}$ that have only finitely many nonzero entries. Then any polynomial p can be written as

$$p = \mathbf{a}^*[\mathbf{x},\overline{\mathbf{x}}] \text{ for some unique } \mathbf{a} = \Big(\overline{p}_{\alpha,\beta}\Big)_{(\alpha,\beta)\in N^n\times\mathbb{N}^n} \in \mathbb{C}_0^{\mathbb{N}^n\times\mathbb{N}^n}. \tag{1.1}$$

Conjugation on complex variables extends linearly to polynomials, for $p = \sum_{\alpha,\beta} p_{\alpha,\beta}\mathbf{x}^\alpha\overline{\mathbf{x}}^\beta$ we define its conjugate polynomial $\overline{p} := \sum_{\alpha,\beta}\overline{p}_{\alpha,\beta}\overline{\mathbf{x}}^\alpha\mathbf{x}^\beta$.

Polynomials equal to their conjugate, i.e., polynomials p such as $p=\overline{p}$, are called *Hermitian*. Hermitian polynomials are noteworthy partly because they take only real values, i.e., $p(\mathbf{x},\overline{\mathbf{x}}) \in \mathbb{R}$ for all $\mathbf{x}\in\mathbb{C}^n$. We denote the space of Hermitian polynomials by $\mathbb{C}[\mathbf{x},\overline{\mathbf{x}}]^h$. As two examples of Hermitian polynomials, we present the following univariate polynomials: $p = x+\overline{x}$ and $q = \mathbf{i}x - \mathbf{i}\overline{x}$. As an example of a non-Hermitian polynomial, we present $r = x-\overline{x}$; indeed $\overline{r}=-r$ and $r(\mathbf{i}) = 2\mathbf{i} \notin \mathbb{R}$. Polynomials of the form $q\overline{q}$ (for some $q \in \mathbb{C}[\mathbf{x},\overline{\mathbf{x}}]$) are called Hermitian squares and are a particular class of Hermitian polynomials that take only nonnegative values. For any $t\in\mathbb{N}\cup\{\infty\}$ we define the cone of *sums of Hermitian squares* (Hermitian SoS) with degree at most $2t$ by

$$\Sigma[\mathbf{x},\overline{\mathbf{x}}]_{2t} := \Big\{\sum_{i\in[k]} q_i\overline{q}_i : k\in\mathbb{N},\ q_1,q_2,...,q_k \in \mathbb{C}[\mathbf{x},\overline{\mathbf{x}}]_t\Big\}.$$

Since each element p of $\Sigma[\mathbf{x},\overline{\mathbf{x}}]_{2t}$ is a conic combination of Hermitian squares, we have that p takes only nonnegative values. If the variables are clear from the context, we write Σ_{2t}, and if additionally $t=\infty$, we write Σ.

SoS-polynomial matrices. A (complex) polynomial matrix $S \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^{m\times m}$ is called an *SoS-polynomial matrix* if $S = UU^*$ for some polynomial matrix $U \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^{m\times k}$ and some integer $k \in \mathbb{N}$, or, equivalently, if

$$S \in \operatorname{cone}\{\vec{\mathbf{p}}\vec{\mathbf{p}}^* : \vec{\mathbf{p}} = (p_1, \dots, p_m) \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^m\}.$$

Clearly, an SoS-polynomial matrix S takes only Hermitian PSD values, i.e., $S(\mathbf{x}, \overline{\mathbf{x}}) \in \mathcal{H}^m_+$ for all $\mathbf{x} \in \mathbb{C}^n$. We have added the arrow notation "$\vec{\mathbf{p}}$" to distinguish vectors with polynomial entries from those with scalar entries. At this point, there may be some confusion on the use of "Hermitian" with regards to a polynomial matrix $S \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^{m\times m}$. Observe that S can be Hermitian as a matrix, i.e., $S = S^*$, which means that $S_{i,j} = \overline{S_{j,i}}$ for all $i, j \in [m]$. However, S can also have an entry $S_{i,j}$ that is Hermitian as a polynomial, i.e., $S_{i,j} = \overline{S_{i,j}}$. Whenever we say a polynomial matrix is Hermitian, it should be understood in the matrix sense unless we specifically say otherwise.

Semi-algebraic sets. It will be useful to work with a special class of sets in $\mathbb{C}^n$ called *semi-algebraic sets.* A set $K \subseteq \mathbb{C}^n$ is called (basic closed) semi-algebraic if there exist integers $N_g, N_h \in \mathbb{N}$, Hermitian polynomials $g_1, ..., g_{N_g} \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h$, and polynomials $h_1, ..., h_{N_h} \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$ such that

$$K = \Big\{\mathbf{x} \in \mathbb{C}^n : \; g_i(\mathbf{x}, \overline{\mathbf{x}}) \geq 0 \ (i \in [N_g]), \; h_j(\mathbf{x}, \overline{\mathbf{x}}) = 0 \ (j \in [N_h])\Big\}. \tag{1.2}$$

We have added the equality constraints for exposition purposes. The inequality constraints alone are sufficient because $h = 0$ is equivalent to $h \geq 0$ and $-h \geq 0$.

1.2. Dual space of polynomials

The algebraic dual space of $\mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$ is the vector space

$$\mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^* := \Big\{L : \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}] \ni p \to L(p) \in \mathbb{C} \mid L \text{ is linear}\Big\}$$

of all linear functionals L mapping polynomials in $\mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$ to $\mathbb{C}$. Just as we sometimes work with polynomials of bounded degree, i.e., $\mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]_t$ for some $t \in \mathbb{N} \cup \{\infty\}$, we will work with the space $\mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*_t$ of linear functionals defined on $\mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]_t$. If $t = \infty$, we omit the subscript and write $\mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$.

Hermitian linear functionals. A linear functional $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ is called *Hermitian* if $L(\overline{p}) = \overline{L(p)}$ for all $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$. A (Hermitian) linear functional $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ is called *positive*, written as $L \geq 0$, if it maps Hermitian squares to nonnegative real numbers, i.e., if $L(q\overline{q}) \geq 0$ for all $q \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$.

Evaluation functionals. As an example of a linear functional, we can define, for any $\mathbf{a} \in \mathbb{C}^n$, the *evaluation functional* $L_{\mathbf{a}} \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ *at* $\mathbf{a}$ as follows:

$$L_{\mathbf{a}} : \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}] \ni p \to p(\mathbf{a}) \in \mathbb{C}.$$

It is easy to see that $L_{\mathbf{a}}$ is both Hermitian and positive.

Polynomial localizing maps gL. Given a polynomial $g \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$ and a linear functional $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ we can construct a new linear functional $gL \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ defined entry-wise as follows:

$$gL : \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}] \ni p \mapsto L(gp) \in \mathbb{C}. \tag{1.3}$$

Via the above construction, we say that g acts on $\mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ by mapping L to gL. If $g \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h$ and L is Hermitian, then gL is Hermitian.

Matrix-valued linear functionals. The notion of matrix-valued linear functionals is similar to the above scalar-valued linear functionals. These functionals act on polynomials $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$ but take values in some matrix space $\mathbb{C}^{m \times m}$, where $m \in \mathbb{N}$ is arbitrary but fixed. Consider the following matrix-valued linear functional:

$$\mathcal{L} : \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}] \ni p \mapsto \mathcal{L}(p) := \big(L_{ij}(p)\big)_{i,j \in [m]} \in \mathbb{C}^{m \times m}. \tag{1.4}$$

Here, $\mathcal{L} = (L_{ij})_{i,j=1}^m$ with each $L_{ij} \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ being a scalar-valued linear functional. We write $(\mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*)^{m \times m}$ for the set of all $m \times m$-matrix-valued linear functionals $\mathcal{L}$ acting on polynomials in $\mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$. We call $\mathcal{L}$ *Hermitian* if $\mathcal{L}(\overline{p}) = \mathcal{L}(p)^*$, i.e., $L_{ij}(\overline{p}) = \overline{L_{ji}(p)}$ for all $i, j \in [m]$ and all $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$. Furthermore, $\mathcal{L}$ is said to be *positive*, written $\mathcal{L} \succeq 0$, if it maps positive elements (i.e., Hermitian squares $p\overline{p}$) to positive elements (i.e., Hermitian positive semidefinite $m \times m$ matrices), i.e., if the following holds:

$$\mathcal{L}(p\overline{p}) = (L_{ij}(p\overline{p}))_{i,j=1}^m \succeq 0 \text{ for all } p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]. \tag{1.5}$$

We define an action of $\mathcal{L}$ on a polynomial matrix $S = (S_{ij})_{i,j=1}^m \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^{m \times m}$:

$$\langle \mathcal{L}, S \rangle := \sum_{i,j=1}^m L_{ij}(S_{ij}). \tag{1.6}$$

If $\mathcal{L}$ and S are both Hermitian, then $\langle \mathcal{L}, S \rangle \in \mathbb{R}$. Furthermore, if $\mathcal{L}$ is positive and $S \succeq 0$, then $\langle \mathcal{L}, S \rangle \geq 0$.

Polynomial matrix localizing maps $g\mathcal{L}$. Similar to what we did with scalar-valued linear maps in (1.3), we can combine a matrix-valued linear functional $\mathcal{L}$ and a polynomial $g \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$ to create a new matrix-valued linear functional

$$g\mathcal{L} : \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}] \ni p \mapsto (g\mathcal{L})(p) = \mathcal{L}(gp) = \big(L_{ij}(gp)\big)_{i,j \in [m]} \in \mathbb{C}^{m \times m}. \tag{1.7}$$

Suppose g and $\mathcal{L}$ are Hermitian (in their respective senses), then $g\mathcal{L}$ is Hermitian. Furthermore, if $g \in \Sigma$ is a sum of squares polynomial and $\mathcal{L}$ is positive, then $g\mathcal{L}$ is positive.

Polynomial matrix localizing maps $G \otimes L$. We can extend the construction of gL in (1.3) to the case when g is no longer a polynomial but rather a polynomial matrix $G = (G_{ij})_{i,j=1}^m \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^{m\times m}$ (we assume for convenience that G is square). Given a linear functional $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$, we can construct a new matrix-valued linear functional $G \otimes L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ defined entry-wise as follows:

$$G \otimes L : \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}] \ni p \mapsto (G \otimes L)(p) := \big((G_{ij}L)(p)\big)_{i,j=1}^m = \big(L(G_{ij}p)\big)_{i,j=1}^m \in \mathbb{C}^{m\times m}.$$

For our applications in Part 2, we will only consider matrix-valued linear functionals of the form $G \otimes L$. However, for ease of proof and clarity of exposition, we will continue work in the general setting whenever possible. It should be noted that the general setting was used in [**80**] to establish the link between the moment method and the DPS hierarchy (based on state extension) for approximating the set of separable states.

1.3. Moment matrices

Functionals on polynomials extend entry-wise to polynomial matrices. Formally, given a linear functional $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ and a polynomial matrix $G = (G_{ij})_{i,j=1}^m \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^{m\times m}$, we define $L(G)$ to be the matrix

$$L(G) := \Big(L(G_{ij})\Big)_{i,j\in[m]} \in \mathbb{C}^{m\times m}.$$

We apply this operation to define the notion of moment matrices.

Moment matrices of scalar-valued linear functionals. Fix $t \in \mathbb{N} \cup \{\infty\}$ and consider the result of applying a linear functional $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]_{2t}$ to the (possibly infinite) polynomial matrix $[\mathbf{x}, \overline{\mathbf{x}}]_t[\mathbf{x}, \overline{\mathbf{x}}]_t^*$ to get

$$M_t(L) := L([\mathbf{x}, \overline{\mathbf{x}}]_t[\mathbf{x}, \overline{\mathbf{x}}]_t^*) = \Big(L(\mathbf{x}^{\alpha+\gamma}\overline{\mathbf{x}}^{\beta+\delta})\Big)_{(\alpha,\beta),(\gamma,\delta)\in(\mathbb{N}^n)^2:\ |\alpha+\beta|,\ |\gamma+\delta|\leq t}. \tag{1.8}$$

The matrix $M_t(L)$ is called the *moment matrix of L of order t*. It satisfies what is called the *moment property*, which means that, for any (α, β), (γ, δ), (α', β'), $(\gamma', \delta') \in (\mathbb{N}^n)^2$ with $|\alpha + \beta|$, $|\gamma + \delta|$, $|\alpha' + \beta'|$, $|\gamma' + \delta'| \leq t$ and $\alpha + \gamma = \alpha' + \gamma'$, $\beta + \delta = \beta' + \delta'$, we have

$$(M_t(L))_{(\alpha,\beta),(\gamma,\delta)} = (M_t(L))_{(\alpha',\beta'),(\gamma',\delta')}.$$

When we make no mention of the order t, we imply that $t = \infty$. It is useful to note that the moment matrix operation is linear, i.e.,

$$M_t(aL_1 + bL_2) = aM_t(L_1) + bM_t(L_2) \tag{1.9}$$

for any two linear functionals $L_1, L_2 \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ and scalars $a, b \in \mathbb{C}$.

Observe that the moment matrix of an evaluation functional $L_{\mathbf{a}}$ at $\mathbf{a} \in \mathbb{C}^n$ is

$$M_t(L_{\mathbf{a}}) = [\mathbf{a}, \overline{\mathbf{a}}]_t [\mathbf{a}, \overline{\mathbf{a}}]_t^*,$$

which is clearly a rank-one matrix. Hence, if L is a linear combination of evaluation functionals, its moment matrix $M_t(L)$ has a finite rank, also if $t = \infty$.

Lemma 1.1. *A linear functional $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ is Hermitian if and only if its moment matrix $M(L)$ is Hermitian.*

Proof. If $L(\overline{p}) = \overline{L(p)}$ for all $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$, then

$$\begin{aligned} M(L)^* &= \overline{L([\mathbf{x}, \overline{\mathbf{x}}][\mathbf{x}, \overline{\mathbf{x}}]^*)}^T = L(\overline{[\mathbf{x}, \overline{\mathbf{x}}][\mathbf{x}, \overline{\mathbf{x}}]^*})^T \\ &= L([\overline{\mathbf{x}}, \mathbf{x}][\overline{\mathbf{x}}, \mathbf{x}]^*)^T = L([\overline{\mathbf{x}}, \mathbf{x}][\mathbf{x}, \overline{\mathbf{x}}]^T)^T = L([\mathbf{x}, \overline{\mathbf{x}}][\mathbf{x}, \overline{\mathbf{x}}]^*) = M(L). \end{aligned}$$

Conversely, if $M(L) = M(L)^*$, then $\overline{L(\mathbf{x}^\alpha \overline{\mathbf{x}}^\beta)} = L(\overline{\mathbf{x}}^\alpha \mathbf{x}^\beta) = L(\overline{\mathbf{x}^\alpha \overline{\mathbf{x}}^\beta})$ for every $\alpha, \beta \in \mathbb{N}^n$. In particular, for any $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$, we have

$$L(\overline{p}) = \sum_{\alpha,\beta} \overline{p_{\alpha,\beta}} L(\overline{\mathbf{x}^\alpha \overline{\mathbf{x}}^\beta}) = \sum_{\alpha,\beta} \overline{p_{\alpha,\beta}} \overline{L(\mathbf{x}^\alpha \overline{\mathbf{x}}^\beta)} = \overline{L(p)}. \qquad \square$$

The above lemma holds mutatis mutandis when applied to a truncated functional $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]_t^*$ and its associated moment matrix $M_t(L)$.

Lemma 1.2. *A linear functional $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$ is positive if and only if its moment matrix $M(L)$ is PSD.*

Proof. The claim becomes clear once one considers the following fact: for any polynomial $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$, written as $p = \mathbf{a}^*[\mathbf{x}, \overline{\mathbf{x}}]$ with $\mathbf{a} \in \mathbb{C}_0^{\mathbb{N}^n \times \mathbb{N}^n}$, we have

$$L(p\overline{p}) = L(\mathbf{a}^*[\mathbf{x}, \overline{\mathbf{x}}][\mathbf{x}, \overline{\mathbf{x}}]^* \mathbf{a}) = \mathbf{a}^* L([\mathbf{x}, \overline{\mathbf{x}}][\mathbf{x}, \overline{\mathbf{x}}]^*)\mathbf{a} = \mathbf{a}^* M(L)\mathbf{a}. \tag{1.10}$$

Hence,

$$L \geq 0 \iff M(L) \succeq 0.$$

$\square$

It may be useful to generalize the above fact in (1.10): if $p = \mathbf{a}^*[\mathbf{x}, \overline{\mathbf{x}}]$ and $q = \mathbf{b}^*[\mathbf{x}, \overline{\mathbf{x}}]$ with $\mathbf{a}, \mathbf{b} \in \mathbb{C}_0^{\mathbb{N}^n \times \mathbb{N}^n}$, then $L(p\overline{q}) = \mathbf{a}^* M(L)\mathbf{b}$.

Moment matrices of polynomial localizing maps gL. As stated in Lemma 1.2, gL is positive if and only if its moment matrix is PSD, i.e.,

$$gL \geq 0 \iff L(g \cdot [\mathbf{x}, \overline{\mathbf{x}}][\mathbf{x}, \overline{\mathbf{x}}]^*) = M(gL) \succeq 0. \tag{1.11}$$

If g is Hermitian (as a polynomial) and L is Hermitian (as a functional), then gL is a Hermitian functional, and hence $M(gL)$ is a Hermitian matrix. For the special case where L is an evaluation functional $L_{\mathbf{a}}$ and g is a polynomial that is nonnegative at $\mathbf{a}$, i.e., $g(\mathbf{a}, \overline{\mathbf{a}}) \geq 0$, we have that $gL \geq 0$. This is clear when considering any $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$ and observing that

$$(gL)(p\overline{p}) = g(\mathbf{a}, \overline{\mathbf{a}})|p(\mathbf{a}, \overline{\mathbf{a}})|^2 \geq 0.$$

In the literature (see [**106**]), $M(gL)$ is often called a *localizing moment matrix.* In a similar vein to (1.11) we have for any $h \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$ the following:

$$hL = 0 \iff L(h \cdot [\mathbf{x}, \overline{\mathbf{x}}][\mathbf{x}, \overline{\mathbf{x}}]^*) = M(hL) = 0. \tag{1.12}$$

Equality here is entry-wise but expressed in matrix form for ease of notation. The implications of (1.11) and (1.12) easily transfer to the truncated setting. Given a (truncated) linear functional $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*_{2t}$, its (truncated) localizing matrix is defined as

$$M_{t-d_g}(gL) = L(g \cdot [\mathbf{x}, \overline{\mathbf{x}}]_{t-d_g}[\mathbf{x}, \overline{\mathbf{x}}]^*_{t-d_g}),$$

where we subtract $d_g := \lceil \frac{\deg(g)}{2} \rceil$ from the degree bound to account for the effect of multiplying the monomials with g. Similarly, a localizing equality constraint $h = 0$ will be encoded by

$$L(h \cdot [\mathbf{x}, \overline{\mathbf{x}}]_{2t-\deg(h)}) = 0.$$

Moment matrices of matrix-valued linear functionals $\mathcal{L}$. Analogous to the scalar-valued linear functional setting, we can, for a matrix-valued linear functional $\mathcal{L} = \left(L_{ij}\right)_{i,j=1}^m \in (\mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*)^{m \times m}$, define a moment matrix block-wise as follows:

$$M(\mathcal{L}) := \mathcal{L}([\mathbf{x}, \overline{\mathbf{x}}][\mathbf{x}, \overline{\mathbf{x}}]^*) = \Big(L_{ij}([\mathbf{x}, \overline{\mathbf{x}}][\mathbf{x}, \overline{\mathbf{x}}]^*)\Big)_{i,j=1}^m = (M(L_{ij}))_{i,j=1}^m. \tag{1.13}$$

Here, we view $M(\mathcal{L})$ as an $m \times m$ block-matrix whose $(i, j)^{th}$ block is the moment matrix $M(L_{ij})$ of the scalar-valued linear functional $L_{ij} \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$. It should be clear that $M(\mathcal{L})$ is a Hermitian matrix if $\mathcal{L}$ is Hermitian.

Positivity of $\mathcal{L}$ and its moment matrix $M(\mathcal{L})$**.** Similar to the scalar-valued case, if $M(\mathcal{L}) \succeq 0$, then $\mathcal{L}$ is positive, i.e., $\mathcal{L}(p\overline{p}) \succeq 0$ for all $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$. However, the reverse implication may not generally hold; we motivate why in the following two lemmas.

LEMMA 1.3. *$\mathcal{L}$ is positive, i.e., (1.5) holds, if and only if any of the following equivalent conditions holds:*

$$M(\mathbf{v}^*\mathcal{L}\mathbf{v}) \succeq 0 \quad \textit{for all } \mathbf{v} \in \mathbb{C}^m, \tag{1.14}$$

$$(\mathbf{v} \otimes \mathbf{a})^* \, M(\mathcal{L}) \, (\mathbf{v} \otimes \mathbf{a}) \geq 0 \textit{ for all } \mathbf{v} \in \mathbb{C}^m \textit{ and } \mathbf{a} \in \mathbb{C}_0^{\mathbb{N}^n \times \mathbb{N}^n}, \tag{1.15}$$

$$\mathbf{v}^*\mathcal{L}(p\overline{p})\mathbf{v} = (\mathbf{v}^*\mathcal{L}\mathbf{v})(p\overline{p}) \geq 0 \textit{ for all } \mathbf{v} \in \mathbb{C}^m \textit{ and } p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]. \tag{1.16}$$

PROOF. The equivalence between (1.5) and (1.16) follows from the definition of PSDness. The equivalence between (1.16) and (1.14) follows from using (1.10) applied to the (scalar-valued) map $\mathbf{v}^*\mathcal{L}\mathbf{v}$ for each $\mathbf{v} \in \mathbb{C}^m$. To see the equivalence of (1.16) and (1.15), take an arbitrary polynomial $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$ and write it as $p = \mathbf{a}^*[\mathbf{x}, \overline{\mathbf{x}}]$ with $\mathbf{a} = (a_{\alpha,\beta}) \in \mathbb{C}_0^{\mathbb{N}^n \times \mathbb{N}^n}$. Use (1.10) and the definition of $M(\mathcal{L})$ from (1.13) to get the following result for any $\mathbf{v} \in \mathbb{C}^m$:

$$\mathbf{v}^*\mathcal{L}(p\overline{p})\mathbf{v} = \mathbf{v}^*(L_{ij}(p\overline{p}))_{i,j=1}^m\mathbf{v} = \mathbf{v}^*(\mathbf{a}^*M(L_{ij})\mathbf{a})_{i,j=1}^m\mathbf{v} = (\mathbf{v}\otimes\mathbf{a})^*M(\mathcal{L})(\mathbf{v}\otimes\mathbf{a}).$$

□

Lemma 1.4. *$M(\mathcal{L}) \succeq 0$ if and only if any one of the following equivalent conditions holds:*

$$\mathbf{w}^*M(\mathcal{L})\mathbf{w} \geq 0 \textit{ for all } \mathbf{w} \in \mathbb{C}^m \otimes \mathbb{C}_0^{\mathbb{N}^n \times \mathbb{N}^n}, \tag{1.17}$$

$$\langle \mathcal{L}, S\rangle \geq 0 \textit{ for all SoS-polynomial matrices } S \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^{m\times m}, \tag{1.18}$$

$$\langle \mathcal{L}, \vec{\mathbf{p}}\vec{\mathbf{p}}^*\rangle = \sum_{i,j=1}^m L_{ij}(p_i\overline{p}_j) \geq 0 \textit{ for all } \vec{\mathbf{p}} = (p_1, ..., p_m) \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^m. \tag{1.19}$$

Proof. Condition (1.17) is just the definition of PSDness. The equivalence between (1.18) and (1.19) follows from the fact that any SoS-polynomial matrix S is a conic combination of rank-one SoS-polynomial matrices, i.e.,

$$S = \sum_{i=1}^k \vec{\mathbf{p}}_i\vec{\mathbf{p}}_i^*,$$

for some $k \in \mathbb{N}$ and $\vec{\mathbf{p}}_1, ..., \vec{\mathbf{p}}_k \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^m$.
To show that (1.19) and (1.17) are equivalent, start by considering an arbitrary vector $\mathbf{w} = (w_{i,(\alpha,\beta)})_{i,(\alpha,\beta)}$ in $\mathbb{C}^m \otimes \mathbb{C}_0^{\mathbb{N}^n \times \mathbb{N}^n}$. For each $i \in [m]$ define the vector $\mathbf{a}_i = (w_{i,(\alpha,\beta)})_{(\alpha,\beta)} \in \mathbb{C}_0^{\mathbb{N}^n \times \mathbb{N}^n}$ and its corresponding polynomial $p_i = \mathbf{a}_i^*[\mathbf{x}, \overline{\mathbf{x}}]$. From these m polynomials define the polynomial vector $\vec{\mathbf{p}} = (p_1, ..., p_m) \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^m$ of length m. Then

$$\mathbf{w}^*M(\mathcal{L})\mathbf{w} = \mathbf{w}^*(M(L_{ij}))_{i,j=1}^m\mathbf{w} = \sum_{i,j=1}^m (\mathbf{a}_i^*L_{ij}([\mathbf{x}, \overline{\mathbf{x}}][\mathbf{x}, \overline{\mathbf{x}}]^*)\mathbf{a}_j)_{i,j=1}^m$$

$$= \sum_{i,j=1}^m (L_{ij}(\mathbf{a}_i^*[\mathbf{x}, \overline{\mathbf{x}}][\mathbf{x}, \overline{\mathbf{x}}]^*\mathbf{a}_j))_{i,j=1}^m = \sum_{i,j=1}^m L_{ij}(p_i\overline{p}_j) = \langle \mathcal{L}, \vec{\mathbf{p}}\vec{\mathbf{p}}^*\rangle. \qquad \square$$

We now observe the similarities and disparities between the characterizations of a positive functional $\mathcal{L}$ and the positivity characterizations of its moment matrix $M(\mathcal{L})$. First, observe that (1.15) is a special case of (1.17), where the vectors $\mathbf{w}$ now must have a tensor product form $\mathbf{w} = \mathbf{v} \otimes \mathbf{a}$. Second, (1.16) is a restriction of (1.19) to the case where $\vec{\mathbf{p}} = (v_1p, v_2p, ..., v_mp)$ for some $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$. Hence, we have the following result.

Lemma 1.5. *If $M(\mathcal{L}) \succeq 0$ then $\mathcal{L}$ is positive.*

Note how (1.18) established the duality relationship between $m \times m$ SoS-polynomial matrices and $m \times m$-matrix valued linear maps $\mathcal{L}$ with $M(\mathcal{L}) \succeq 0$.

In the particular case that $\mathcal{L} = G(\mathbf{x}, \overline{\mathbf{x}}) \otimes L$ for some SoS-polynomial matrix $G(\mathbf{x}, \overline{\mathbf{x}})$ and L is a sum of evaluation functionals, we have that $M(\mathcal{L}) \succeq 0$.

On the tractability of showing $M_t(\mathcal{L}) \succeq 0$ ***vs. showing*** $\mathcal{L} \succeq 0$ ***on*** Σ_{2t}***.*** For a fixed $t \in \mathbb{N}$ and a fixed choice of $\mathcal{L} \in (\mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*)^{m \times m}$, it is computationally tractable to check if $M_t(\mathcal{L}) \succeq 0$ since this amounts to checking whether a single $(\binom{n+t}{t} \cdot m)$-sized matrix is PSD. In contradistinction, it is not clear how to check if $\mathcal{L}$ is positive on all sums of squares polynomials of degree at most $2t$, as this would require checking for every $\mathbf{v} \in \mathbb{C}^n$ if the $\binom{n+t}{t}$-sized matrix $M_t(\mathbf{v}^* \mathcal{L} \mathbf{v})$ is PSD. Hence, $M_t(\mathcal{L}) \succeq 0$ is both a stronger and easier-to-verify condition than $\mathcal{L} \succeq 0$ on Σ_{2t}.

Moment matrices of polynomial matrix localizing maps $G \otimes L$***.*** Polynomial matrix localizing maps $G \otimes L$ are a particular class of matrix-valued linear maps. As such, all of the above results transfer to this setting. We examine this particular specialization because the results will be used in Chapter 3 and in Part 2 to construct moment hierarchies especially suited for matrix factorization ranks. The moment matrix of $G \otimes L$ is

$$M(G \otimes L) = L(G \otimes [\mathbf{x}, \overline{\mathbf{x}}][\mathbf{x}, \overline{\mathbf{x}}]^*) = ((G_{ij} L)([\mathbf{x}, \overline{\mathbf{x}}][\mathbf{x}, \overline{\mathbf{x}}]^*))_{i,j=1}^m. \tag{1.20}$$

Given a (truncated) linear functional $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]_{2t}^*$, its (truncated) localizing matrix is defined as follows:

$$M_t(G \otimes L) := L(G \otimes [\mathbf{x}, \overline{\mathbf{x}}]_{t-d_G} [\mathbf{x}, \overline{\mathbf{x}}]_{t-d_G}^*), \tag{1.21}$$

where we subtract

$$d_G := \max_{i,j \in [m]} \lceil \frac{\deg(G_{ij})}{2} \rceil$$

from the degree bound to account for the effect of multiplying the monomials with the entries of G. We collect some observations pertaining to $G \otimes L$ and its moment matrix $M(G \otimes L)$.

The matrix $M(G \otimes L_{\mathbf{a}})$ ***has a tensor product structure.*** For a (scalar-valued) evaluation map $L_{\mathbf{a}}$ and some polynomial matrix G, the moment matrix of $G \otimes L_{\mathbf{a}}$ is

$$\begin{aligned} M(G \otimes L_{\mathbf{a}}) &= L_{\mathbf{a}}(G \otimes [\mathbf{x}, \overline{\mathbf{x}}][\mathbf{x}, \overline{\mathbf{x}}]^*) \\ &= L_{\mathbf{a}}(G) \otimes L_{\mathbf{a}}([\mathbf{x}, \overline{\mathbf{x}}][\mathbf{x}, \overline{\mathbf{x}}]^*) = G(\mathbf{a}, \overline{\mathbf{a}}) \otimes [\mathbf{a}, \overline{\mathbf{a}}][\mathbf{a}, \overline{\mathbf{a}}]^*. \end{aligned} \tag{1.22}$$

Thus, if $G(\mathbf{a}, \overline{\mathbf{a}}) \succeq 0$, then $M(G \otimes L_{\mathbf{a}}) \succeq 0$. Hence, if L is a conic combination of evaluation maps $L = \sum_{\mathbf{a} \in A} L_{\mathbf{a}}$ and $G(\mathbf{a}, \overline{\mathbf{a}}) \succeq 0$ for all $\mathbf{a} \in A$, then the moment matrix $M(G \otimes L)$ is Hermitian PSD.

The following is a corollary of Lemma 1.4 and Lemma 1.5 for the truncated setting when $\mathcal{L}$ is of the form $G \otimes L$.

Corollary 1.6. *Let $t \in \mathbb{N} \cup \{\infty\}$, $G \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^{m \times m}$, and $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*_{2t}$. If*

$$M_{t-d_G}(G \otimes L) = L(G \otimes [\mathbf{x}, \overline{\mathbf{x}}]_{t-d_G} [\mathbf{x}, \overline{\mathbf{x}}]^*_{t-d_G}) \succeq 0,$$

then $G \otimes L$ is positive on $\Sigma_{2(t-d_G)}$, i.e., both of the following two conditions hold:

$$L(\mathbf{v}^* G \mathbf{v} \cdot p\overline{p}) \geq 0 \ \textit{ for all } \mathbf{v} \in \mathbb{C}^m \textit{ and } p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]_{t-d_G},$$
$$M_{t-d_G}((\mathbf{v}^* G \mathbf{v}) L) \succeq 0 \ \textit{ for all } \mathbf{v} \in \mathbb{C}^m.$$

1.4. The real analogs of complex objects

Thus far, this chapter has been phrased in terms of complex objects. However, sometimes we work over the reals, in which case things simplify substantially. Indeed, observe that if $\mathbf{x} \in \mathbb{R}^n$, there is no need for conjugates $\overline{\mathbf{x}}$; as such, the vector of monomials becomes $[\mathbf{x}] := (\mathbf{x}^\alpha)_{\alpha \in \mathbb{N}^n}$. The ring of real polynomials in $\mathbf{x}$ is $\mathbb{R}[\mathbf{x}]$, the dual space $\mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ becomes $\mathbb{R}[\mathbf{x}]^*$, and so forth. The notion of Hermitian is not applicable in the real setting; as such, Hermitian matrices become just symmetric. The cone of symmetric real $n \times n$ matrices is written as $\mathcal{S}^n$, and the cone of PSD symmetric real $n \times n$ matrices is $\mathcal{S}^n_+$.

Some core classical results (like the forthcoming Theorem 2.8 in Chapter 2 and Lemma 3.1 in Chapter 3) are stated with real variables. We will use these results in Chapter 7 to lower bound the separable rank of a complex-valued matrix. To this end, we must express complex results in terms of their familiar real analogs. We give some miscellaneous conversion results for the reader's convenience. Readers may postpone a thorough readthrough of this chapter until they wish to delve into Chapter 7.

Vectors and matrices. Let $\mathbf{i} := \sqrt{-1} \in \mathbb{C}$ denote the *imaginary unit.* Any complex scalar $x \in \mathbb{C}$ can be written (uniquely) as $x = x_{\text{Re}} + \mathbf{i} x_{\text{Im}}$, where $x_{\text{Re}} := \text{Re}(x)$ and $x_{\text{Im}} := \text{Im}(x)$ denote, respectively, the real and imaginary parts of x. This notation extends to vectors and matrices by letting the maps $\text{Re}(\cdot)$ and $\text{Im}(\cdot)$ act entry-wise. Any vector $\mathbf{x} \in \mathbb{C}^n$ can be written $\mathbf{x} = \mathbf{x}_{\text{Re}} + \mathbf{i}\mathbf{x}_{\text{Im}}$ with $\mathbf{x}_{\text{Re}} := \text{Re}(\mathbf{x})$, $\mathbf{x}_{\text{Im}} := \text{Im}(\mathbf{x}) \in \mathbb{R}^n$. This gives a bijection

$$\phi : \mathbb{C}^n \ni \mathbf{x} \mapsto (\mathbf{x}_{\text{Re}}, \mathbf{x}_{\text{Im}}) \in \mathbb{R}^n \times \mathbb{R}^n. \tag{1.23}$$

Similarly, for a complex matrix $G \in \mathbb{C}^{m \times m'}$, set $G_{\text{Re}} := \text{Re}(G), G_{\text{Im}} := \text{Im}(G) \in \mathbb{R}^{m \times m'}$ and define the $2m \times 2m'$ real matrix

$$G^{\mathbb{R}} := \begin{bmatrix} G_{\text{Re}} & -G_{\text{Im}} \\ G_{\text{Im}} & G_{\text{Re}} \end{bmatrix}. \tag{1.24}$$

Then, $G \in \mathbb{C}^{m \times m}$ is Hermitian, i.e., $G = G^*$, if and only if $G_{\text{Re}} = G_{\text{Re}}^T$ and $G_{\text{Im}}^T = -G_{\text{Im}}$. Moreover, for a Hermitian matrix $G \in \mathbb{C}^{m \times m}$ and a complex vector $\mathbf{w} \in \mathbb{C}^m$ we have the identity

$$\begin{aligned} \mathbf{w}^* G \mathbf{w} &= (\mathbf{w}_{\mathrm{Re}} - \mathbf{i}\mathbf{w}_{\mathrm{Im}})^T (G_{\mathrm{Re}} + \mathbf{i} G_{\mathrm{Im}})(\mathbf{w}_{\mathrm{Re}} + \mathbf{i}\mathbf{w}_{\mathrm{Im}}) \\ &= \begin{bmatrix} \mathbf{w}_{\mathrm{Re}}^T & \mathbf{w}_{\mathrm{Im}}^T \end{bmatrix} \begin{bmatrix} G_{\mathrm{Re}} & -G_{\mathrm{Im}} \\ G_{\mathrm{Im}} & G_{\mathrm{Re}} \end{bmatrix} \begin{bmatrix} \mathbf{w}_{\mathrm{Re}} \\ \mathbf{w}_{\mathrm{Im}} \end{bmatrix}, \end{aligned} \tag{1.25}$$

which implies the well-known equivalence

$$G \succeq 0 \iff G^{\mathbb{R}} = \begin{bmatrix} G_{\mathrm{Re}} & -G_{\mathrm{Im}} \\ G_{\mathrm{Im}} & G_{\mathrm{Re}} \end{bmatrix} \succeq 0.$$

Polynomials. Polynomials in $\mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$ with complex variables $\mathbf{x} \in \mathbb{C}^n$ can be transformed into polynomials in $\mathbb{R}[\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}]$ with real variables $\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}} \in \mathbb{R}^n$, via the change of variables $\mathbf{x} = \mathbf{x}_{\mathrm{Re}} + \mathbf{i}\mathbf{x}_{\mathrm{Im}}$. In this way, any $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$ corresponds to a unique pair of real polynomials

$$\begin{aligned} p_{\mathrm{Re}}(\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}) &:= \mathrm{Re}(p(\mathbf{x}_{\mathrm{Re}} + \mathbf{i}\mathbf{x}_{\mathrm{Im}}, \mathbf{x}_{\mathrm{Re}} - \mathbf{i}\mathbf{x}_{\mathrm{Im}})) \in \mathbb{R}[\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}], \\ p_{\mathrm{Im}}(\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}) &:= \mathrm{Im}(p(\mathbf{x}_{\mathrm{Re}} + \mathbf{i}\mathbf{x}_{\mathrm{Im}}, \mathbf{x}_{\mathrm{Re}} - \mathbf{i}\mathbf{x}_{\mathrm{Im}})) \in \mathbb{R}[\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}], \end{aligned}$$

satisfying the following identity:

$$p(\mathbf{x}, \overline{\mathbf{x}}) = p(\mathbf{x}_{\mathrm{Re}} + \mathbf{i}\mathbf{x}_{\mathrm{Im}}, \mathbf{x}_{\mathrm{Re}} - \mathbf{i}\mathbf{x}_{\mathrm{Im}}) = p_{\mathrm{Re}}(\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}) + \mathbf{i} p_{\mathrm{Im}}(\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}). \tag{1.26}$$

Note that the degrees are preserved because

$$\deg_{\mathbf{x}, \overline{\mathbf{x}}}(p) = \max\{\deg_{\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}}(p_{\mathrm{Re}}),\ \deg_{\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}}(p_{\mathrm{Im}})\}.$$

A polynomial p is Hermitian, i.e., $\overline{p} = p$, if and only if its imaginary component is zero, i.e., $p_{\mathrm{Im}} = 0$. As a consequence, the Re map is injective on Hermitian polynomials:

$$\mathrm{Re}:\ \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h \ni p(\mathbf{x}, \overline{\mathbf{x}}) \mapsto p_{\mathrm{Re}}(\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}) \in \mathbb{R}[\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}]. \tag{1.27}$$

The Re map is also surjective. Take any $f \in \mathbb{R}[\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}]$ and define the polynomial $p(\mathbf{x}, \overline{\mathbf{x}}) := f(\frac{\mathbf{x}+\overline{\mathbf{x}}}{2}, \frac{\mathbf{x}-\overline{\mathbf{x}}}{2\mathbf{i}}) \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$, then p is Hermitian and satisfies $f = p_{\mathrm{Re}}$. Using this, we can, for any $h \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$, recast the constraint $h(\mathbf{x}, \overline{\mathbf{x}}) = 0$ as $h_{\mathrm{Re}}(\frac{\mathbf{x}+\overline{\mathbf{x}}}{2}, \frac{\mathbf{x}-\overline{\mathbf{x}}}{2\mathbf{i}}) = 0$ and $h_{\mathrm{Im}}(\frac{\mathbf{x}+\overline{\mathbf{x}}}{2}, \frac{\mathbf{x}-\overline{\mathbf{x}}}{2\mathbf{i}}) = 0$, which involves only Hermitian polynomials h_{Re} and h_{Im}. Finally, since any $p\overline{p}$ is Hermitian, we have that the Re map preserves sums of squares, i.e.,

$$\mathrm{Re}(p\overline{p}) = p_{\mathrm{Re}}^2 + p_{\mathrm{Im}}^2.$$

By linearity, this means that sums of Hermitian squares in $\mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$ are mapped to sums of (real) squares in $\mathbb{R}[\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}]$ and vice versa.

Polynomial matrices. For vectors and matrices with polynomial entries in $\mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$, the maps $\mathrm{Re}(\cdot)$ and $\mathrm{Im}(\cdot)$ act entry-wise. Additionally, for a polynomial matrix $G \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^{m \times m'}$, we can define the real polynomial matrix

$G^{\mathbb{R}} \in \mathbb{R}[\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}]^{2m\times 2m'}$ using relation (1.24), where $G_{\mathrm{Re}}, G_{\mathrm{Im}}$ are defined entry-wise as follows:

$$\text{if } G = (G_{ij})_{i,j\in[m]} \text{ then } G_{\mathrm{Re}} := ((G_{ij})_{\mathrm{Re}})_{i,j\in[m]} \text{ and } G_{\mathrm{Im}} := ((G_{ij})_{\mathrm{Im}})_{i,j\in[m]}.$$

From the above definition, G is Hermitian if and only if $G^{\mathbb{R}}$ is symmetric. Next, observe that this correspondence extends to polynomial matrix sums of squares.

LEMMA 1.7. *Let $G \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^{m\times m}$ be a polynomial matrix and let $G^{\mathbb{R}} \in \mathbb{R}[\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}]^{2m\times 2m}$ be the corresponding real polynomial matrix defined via (1.24). Then, G is a Hermitian SoS-polynomial matrix if and only if $G^{\mathbb{R}}$ is a (real) SoS-polynomial matrix.*

PROOF. Assume G is a Hermitian SoS-polynomial matrix. Let $G = UU^*$ with $U \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^{m\times k}$. Applying the change of variables from complex to real, we get

$$\begin{aligned} G(\mathbf{x}, \overline{\mathbf{x}}) &= G_{\mathrm{Re}}(\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}) + \mathbf{i}G_{\mathrm{Im}}(\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}) \\ &= U(\mathbf{x}_{\mathrm{Re}} + \mathbf{i}\mathbf{x}_{\mathrm{Im}}, \mathbf{x}_{\mathrm{Re}} - \mathbf{i}\mathbf{x}_{\mathrm{Im}})U^*(\mathbf{x}_{\mathrm{Re}} + \mathbf{i}\mathbf{x}_{\mathrm{Im}}, \mathbf{x}_{\mathrm{Re}} - \mathbf{i}\mathbf{x}_{\mathrm{Im}}) \\ &= (U_{\mathrm{Re}} + \mathbf{i}U_{\mathrm{Im}})(U_{\mathrm{Re}}^T - \mathbf{i}U_{\mathrm{Im}}^T) \\ &= U_{\mathrm{Re}}U_{\mathrm{Re}}^T + U_{\mathrm{Im}}U_{\mathrm{Im}}^T + \mathbf{i}\big(U_{\mathrm{Im}}U_{\mathrm{Re}}^T - U_{\mathrm{Re}}U_{\mathrm{Im}}^T\big). \end{aligned}$$

This implies $G_{\mathrm{Re}} = U_{\mathrm{Re}}U_{\mathrm{Re}}^T + U_{\mathrm{Im}}U_{\mathrm{Im}}^T$ and $G_{\mathrm{Im}} = U_{\mathrm{Im}}U_{\mathrm{Re}}^T - U_{\mathrm{Re}}U_{\mathrm{Im}}^T$. Thus

$$\begin{aligned} G^{\mathbb{R}} := \begin{bmatrix} G_{\mathrm{Re}} & -G_{\mathrm{Im}} \\ G_{\mathrm{Im}} & G_{\mathrm{Re}} \end{bmatrix} &= \begin{bmatrix} U_{\mathrm{Re}}U_{\mathrm{Re}}^T + U_{\mathrm{Im}}U_{\mathrm{Im}}^T & -(U_{\mathrm{Im}}U_{\mathrm{Re}}^T - U_{\mathrm{Re}}U_{\mathrm{Im}}^T) \\ U_{\mathrm{Im}}U_{\mathrm{Re}}^T - U_{\mathrm{Re}}U_{\mathrm{Im}}^T & U_{\mathrm{Re}}U_{\mathrm{Re}}^T + U_{\mathrm{Im}}U_{\mathrm{Im}}^T \end{bmatrix} \\ &= \begin{bmatrix} U_{\mathrm{Re}} & -U_{\mathrm{Im}} \\ U_{\mathrm{Im}} & U_{\mathrm{Re}} \end{bmatrix} \begin{bmatrix} U_{\mathrm{Re}}^T & U_{\mathrm{Im}}^T \\ -U_{\mathrm{Im}}^T & U_{\mathrm{Re}}^T \end{bmatrix} = U^{\mathbb{R}}(U^{\mathbb{R}})^T, \end{aligned}$$

which shows $G^{\mathbb{R}}$ is an SoS-polynomial matrix. The converse result follows from retracing the above steps. □

Linear functionals. A linear functional $L : \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}] \to \mathbb{C}$ decomposes into real and imaginary parts $L(p) = \mathrm{Re}(L(p)) + \mathbf{i}\mathrm{Im}(L(p))$ for all $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$. Recall that L is Hermitian if $\overline{L(p)} = L(\overline{p})$. Just as the case of Hermitian polynomials, Hermitian functionals L map injectively to real linear functionals $L^{\mathbb{R}} : \mathbb{R}[\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}] \to \mathbb{R}$ by

$$L^{\mathbb{R}}(f) := L\big(f\big(\frac{\mathbf{x} + \overline{\mathbf{x}}}{2}, \frac{\mathbf{x} - \overline{\mathbf{x}}}{2\mathbf{i}}\big)\big) \quad \text{for any} \quad f \in \mathbb{R}[\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}]. \tag{1.28}$$

For a Hermitian polynomial $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h$, by (1.27) we have $p_{\mathrm{Re}}(\frac{\mathbf{x}+\overline{\mathbf{x}}}{2}, \frac{\mathbf{x}-\overline{\mathbf{x}}}{2\mathbf{i}}) = p(\mathbf{x}, \overline{\mathbf{x}})$ and thus

$$L(p) = L^{\mathbb{R}}(p_{\mathrm{Re}}) \text{ for any } p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h. \tag{1.29}$$

Then, for any $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$, we have

$$L(p) = L\big(p_{\mathrm{Re}}(\frac{\mathbf{x}+\overline{\mathbf{x}}}{2}, \frac{\mathbf{x}-\overline{\mathbf{x}}}{2\mathbf{i}})\big)+\mathbf{i}L\big(p_{\mathrm{Im}}(\frac{\mathbf{x}+\overline{\mathbf{x}}}{2}, \frac{\mathbf{x}-\overline{\mathbf{x}}}{2\mathbf{i}})\big) = L^{\mathbb{R}}(p_{\mathrm{Re}})+\mathbf{i}L^{\mathbb{R}}(p_{\mathrm{Im}}). \tag{1.30}$$

In particular, we have $L(p\overline{p}) = L^{\mathbb{R}}(p_{\mathrm{Re}}^2+p_{\mathrm{Im}}^2)$ for any $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$. This implies that L is positive (on sums of Hermitian squares) if and only if $L^{\mathbb{R}}$ is positive (on sums of real squares). Since $\mathrm{Re}(\cdot)$ preserves degrees, the restriction of $L^{\mathbb{R}}$ to $\mathbb{R}[\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}]_t$ corresponds to the restriction of L to $\mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]_t$.

Matrix-valued linear functionals. Consider a complex matrix-valued linear map, seen before in (1.4):

$$\mathcal{L} : \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}] \ni p \mapsto \mathcal{L}(p) := \big(L_{ij}(p)\big)_{i,j\in[m]} \in \mathbb{C}^{m\times m},$$

where each $L_{ij} : \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}] \to \mathbb{C}$ is a scalar-valued linear functional. Recall that the map $\mathcal{L}$ is Hermitian if and only if, for all $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$, we have $\mathcal{L}(\overline{p}) = \mathcal{L}(p)^*$. In terms of real and imaginary components, this means that

$$\Big(\mathrm{Re}(L_{ij}(\overline{p})) + \mathbf{i}\mathrm{Im}(L_{ij}(\overline{p}))\Big)_{i,j=1}^m = \Big(\mathrm{Re}(L_{ji}(p)) - \mathbf{i}\mathrm{Im}(L_{ji}(p))\Big)_{i,j=1}^m.$$

Equivalently, we have $\mathrm{Re}(L_{i,j}(\overline{p})) = \mathrm{Re}(L_{j,i}(p))$ and $\mathrm{Im}(L_{i,j}(\overline{p})) = -\mathrm{Im}(L_{j,i}(p))$ for all $i, j \in [m]$. Hence, if $\mathcal{L}$ is Hermitian and p is Hermitian, then the complex matrix $\mathcal{L}(p)$ is Hermitian. Assume that $\mathcal{L}$ is Hermitian. Then, we define the real analog matrix-valued linear functional

$$\mathcal{L}^{\mathbb{R}} : \mathbb{R}[\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}] \ni f \in \mathbb{R}[\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}] \mapsto \mathcal{L}^{\mathbb{R}}(f) \in \mathbb{R}^{2m\times 2m},$$

$$\begin{aligned}\mathcal{L}^{\mathbb{R}}(f) &:= \Big(\mathcal{L}(f(\frac{\mathbf{x}+\overline{\mathbf{x}}}{2}, \frac{\mathbf{x}-\overline{\mathbf{x}}}{2i}))\Big)^{\mathbb{R}} \\ &= \begin{bmatrix}\mathrm{Re}(\mathcal{L}(f(\frac{\mathbf{x}+\overline{\mathbf{x}}}{2}, \frac{\mathbf{x}-\overline{\mathbf{x}}}{2\mathbf{i}}))) & -\mathrm{Im}(\mathcal{L}(f(\frac{\mathbf{x}+\overline{\mathbf{x}}}{2}, \frac{\mathbf{x}-\overline{\mathbf{x}}}{2\mathbf{i}}))) \\ \mathrm{Im}(\mathcal{L}(f(\frac{\mathbf{x}+\overline{\mathbf{x}}}{2}, \frac{\mathbf{x}-\overline{\mathbf{x}}}{2\mathbf{i}}))) & \mathrm{Re}(\mathcal{L}(f(\frac{\mathbf{x}+\overline{\mathbf{x}}}{2}, \frac{\mathbf{x}-\overline{\mathbf{x}}}{2\mathbf{i}})))\end{bmatrix}.\end{aligned} \tag{1.31}$$

Since $f(\frac{\mathbf{x}+\overline{\mathbf{x}}}{2}, \frac{\mathbf{x}-\overline{\mathbf{x}}}{2\mathbf{i}})$ is Hermitian, it follows that

$$-\mathrm{Im}(\mathcal{L}(f(\frac{\mathbf{x}+\overline{\mathbf{x}}}{2}, \frac{\mathbf{x}-\overline{\mathbf{x}}}{2\mathbf{i}}))) = \mathrm{Im}(\mathcal{L}(f(\frac{\mathbf{x}+\overline{\mathbf{x}}}{2}, \frac{\mathbf{x}-\overline{\mathbf{x}}}{2\mathbf{i}})))^T.$$

Hence, $\mathcal{L}^{\mathbb{R}}$ takes its values in the cone $\mathcal{S}^{2m}$ of symmetric matrices.

Lemma 1.8. *Given a Hermitian linear map $\mathcal{L} : \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}] \to \mathbb{C}^{m\times m}$ and the corresponding map $\mathcal{L}^{\mathbb{R}}$ from (1.31), $g \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h$ and $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$, we have the following equivalence*

$$\mathcal{L}(gp\overline{p}) \succeq 0 \iff \mathcal{L}^{\mathbb{R}}(g_{\mathrm{Re}}(p_{\mathrm{Re}}^2 + p_{\mathrm{Im}}^2)) \succeq 0.$$

Proof. From (1.24), (1.27), and (1.31) we obtain that

$$0 \preceq \mathcal{L}(gp\overline{p}) \iff 0 \preceq \begin{bmatrix}\mathrm{Re}(\mathcal{L}(gp\overline{p})) & -\mathrm{Im}(\mathcal{L}(gp\overline{p})) \\ \mathrm{Im}(\mathcal{L}(gp\overline{p})) & \mathrm{Re}(\mathcal{L}(gp\overline{p}))\end{bmatrix} = \mathcal{L}^{\mathbb{R}}(g_{\mathrm{Re}}(p_{\mathrm{Re}}^2 + p_{\mathrm{Im}}^2)),$$

because $gp\overline{p}$ is Hermitian. □

CHAPTER 2

The generalized moment problem

We approach generalized moment problems (GMP) from the perspective of linear optimization problems over measures. This should be understood in contradistinction to “the moment problem”, which is a classical topic where one seeks a representing measure for a given (partial) set of moments. Moment problems have been actively studied for at least a century, and as such, the field is very rich and broad in applications; see, e.g., Akhiezer [**4**], Schmüdgen [**142**], and Lasserre [**106**]. We will also follow the recent survey [**42**].

First, we state some notation and classical results involving measures, GMPs, and optimization problems (Section 2.1).

Second, we introduce the novel notion of ideal sparsity in GMPs (Section 2.2), which we developed in [**100**]. Ideal sparsity is a new technique for reformulating a GMP with monomial ideal constraints into an equivalent GMP without ideal constraints, now involving more measures with smaller supports than the measure in the original GMP. Though the resulting GMP is equivalent, its associated moment hierarchy often yields better bounds and faster computations than the analogous hierarchies for the original GMP.

Third, we establish well-known and fundamental links between linear functionals $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ and measures. By doing so, we can recast GMPs as optimization problems over linear functionals (Section 2.3) and thereby open the way to the moment method, which we will treat in the next chapter.

2.1. Preliminaries

This section introduces a few basic but widely used terms from measure theory and polynomial optimization. Our goal is not a full exposition but rather a quick primer to ease the reader into the field before stating our contributions. We leave adequate references to more informative sources.

Measures. Let $\mathcal{X} \subseteq \mathbb{C}^n$ be a set and $\mathcal{S}$ a σ-algebra over $\mathcal{X}$. Recall that $\mathcal{S} \subseteq \mathcal{P}(\mathcal{X})$ is *σ-algebra over $\mathcal{X}$* if it satisfies:

(i) $\emptyset \in \mathcal{S}$,
(ii) $S \in \mathcal{S} \implies \mathcal{X} \setminus S \in \mathcal{S}$,
(iii) For any $S_0, S_1, S_2, ... \in \mathcal{S}$ we have $\bigcup_{i \in \mathbb{N}} S_i \in \mathcal{S}$.

The pair $(\mathcal{X}, \mathcal{S})$ is a *measurable space.* A function μ that maps the elements of $\mathcal{S}$ to the extended real line $\mathbb{R} \cup \{\infty\}$ is called a (nonnegative) measure if it satisfies the following three properties:

(i) *Null empty set*: $\mu(\emptyset) = 0$,
(ii) *Nonnegativity*: $\mu(S) \geq 0$ for all $S \in \mathcal{S}$,
(iii) *σ-additivity*: For all countable collections $\left\{S_k\right\}_{k\in\mathbb{N}}$ of pairwise disjoint sets in $\mathcal{S}$, we have $\mu(\bigcup_{k\in\mathbb{N}} S_k) = \sum_{k\in\mathbb{N}} \mu(S_k)$.

A measure μ is called finite if it takes values in $\mathbb{R}$. When $\mu(\mathcal{X}) = 1$ we call μ a *probability measure.* A *measurable function* f is a map between two measurable spaces $(\mathcal{X}, \mathcal{S})$ and $(\mathcal{Y}, \mathcal{T})$ with the property that, for any set $T \in \mathcal{T}$, its preimage is a set in the σ-algebra $\mathcal{S}$, i.e., $f^{-1}(T) \in \mathcal{S}$. The *support* supp$(\mu)$ *of a measure* μ is defined to be the intersection of all measurable sets $S \in \mathcal{S}$ with the property that $\mu(\mathcal{X} \setminus S) = 0$, i.e.,

$$\text{supp}(\mu) := \bigcap_{S\in\mathcal{S}:\mu(\mathcal{X}\setminus S)=0} S.$$

So as not to derail the topic of this chapter further with measure-theoretic definitions, we refer the reader to [**93**] for a proper treatment of measures and integration. From this point forward, we assume the reader has at least an intuitive grasp of integration. The support of a measure can also be characterized in terms of measurable functions and integrals as follows: supp(μ) is contained in a set $K \subseteq \mathbb{R}^n$ if

$$\int f d\mu = \int_K f d\mu \text{ for any measurable function } f : \mathbb{R}^n \to \mathbb{R}.$$

***Dirac delta measure* $\delta_{\mathbf{x}}$.** As an example of a measure, we present the *Dirac delta measure $\delta_{\mathbf{x}}$ supported at the point* $\mathbf{x} \in \mathbb{R}^n$. For any measurable set A and measurable function f, $\delta_{\mathbf{x}}$ is acts as follows:

$$\int_A f d\delta_{\mathbf{x}} = \begin{cases} f(\mathbf{x}) & \mathbf{x} \in A, \\ 0 & \mathbf{x} \notin A. \end{cases}$$

A measure μ is called *finite atomic* if it is a weighted sum of Dirac delta measures, i.e.,

$$\mu = \sum_{\ell\in[N]} c_\ell \delta_{\mathbf{x}^{(\ell)}},$$

for some points $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, ..., \mathbf{x}^{(N)} \in \mathbb{R}^n$ (called the atoms of μ) and scalars $c_1, c_2, ..., c_N \in \mathbb{R}_+$ (called the weights of μ).

Borel measures. When $\mathcal{X}$ is a Euclidean space like $\mathbb{C}^n$ or $\mathbb{R}^n$, the Euclidean norm induces a topology, which induces a σ-algebra $\mathcal{B}(\mathcal{X})$. The elements of $\mathcal{B}(\mathcal{X})$ are generated by the familiar open and closed sets of $\mathbb{C}^n$. We denote the space of finite positive Borel measures supported on a set $K \subset \mathcal{X}$ by $\mathscr{M}(K)$.

2.1.1. Optimization problems. We consider the following optimization problem over (finite positive) Borel measures on $\mathbb{R}^n$:

$$\mathbf{val} := \inf_{\mu \in \mathscr{M}(K)} \Big\{ \int f_0 d\mu : \int f_i d\mu = a_i \ (i \in [N_f]) \Big\}. \tag{2.1}$$

Here, $f_0, f_1, ..., f_{N_f} \in \mathbb{R}[\mathbf{x}]$ are polynomials, $a_1, ..., a_{N_f} \in \mathbb{R}$ are scalars, and

$$K = \{\mathbf{x} \in \mathbb{R}^n : g_i(\mathbf{x}) \geq 0 \ (i \in [N_g]), \ h_j(\mathbf{x}) = 0 \ (j \in [N_h])\} \tag{2.2}$$

is a basic closed semi-algebraic set defined by polynomials $g_1, ..., g_{N_g}, h_1, ..., h_{N_h}$. Problem (2.1) is an instance of what is called a *generalized moment problem* (abbreviated as GMP). GMPs are extremely potent modeling tools and have received much attention recently, partly because measures are extremely rich in descriptive power. Their uses include polynomial optimization (minimization of a polynomial or rational function over K), volume computation, control theory, option pricing in finance, and much more. See, e.g., [**105, 106, 107, 84**] and further references therein.

As an illustration, we demonstrate how a polynomial optimization problem can be phrased as a special instance of a GMP.

Global optimization over polynomials. Consider the constrained optimization problem

$$f^* := \inf f(\mathbf{x}) \text{ s.t. } \mathbf{x} \in K, \tag{2.3}$$

where $K \subseteq \mathbb{R}^n$ is as in (2.2). Problem (2.3) can be recast in the language of measures.

THEOREM 2.1. *Problem (2.3) is equivalent to the GMP (2.4) in the sense that they have the same optimal values, i.e.,*

$$f^* = f^*_{\mathrm{GMP}} := \inf_{\mu \in \mathscr{M}(K)} \int f d\mu \text{ s.t.} \int d\mu = 1. \tag{2.4}$$

PROOF. To simplify the exposition, we assume K is compact so that f is guaranteed to have a global optimizer over it.

($f^* \geq f^*_{\mathrm{GMP}}$) For any global minimizer $\mathbf{x}^*$ of (2.3), one can readily observe that the Dirac measure $\delta_{\mathbf{x}^*}$ supported at $\mathbf{x}^*$ is a feasible solution to (2.4) with objective value $\int f d\delta_{\mathbf{x}^*} = f(\mathbf{x}^*) = f^*$.

($f^* \leq f^*_{\mathrm{GMP}}$) Observe that for any probability measure $\mu \in \mathscr{M}(K)$ we have

$$\int f d\mu \geq f^* \cdot \int d\mu = f^*. \qquad \Box$$

In the case when an optimizer μ of (2.4) is finite atomic, i.e., of the form

$$\mu = \sum_{\ell \in [N]} c_\ell \delta_{\mathbf{x}^{(\ell)}},$$

with positive weights $c_1, ..., c_N > 0$, the atoms $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, ..., \mathbf{x}^{(N)} \in \mathbb{R}^n$ are global optimizers of (2.3).

An interesting (and critical for Part 2 of this thesis) observation is that one can also have a polynomial matrix constraint defining the semi-algebraic set. For example, if $G \in (\mathbb{R}[\mathbf{x}])^{m\times m}$ is a polynomial matrix, then we can define its positivity domain in two equivalent ways

$$\{\mathbf{x} \in \mathbb{R}^n : G(\mathbf{x}) \succeq 0\} = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{v}^T G(\mathbf{x})\mathbf{v} \geq 0 \ (\mathbf{v} \in \mathbb{R}^m)\}.$$

The former has only one semi-definite constraint ($G(\mathbf{x}) \succeq 0$), and the latter has (uncountably) infinitely many scalar constraints ($\mathbf{v}^T G(\mathbf{x})\mathbf{v} \geq 0$). Applications of these constraints emerge naturally in the setting of matrix factorization rank, which we briefly discuss next.

Matrix factorization rank. A nonnegative matrix $A \in \mathbb{R}^{n\times n}_+$ is called *completely positive* (CP) if, for some $r \in \mathbb{N}$, there exists a nonnegative matrix $B \in \mathbb{R}^{r\times n}_+$, with the property that

$$A = B^T B.$$

The *completely positive rank* (CP-rank) of a CP matrix A, denoted $\mathrm{rank}_{\mathrm{cp}}(A)$, is the smallest positive integer $r \in \mathbb{N}$ such that this factorization is possible. In other words,

$$\mathrm{rank}_{\mathrm{cp}}(A) := \min\{r \in \mathbb{N} : \exists B \in \mathbb{R}^{r\times n} \text{ s.t. } A = B^T B\}.$$

By convention, if A is not CP, then we set $\mathrm{rank}_{\mathrm{cp}}(A) = \infty$. A crucial observation made by Fawzi and Parilo in [**67**] is that $\mathrm{rank}_{\mathrm{cp}}(A)$ can be lower bounded by the following GMP:

$$\tau_{\mathrm{cp}}(A) := \inf_{\mu \in \mathscr{M}(K_A)} \Big\{ \int_{K_A} 1 d\mu : \int_{K_A} x_i x_j d\mu = A_{ij} \ (i, j \in V) \Big\}, \tag{2.5}$$

where the semi-algebraic set is given by

$$\begin{aligned} K_A = \{x \in \mathbb{R}^n : \ & \sqrt{A_{ii}} x_i - x_i^2 \geq 0 \ (i \in [n]), \\ & A_{ij} - x_i x_j \geq 0 \ (\{i, j\} \in E_A), \\ & x_i x_j = 0 \ (\{i, j\} \in \overline{E}_A), \\ & A - \mathbf{x}\mathbf{x}^T \succeq 0\}, \end{aligned} \tag{2.6}$$

where

$$E_A := \Big\{\{i, j\} : A_{ij} \neq 0, \ i, j \in V, \ i \neq j\Big\},$$

$$\overline{E}_A := \Big\{\{i, j\} : A_{ij} = 0, \ i, j \in V, \ i \neq j\Big\}.$$

Note the polynomial matrix constraint $A - \mathbf{x}\mathbf{x}^T \succeq 0$ in the definition of K_A. We will look in depth at the CP-rank in Chapter 6, where we will give quick proof that $\tau_{\mathrm{cp}}(A) \leq \mathrm{rank}_{\mathrm{cp}}(A)$ following (6.4). Though other choices for K_A

would suffice to define $\tau_{\mathrm{cp}}(A)$, we motivate this particular definition of K_A later in (6.6).

2.2. Ideal sparsity

Recall the GMP in (2.1). Now, we assume that K is a (basic closed) semi-algebraic set involving equality constraints of a special form, namely,

$$K = \Big\{\mathbf{x} \in \mathbb{R}^n : g_j(\mathbf{x}) \geq 0 \ (j \in [N_g]),\ \mathbf{x}^S := \prod_{i \in S} x_i = 0 \ (S \in \mathcal{S})\Big\}, \tag{2.7}$$

where $g_1, ..., g_{N_g} \in \mathbb{R}[\mathbf{x}]$ are polynomials and $\mathcal{S} \subseteq \mathcal{P}(V)$ is a collection of subsets of $V = [n]$. Hence, the set K is contained in the set of all $\mathbf{x} \in \mathbb{R}^n$ for which every polynomial $p \in I_{\mathcal{S}}$ vanishes, where

$$I_{\mathcal{S}} := \Big\{ \sum_{S \in \mathcal{S}} u_S \mathbf{x}^S : u_S \in \mathbb{R}[\mathbf{x}] \Big\} \subseteq \mathbb{R}[\mathbf{x}]. \tag{2.8}$$

2.2.1. Ideal-sparse GMP. One can exploit the fact that $p(\mathbf{x}) = 0$ for every $\mathbf{x} \in K$ and $p \in I_{\mathcal{S}}$ to create a new GMP. Due to the special form of the ideal $I_{\mathcal{S}}$, many monomials are set to zero, and hence we can omit these monomials in the GMP formulation. We accomplish this by instead of optimizing over a *single* measure μ supported on $K \subseteq \mathbb{R}^n$, we optimize over *several* measures $\mu_1, ..., \mu_p$ that are each supported on a smaller space than the original μ. Loosely speaking, the supports of the measures $\mu_1, ..., \mu_p$ will be as large as possible without supporting any monomial of the form $\mathbf{x}^S$, where $S \in \mathcal{S}$. We have that this latter formulation gives an equivalent GMP. We will later show that its associated moment relaxations give possibly tighter bounds than those associated with the original GMP involving a single measure.

Covering the support K. To begin, let $V_1, ..., V_p$ denote the maximal subsets of $V := [n]$ such that $S \not\subseteq V_k$ for all $S \in \mathcal{S}$ and $k \in [p]$. Define, for each $k \in [p]$, the following subset of K:

$$\widehat{K_k} := \{\mathbf{x} \in K : \mathrm{supp}(\mathbf{x}) \subseteq V_k\} \subseteq K \subseteq \mathbb{R}^n. \tag{2.9}$$

Recall that $\mathrm{supp}(\mathbf{x}) := \{i \in [n] : x_i \neq 0\}$ denotes the *support* of vector $\mathbf{x} \in \mathbb{R}^n$. Observe that

$$K = \widehat{K_1} \cup ... \cup \widehat{K_p}. \tag{2.10}$$

This is because for any $\mathbf{x} \in K$ one must have that $S \not\subseteq \mathrm{supp}(\mathbf{x})$ for every $S \in \mathcal{S}$. Thus, for any $\mathbf{x} \in K$ there exists a $k \in [p]$ such that $\mathrm{supp}(\mathbf{x}) \subseteq V_k$. Hence, $\mathbf{x} \in \widehat{K_k}$ for some $k \in [p]$.

For each $k \in [p]$, we define the projection

$$K_k := \{\mathbf{y} \in \mathbb{R}^{|V_k|} : (\mathbf{y}, 0_{V \setminus V_k}) \in \widehat{K_k}\} \subseteq \mathbb{R}^{|V_k|} \tag{2.11}$$

of $\widehat{K_k}$ onto the subspace indexed by V_k. We can now state the ideal-sparse formulation of the problem (2.1).

The ideal-sparse GMP formulation. Assume that K is as defined in (2.7). Then, the ideal-sparse analog of GMP (2.1) is

$$\mathbf{val}^{\mathrm{isp}} := \inf_{\substack{\mu_k \in \mathscr{M}(K_k) \\ k \in [p]}} \Big\{ \sum_{k\in[p]} \int f_{0|V_k} d\mu_k : \sum_{k\in[p]} \int f_{i|V_k} d\mu_k = a_i \ (i \in [N_f]). \tag{2.12}$$

Here, we use the restriction notation $f_{i|V_k}$ to denote the function in $|V_k|$-many variables $\mathbf{x}(V_k) := \{x_i : i \in V_k\}$ obtained from f_i by setting to zero all the variables x_i indexed by $i \in V \setminus V_k$ (recall Section 1.1).

Observe the similar overall structure shared between (2.1) and (2.12). We note some key differences. While problem (2.1) optimizes over a single measure μ supported on the space $\mathbb{R}^{|V|}$, problem (2.12) involves p-many measures, where each μ_k is on the smaller dimensional space $\mathbb{R}^{|V_k|}$. We now show that both formulations (2.1) and (2.12) are equivalent, i.e., have equal optimal values: $\mathbf{val} = \mathbf{val}^{\mathrm{isp}}$. Here, and throughout the rest of the thesis, we use the superscript "isp" as a reminder that the formulation exploits ideal sparsity. Later chapters will use the same notation when defining the corresponding moment hierarchy and associated parameters.

PROPOSITION 2.2. *(Proposition 6 of* [**100**]*) Assume that K is as in (2.7). Then, problems (2.1) and (2.12) are equivalent, i.e., their optimum values are equal:*

$$\mathbf{val} = \mathbf{val}^{\mathrm{isp}}.$$

PROOF. **($\mathbf{val} \le \mathbf{val}^{\mathrm{isp}}$)** Assume $(\mu_1, ..., \mu_p)$ is feasible for problem (2.12). Consider the measure μ on $\mathbb{R}^{|V|}$, defined by $\int f d\mu = \sum_{k=1}^p \int_{K_k} f_{|V_k} d\mu_k$ for any measurable function f on $\mathbb{R}^{|V|}$. We have $\mathrm{supp}(\mu) \subseteq K$. Indeed,

$$\int_K f d\mu = \int f \chi^K d\mu = \sum_{k\in[p]} \int_{K_k} f_{|V_k} \chi^K_{|V_k} d\mu_k = \sum_{k\in[p]} \int_{K_k} f_{|V_k} d\mu_k = \int f d\mu,$$

since $\chi^K_{|V_k}(\mathbf{y}) = \chi^K(\mathbf{y}, 0_{V\setminus V_k}) = 1$ for all $\mathbf{y} \in K_k$ as $(\mathbf{y}, 0_{V\setminus V_k}) \in \widehat{K_k} \subseteq K$. Then, μ is feasible for (2.1), with the same objective value as $(\mu_1, ..., \mu_p)$, which shows $\mathbf{val} \le \mathbf{val}^{\mathrm{isp}}$.

($\mathbf{val}^{\mathrm{isp}} \le \mathbf{val}$) For the reverse inequality, assume μ is feasible for (2.1). We now define a feasible solution $(\mu_1, ..., \mu_p)$ to (2.12), with the same objective value as (2.1). For $k \in [p]$, define the set

$$\Lambda_k = \{\mathbf{x} \in K : \mathrm{supp}(\mathbf{x}) \subseteq V_k, \ \mathrm{supp}(\mathbf{x}) \not\subseteq V_h \text{ for } h \in [k-1]\}.$$

As each $\mathbf{x} \in K$ has its support contained in some V_k, it follows that the sets $\Lambda_1, ..., \Lambda_p$ form a partition of K. Note that $\Lambda_k \subseteq \widehat{K_k}$ and thus $\mathbf{x}(V_k) \in K_k$ for any $\mathbf{x} \in \Lambda_k$. Consider the measure μ_k on $\mathbb{R}^{|V_k|}$, defined, for any measurable function f on $\mathbb{R}^{|V_k|}$, by $\int f d\mu_k = \int_{\Lambda_k} f(\mathbf{x}(V_k)) d\mu(\mathbf{x})$. Then, $\operatorname{supp}(\mu_k) \subseteq K_k$, since

$$\begin{aligned}\int_{K_k} f d\mu_k = \int f \chi^{K_k} d\mu_k &= \int_{\Lambda_k} f(\mathbf{x}(V_k)) \chi^{K_k}(\mathbf{x}(V_k)) d\mu(\mathbf{x}) \\ &= \int_{\Lambda_k} f(\mathbf{x}(V_k)) d\mu(\mathbf{x}) = \int f d\mu_k,\end{aligned}$$

as $\chi^{K_k}(\mathbf{x}(V_k)) = 1$ for all $\mathbf{x} \in \Lambda_k$. Next, we show that $\int p d\mu = \sum_k \int p_{|V_k} d\mu_k$ for any measurable function $p : \mathbb{R}^{|V|} \to \mathbb{R}$. Indeed, as the sets $\Lambda_1, ..., \Lambda_p$ partition the set K, we have $\int p d\mu = \int_K p d\mu = \sum_k \int_{\Lambda_k} p d\mu$. Combining with

$$\int_{\Lambda_k} p(\mathbf{x}) d\mu(\mathbf{x}) = \int_{\Lambda_k} p_{|V_k}(\mathbf{x}(V_k)) d\mu(\mathbf{x}) = \int_{K_k} p_{|V_k} d\mu_k,$$

gives the desired identity $\int p d\mu = \sum_k \int p_{|V_k} d\mu_k$. Therefore, $(\mu_1, ..., \mu_p)$ is a feasible solution to (2.12) with the same objective value as μ, which shows the desired inequality $\mathbf{val}^{\text{isp}} \leq \mathbf{val}$. □

A special instance of ideal sparsity. We will consider a special instance for ideal sparsity, where the collection $\mathcal{S}$ corresponds to the set $\overline{E}$ of *nonedges* of some graph $G = (V, E)$. The semi-algebraic set K now takes the form of

$$K = \Big\{ \mathbf{x} \in \mathbb{R}^n : g_j(\mathbf{x}) \geq 0 \ (j \in [N_g]), \ x_i x_j = 0 \ (\{i, j\} \in \overline{E}) \Big\}.$$

The sets $V_1, ..., V_p$ can now be interpreted as the maximal cliques of the graph G.

This particular setting is motivated by its application to matrix factorization ranks, which we will elaborate on in Part 2 of this thesis. At this point, it suffices to say that the sparsity (presence of zeros in the matrix) of a symmetric matrix $A \in \mathbb{R}^{m \times m}_{+}$ can be captured by its support graph G_A and exploited to create a hierarchy of lower bounds on its completely positive rank. We leave the construction of the hierarchy to Chapter 3. Now we provide a concrete example of a matrix A, its support graph G_A, and its associated maximal cliques.

Example 2.3. ***Example matrix, support graph, and maximal cliques.*** *Consider the matrix A and its support graph G_A in Fig. 1.*

The maximal cliques of G_A are all the triangles $\mathcal{T} \subseteq \mathcal{P}(V)$ in G_A, namely $\mathcal{T} := \{\{1,3,5\}, \{1,3,6\}, \{1,4,5\}, \{2,3,6\}, \{2,3,5\}, \{1,4,5\}, \{1,4,5\}, \{2,4,5\}\}$. By definition, each clique $T \in \mathcal{T}$ corresponds to a sub-matrix $A_{T,T} \in \mathbb{R}^{|T| \times |T|}$ with no zero entries. This seemingly innocuous fact will be used extensively

$$A = \begin{pmatrix} 1 & 0 & 3 & 4 & 5 & 6 \\ 0 & 3 & 4 & 5 & 6 & 7 \\ 3 & 4 & 5 & 0 & 7 & 8 \\ 4 & 5 & 0 & 7 & 8 & 9 \\ 5 & 6 & 7 & 8 & 9 & 0 \\ 6 & 7 & 8 & 9 & 0 & 11 \end{pmatrix}$$

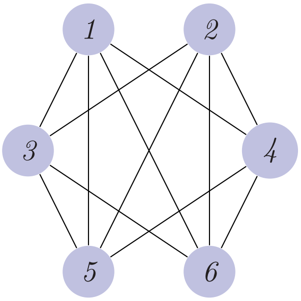


FIGURE 1. A matrix A and its support graph G_A.

when we work with matrix factorization ranks. Recalling the previous section, we can now reformulate the GMP (2.5) into an ideal-sparse form like (2.12), where the V_k's are the triangles in $\mathcal{T}$. We are purposefully withholding details here because the topic will be explored in depth in Part 2 of this thesis.

2.3. Measures, linear functionals, and polynomial optimization

Next, we establish the link between measures and linear functionals acting on polynomials. If not done already the reader is encouraged to peruse the contents of the preceding chapter. Viewing measures via linear functionals will be the key to the moment method, which, in a nutshell, attempts to approach a GMP by a hierarchy of semi-definite programs (SDPs).

Measure-induced linear functional acting on polynomials. For a measure $\mu \in \mathscr{M}(\mathbb{C}^n)$, its *moments* are defined to be the collection of values

$$y_{\alpha,\beta} := \int_{\mathbb{C}^n} \mathbf{x}^\alpha \overline{\mathbf{x}}^\beta d\mu \in \mathbb{C} \text{ for all } \alpha, \beta \in \mathbb{N}^n. \tag{2.13}$$

A given sequence of numbers $(y_{\alpha,\beta})_{\alpha,\beta\in\mathbb{N}^n}$ is said to have a *representing measure* if there exists a measure $\mu \in \mathscr{M}(\mathbb{C}^n)$ such that (2.13) holds. Given a measure μ we can define an associated linear functional $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ in terms of the moments of μ as follows:

$$L(\mathbf{x}^\alpha \overline{\mathbf{x}}^\beta) := \int_{\mathbb{C}^n} \mathbf{x}^\alpha \overline{\mathbf{x}}^\beta d\mu \text{ for all } \alpha, \beta \in \mathbb{N}^n.$$

So, by linearity,

$$L(p) = \int_{\mathbb{C}^n} p d\mu \text{ for any polynomial } p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}].$$

In light of this definition, we are justified in calling $M(L)$ from (1.8) the moment matrix of L. Since each functional $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ is associated with a sequence of numbers $(L(\mathbf{x}^\alpha \overline{\mathbf{x}}^\beta))_{\alpha,\beta\in\mathbb{N}^n}$ we can ask the reverse question: is there a measure μ that induces L? If there is such a measure μ, we call it a representing measure of L.

When looking only at monomials of degree at most $t \in \mathbb{N}$, we call the set of values

$$\int_{\mathbb{C}^n} \mathbf{x}^\alpha \overline{\mathbf{x}}^\beta d\mu \text{ for all } \alpha, \beta \in \mathbb{N}_t^n$$

the moments of μ up to order t. Generally, any proper subset of the full moment set is called a truncated set of moments.

Positivity domain. Given a set of Hermitian polynomials $H \subseteq \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h$ we define the *positivity domain* of the set H to be the set of vectors

$$\mathscr{D}(H) := \{\mathbf{x} \in \mathbb{C}^n : g(\mathbf{x}) \geq 0 \text{ for every } g \in H\}. \tag{2.14}$$

Given a Hermitian polynomial matrix $G \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^{m \times m}$ we define an (infinite) set of Hermitian polynomials

$$H_G := \{\mathbf{v}^* G \mathbf{v} : \mathbf{v} \in \mathbb{C}^d,\ \|\mathbf{v}\| = 1\} \subseteq \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h. \tag{2.15}$$

Thus, it makes sense to refer to the positivity domain of G, denoted by

$$\mathscr{D}(G) := \mathscr{D}(H_G) = \{\mathbf{x} \in \mathbb{C}^n : G(\mathbf{x}) \succeq 0\}. \tag{2.16}$$

Real analog of positivity domains. The complex positivity domain $\mathscr{D}(H)$ has a natural analog, the real positivity domain

$$\mathscr{D}^{\mathbb{R}}(H_{\mathrm{Re}}) := \{(\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}) \in \mathbb{R}^{2n} : g_{\mathrm{Re}}(\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}) \geq 0\ \forall\ g \in H\}.$$

Here, $H_{\mathrm{Re}} \subseteq \mathbb{R}[\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}]$ is the real analog of $H \subseteq \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h$. In view of (1.27) and the complex/real bijection map ϕ from (1.23), we have

$$\mathscr{D}^{\mathbb{R}}(H_{\mathrm{Re}}) = \phi(\mathscr{D}(H)).$$

Real analog of measures support. Given a measure $\mu^{\mathbb{R}}$ on $\mathbb{R}^{2n}$ we define the complex measure μ on $\mathbb{C}^n$ as $\mu = \mu^{\mathbb{R}} \circ \phi$, the *push-forward* of $\mu^{\mathbb{R}}$ by the map ϕ of (1.23). Hence, we have

$$\begin{aligned}\int_{\mathbb{C}^n} p(\mathbf{x}) d\mu &= \int_{\mathbb{R}^{2n}} p \circ \phi^{-1}(\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}) d\mu^{\mathbb{R}} \\ &= \int_{\mathbb{R}^{2n}} p_{\mathrm{Re}}(\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}) d\mu^{\mathbb{R}} + \mathbf{i} \int_{\mathbb{R}^{2n}} p_{\mathrm{Im}}(\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}) d\mu^{\mathbb{R}}\end{aligned} \tag{2.17}$$

for any $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$ (using (1.26)). If $\mu^{\mathbb{R}}$ is supported by the set $\mathscr{D}^{\mathbb{R}}(H_{\mathrm{Re}})$ (i.e., $\mu^{\mathbb{R}}(\mathbb{R}^{2n} \backslash \mathscr{D}^{\mathbb{R}}(H_{\mathrm{Re}})) = 0$), then μ is supported by $\mathscr{D}(H)$ (i.e., $\mu(\mathbb{C}^n \backslash \mathscr{D}(H)) = 0$). This follows from the fact that

$$\phi(\mathbb{C}^n \setminus \mathscr{D}(H)) = \mathbb{R}^{2n} \setminus \mathscr{D}^{\mathbb{R}}(H_{\mathrm{Re}}).$$

Truncated quadratic module. For $t \in \mathbb{N} \cup \{\infty\}$ and $H \subseteq \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h$, the set

$$\mathcal{M}_{2t}(H) := \text{cone}\{gp\overline{p} : p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}],\ g \in H \cup \{1\},\ \deg(gp\overline{p}) \leq 2t\} \tag{2.18}$$

denotes the *quadratic module* generated by H, *truncated at order* $2t$ when $t \in \mathbb{N}$. If $t = \infty$ we simply write $\mathcal{M}(H)$. Note, the definition only makes sense if $2t \geq d_H := \max_{g \in H}\{\deg(g)\}$. The quadratic module $\mathcal{M}(H)$ is said to be *Archimedean* if, for some scalar $R > 0$,

$$R - \sum_{i=1}^{n} x_i \overline{x_i} \in \mathcal{M}(H). \tag{2.19}$$

A polynomial like the one above in (2.19) is called an *algebraic certificate of boundedness* for the associated positivity domain $\mathscr{D}(H)$. We will frequently use the condition that a linear functional $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ is positive on $\mathcal{M}(H)$ for some $H \subseteq \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h$. This is stronger than just saying that L is positive (i.e., $L(p\overline{p}) \geq 0$ for all $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$) as the nonnegativity of L also extends to terms of the form $gp\overline{p}$, i.e. $L(gp\overline{p}) \geq 0$ for all $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$ and $g \in H \cup \{1\}$.

Quadratic modules are vital in modeling positivity constraints in polynomial optimization and for GMPs with supports that involve measures restricted to semi-algebraic domains.

Real analogs of quadratic modules. A set $H \subseteq \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h$ of Hermitian polynomials has a real analog obtained via the map $\text{Re}(\cdot)$ (from (1.27)). Apply $\text{Re}(\cdot)$ element-wise to the set H to get

$$H_{\text{Re}} := \text{Re}(H) = \{p_{\text{Re}} : p \in H\} \subseteq \mathbb{R}[\mathbf{x}_{\text{Re}}, \mathbf{x}_{\text{Im}}]. \tag{2.20}$$

The corresponding real analog of $\mathcal{M}_{2t}(H)$ is denoted by

$$\mathcal{M}^{\mathbb{R}}_{2t}(H_{\text{Re}}) := \text{cone}\{g_{\text{Re}} f^2 : f \in \mathbb{R}[\mathbf{x}_{\text{Re}}, \mathbf{x}_{\text{Im}}]\ ,\ g \in H,\ \ \deg(g_{\text{Re}} f^2) \leq 2t\}.$$

Observe the following correspondences for $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$ and $g \in H$:

$$\begin{aligned} \text{Re}(gp\overline{p}) &= g_{\text{Re}}(p_{\text{Re}}^2 + p_{\text{Im}}^2), \\ gp\overline{p} \in \mathcal{M}_{2t}(H) \iff& g_{\text{Re}}(p_{\text{Re}}^2 + p_{\text{Im}}^2) \in \mathcal{M}^{\mathbb{R}}_{2t}(H_{\text{Re}}), \\ \text{Re}(\mathcal{M}_{2t}(H)) &= \mathcal{M}^{\mathbb{R}}_{2t}(H_{\text{Re}}). \end{aligned} \tag{2.21}$$

Applying the above relations to the Archimedean certificate $R^2 - \mathbf{x}^*\mathbf{x} \in \mathcal{M}(H)$ gives us the next two lemmas.

LEMMA 2.4. *For $H \subseteq \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h$, $\mathcal{M}(H)$ is Archimedean if and only if $\mathcal{M}^{\mathbb{R}}(H_{\text{Re}})$ is Archimedean.*

LEMMA 2.5. *For any $t \in \mathbb{N}$, $H \subseteq \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h_{2t}$, and $f \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h_{2t}$ we have*

$$f \in \mathcal{M}_{2t}(H) \iff f_{\text{Re}} \in \mathcal{M}^{\mathbb{R}}_{2t}(H_{\text{Re}}).$$

The positivity of linear functionals also holds across complex and real analogs.

LEMMA 2.6. *For a Hermitian map $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}] \to \mathbb{C}$, a set $H \subseteq \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h$, and an integer $t \in \mathbb{N} \cup \{\infty\}$, we have*

$$L \geq 0 \text{ on } \mathcal{M}_{2t}(H) \iff L^{\mathbb{R}} \geq 0 \text{ on } \mathcal{M}^{\mathbb{R}}_{2t}(H_{\mathrm{Re}}).$$

PROOF. By the linearity of L and $L^{\mathbb{R}}$, (2.21), and (1.29) (which says $L(gp\overline{p}) = L^{\mathbb{R}}(g_{\mathrm{Re}}(p^2_{\mathrm{Re}} + p^2_{\mathrm{Im}})))$, the result follows. □

COROLLARY 2.7. *Given $H \subseteq \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h$, and a Hermitian map $\mathcal{L} : \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}] \to \mathbb{C}^{m \times m}$, we have*

$$\mathcal{L} \succeq 0 \text{ on } \mathcal{M}(H) \iff \mathcal{L}^{\mathbb{R}} \succeq 0 \text{ on } \mathcal{M}^{\mathbb{R}}(H_{\mathrm{Re}}).$$

Link between complex and real evaluation functionals. The evaluation functional $L_{\mathbf{w}}$ at $\mathbf{w} \in \mathbb{C}^d$ corresponds to the evaluation functional $L_{(\mathbf{w}_{\mathrm{Re}}, \mathbf{w}_{\mathrm{Im}})}$ at $(\mathbf{w}_{\mathrm{Re}}, \mathbf{w}_{\mathrm{Im}}) \in \mathbb{R}^{2d}$ because

$$\begin{aligned} L_{\mathbf{w}}(p) = p(\mathbf{w}, \overline{\mathbf{w}}) &= p_{\mathrm{Re}}(\mathbf{w}_{\mathrm{Re}}, \mathbf{w}_{\mathrm{Im}}) + \mathbf{i} p_{\mathrm{Im}}(\mathbf{w}_{\mathrm{Re}}, \mathbf{w}_{\mathrm{Im}}) \\ &= L^{\mathbb{R}}_{(\mathbf{w}_{\mathrm{Re}}, \mathbf{w}_{\mathrm{Im}})}(p_{\mathrm{Re}}) + \mathbf{i} L^{\mathbb{R}}_{(\mathbf{w}_{\mathrm{Re}}, \mathbf{w}_{\mathrm{Im}})}(p_{\mathrm{Im}}), \end{aligned}$$

for every $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$.

Ideal generated by a set of polynomials. For a set $H \subseteq \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$ of polynomials and an integer $t \in \mathbb{N} \cup \{\infty\}$, one can define the following (truncated) ideal (of order t):

$$I_t(H) := \Big\{ \sum_{h \in H} h p_h : p_h \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}],\ \deg(h p_h) \leq t \Big\}.$$

We consider the *graph-induced ideal* as a specific example. Given an undirected graph $G = (V := [n], E)$, and $\overline{E}$ the associated set of non-edges, we consider the ideal generated by the monomials indexed by the non-edges of G, i.e.,

$$I_E := \Big\{ \sum_{\{i,j\} \in \overline{E}} u_{ij} x_i x_j : u_{ij} \in \mathbb{R}[\mathbf{x}] \Big\} \subseteq \mathbb{R}[\mathbf{x}]. \tag{2.22}$$

Results on measures μ and their induced linear functionals L. We now state several well-known results with regard to the existence of representing measures. The classical results (in a real setting) are presented first, then their complex analogs, followed by the derivation. These results form the cornerstones of the moment method (more on this in Chapter 3). The reader may skip the derivation, which is straightforward but technical.

THEOREM 2.8. *Let $H \subseteq \mathbb{R}[\mathbf{x}]$ be such that $\mathcal{M}^{\mathbb{R}}(H)$ is Archimedean, let $L \in \mathbb{R}[\mathbf{x}]^*$, and assume L is positive on $\mathcal{M}^{\mathbb{R}}(H)$. Then the following holds:*

(i) *(Putinar [**132**]) There exists a measure $\mu^{\mathbb{R}}$ representing L (i.e., $L(p) = \int p d\mu^{\mathbb{R}}$ for all $p \in \mathbb{R}[\mathbf{x}]$) and supported on*

$$\mathscr{D}^{\mathbb{R}}(H) = \{\mathbf{a} \in \mathbb{R}^n : g(\mathbf{a}) \geq 0 \text{ for all } g \in H\}.$$

(ii) *(Tchakaloff [**154**]) For any $k \in \mathbb{N}$, there exists a $\widehat{L} \in \mathbb{R}[\mathbf{x}]^*$ s.t.*

$$\widehat{L}(p) = L(p) \ (p \in \mathbb{R}[\mathbf{x}]_k) \quad \textit{and} \quad \widehat{L} = \sum_{\ell \in [K]} \lambda_\ell L_{\mathbf{a}^{(\ell)}}$$

for $1 \leq K \in \mathbb{N}$, $\lambda_1, ..., \lambda_K > 0$, and atoms $\mathbf{a}^{(1)}, ..., \mathbf{a}^{(K)} \in \mathscr{D}^{\mathbb{R}}(H)$.

The complex analog of Theorem 2.8 reads as follows.

Theorem 2.9. *Let $H \subseteq \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h$ be such that $\mathcal{M}(H)$ is Archimedean, and let $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ be positive on $\mathcal{M}(H)$. Then the following holds:*

(i) *(Based on [**132**]) The functional L has a representing measure μ supported on $\mathscr{D}(H)$.*

(ii) *For any $k \in \mathbb{N}$, there exists a linear functional $\widehat{L} \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ with finite atomic representing measure $\sum_{\ell \in [K]} \lambda_\ell \delta_{\mathbf{v}^{(\ell)}}$ supported by $\mathscr{D}(H)$ such that $\widehat{L}$ coincides with L on $\mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]_k$, i.e.,*

$$\widehat{L}(p) = L(p) \ (p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]_k), \tag{2.23}$$

$$\widehat{L} = \sum_{\ell \in [K]} \lambda_\ell L_{\mathbf{v}^{(\ell)}}, \tag{2.24}$$

for some integer $K \geq 1$, weights $\lambda_1, \lambda_2, ..., \lambda_K > 0$, and atoms $\mathbf{v}^{(1)}$, $\mathbf{v}^{(2)}$,...,$\mathbf{v}^{(K)} \in \mathscr{D}(H)$.

In Chapter 7, we will define a moment hierarchy of lower bounds for the separable rank of a quantum system represented as a complex matrix. We will require Theorem 2.9 to prove that this hierarchy converges.

Adding constraints of the form* $(G \otimes L)(p\overline{p}) \succeq 0$ *to Theorem 2.9. Consider the positivity constraint $(G \otimes L)(p\overline{p}) \succeq 0$ for all $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$, for some Hermitian polynomial matrix $G \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^{m \times m}$. Theorem 2.9 still applies if we additionally impose such constraints on L. Indeed, this is equivalent to replacing H with $H \cup H_G$, where $H_G := \{\mathbf{v}^T G \mathbf{v} \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}] : \mathbf{v} \in \mathbb{C}^n\}$. Thus, the resulting measure μ will be supported on $\mathscr{D}(H \cup H_G) \subseteq \{\mathbf{x} : G(\mathbf{x}, \overline{\mathbf{x}}) \succeq 0\}$.

Deriving Theorem 2.9 (i) from Theorem 2.8 (i). Assume the prerequisites of Theorem 2.9. Consider the set $H_{\mathrm{Re}} \subseteq \mathbb{R}[\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}]$ (defined in (2.20)) and the linear map $L^{\mathbb{R}} : \mathbb{R}[\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}] \to \mathbb{R}$ (defined in (1.28)). By Lemma 2.4, the quadratic module $\mathcal{M}^{\mathbb{R}}(H_{\mathrm{Re}})$ is Archimedean. By Lemma 2.6, $L^{\mathbb{R}} \geq 0$ on $\mathcal{M}^{\mathbb{R}}(H_{\mathrm{Re}})$. Hence, we may apply Theorem 2.8 (i) to H_{Re} and $L^{\mathbb{R}}$ to get a (real) measure $\mu^{\mathbb{R}}$ representing $L^{\mathbb{R}}$ and supported on $\mathscr{D}^{\mathbb{R}}(H_{\mathrm{Re}})$. Consider the (complex) measure μ supported on $\mathscr{D}(H)$ (defined in (2.17)). Observe that the measure μ is a representing measure for L because we have, via (1.30), that

$$L(p) = L^{\mathbb{R}}(p_{\mathrm{Re}}) + \mathbf{i}L^{\mathbb{R}}(p_{\mathrm{Im}}) = \int p_{\mathrm{Re}} d\mu^{\mathbb{R}} + \mathbf{i} \int p_{\mathrm{Im}} d\mu^{\mathbb{R}} = \int p d\mu \ (p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]).$$

This completes the proof of Theorem 2.9 (i). □

Deriving Theorem 2.9 (ii) from Theorem 2.8 (ii). We continue from the above and fix an integer $k \in \mathbb{N}$. By Theorem 2.8 (ii), there exists a real functional $\widehat{L} = \sum_{\ell \in [K]} \lambda_\ell L_{\mathbf{a}^{(\ell)}} \in \mathbb{R}[\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}]^*$, with $K \in \mathbb{N}$, weights $\lambda_\ell > 0$, and atoms $\mathbf{a}^{(\ell)} \in \mathscr{D}^{\mathbb{R}}(H_{\mathrm{Re}})$, such that $\widehat{L}(p) = L^{\mathbb{R}}(p)$ for all $p \in \mathbb{R}[\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}]_k$. Define a complex functional $\widetilde{L} \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ by its action on polynomials $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$ as follows:

$$\widetilde{L}(p) := \widehat{L}(p_{\mathrm{Re}}) + \mathbf{i}\widehat{L}(p_{\mathrm{Im}}).$$

Then, $\widetilde{L}(p) = L(p)$ for every $p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]_k$, by virtue of (1.30). For each atom $\mathbf{a}^{(\ell)}$ ($\ell \in [K]$) define the complex vector $\mathbf{v}^{(\ell)} \in \mathbb{C}^n$ such that $(\mathbf{v}^{(\ell)}_{\mathrm{Re}}, \mathbf{v}^{(\ell)}_{\mathrm{Im}}) = \mathbf{a}^{(\ell)}$. Then, each $\mathbf{v}^\ell$ belongs to $\mathscr{D}(H)$ and $\widetilde{L} = \sum_\ell \lambda_\ell L_{\mathbf{v}^{(\ell)}}$ because

$$\widetilde{L}(p) = \widehat{L}(p_{\mathrm{Re}}) + \mathbf{i}\widehat{L}(p_{\mathrm{Im}}) = \sum_{\ell \in [K]} \lambda_\ell (p_{\mathrm{Re}}(\mathbf{a}^{(\ell)}) + \mathbf{i} p_{\mathrm{Im}}(\mathbf{a}^{(\ell)})) = \sum_{\ell \in [K]} \lambda_\ell p(\mathbf{v}^{(\ell)}).$$

Thus, this concludes the derivation of Theorem 2.9 (ii). □

Next, we now state a closely related classical result due to Putinar [**132**, Theorem 1.2] (see Theorem 2.10 below) and use it to prove its complex analog (see Theorem 2.11 below).

THEOREM 2.10. *Let $f \in \mathbb{R}[\mathbf{x}]$ and $H \subseteq \mathbb{R}[\mathbf{x}]$ such that $\mathcal{M}^{\mathbb{R}}(H)$ is Archimedean. If $f > 0$ on $\mathscr{D}^{\mathbb{R}}(H)$, then $f \in \mathcal{M}^{\mathbb{R}}(H)$.*

THEOREM 2.11. *Let $f \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h$ and $H \subseteq \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h$ such that $\mathcal{M}(H)$ is Archimedean. If $f > 0$ on $\mathscr{D}(H)$, then $f \in \mathcal{M}(H)$.*

PROOF. By the bijective map in (1.27) we have the real analog polynomial $f_{\mathrm{Re}} \in \mathbb{R}[\mathbf{x}_{\mathrm{Re}}, \mathbf{x}_{\mathrm{Im}}]$ of the complex polynomial $f \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^h$, for which it holds that

$$f > 0 \text{ on } \mathscr{D}(H) \implies f_{\mathrm{Re}} > 0 \text{ on } \mathscr{D}^{\mathbb{R}}(H_{\mathrm{Re}}).$$

By Lemma 2.4, we have that $\mathcal{M}^{\mathbb{R}}(H_{\mathrm{Re}})$ is Archimedean. Hence, by Theorem 2.10 we have $f_{\mathrm{Re}} \in \mathcal{M}^{\mathbb{R}}(H_{\mathrm{Re}})$, and thus by Lemma 2.5 we have

$$f \in \mathcal{M}(H).$$

□

CHAPTER 3

Moment hierarchies

We begin this chapter by recalling the classical moment approach that facilitates the building of hierarchies of semidefinite approximations for GMPs like (2.1). Using the results of Section 2.3, we can approach the GMP from the perspective of linear functionals.

The core idea of the moment method is to recast a GMP in terms of a linear functional L and then to impose positivity conditions on L that are necessary for L to have a representing measure. We saw in Theorem 2.9 that a linear functional L has a representing measure μ supported on the semialgebraic set $\mathscr{D}(H)$, provided the associated quadratic module $\mathcal{M}(H)$ is Archimedean and L is positive on $\mathcal{M}(H)$. By relaxing the positivity of L on the quadratic module $\mathcal{M}(H)$ to only positivity on the truncated quadratic module $\mathcal{M}_{2t}(H)$, one obtains lower bounds on the optimal value for the GMP. These relaxations form a hierarchy of semidefinite programs, with each level t in the hierarchy corresponding to a different order of truncation.

Natural questions concerning the bounds of the resulting hierarchy are: how close are the lower bounds to the optimal value of the GMP, under what conditions does a finite level t hierarchy bound coincide to the GMP optimal value, and when/how can one recover optimizers? We review some of the classical results addressing the above questions. For details, see, e.g., the monograph by Lasserre [**106**], or the survey [**42**].

After that, we present the ideal-sparse hierarchy obtained from applying the moment method to the ideal-sparse GMP (2.12). The ideal-sparse hierarchy promises better bounds and (possibly) faster computations than its dense counterpart, assuming sufficient sparsity is present in the GMP.

3.1. The moment method

We now state several widely used definitions and results from polynomial optimization.

3.1.1. Hierarchy of relaxations. Consider the following general complex GMP and its associated semialgebraic domain:

$$\begin{aligned} \mathbf{val} &:= \inf_{\mu \in \mathscr{M}(K)} \left\{ \int f_0 d\mu : \int f_i d\mu = a_i \ (i \in [N_f]) \right\}, \\ K &:= \left\{ \mathbf{x} \in \mathbb{C}^n : g_j(\mathbf{x}, \overline{\mathbf{x}}) \geq 0 \ (j \in [N_g]),\ h_k(\mathbf{x}, \overline{\mathbf{x}}) = 0 \ (k \in [N_h]) \right\}, \end{aligned} \tag{3.1}$$

where $f_0, f_1, ..., f_{N_f}, g_1, ..., g_{N_g}, h_1, ..., h_{N_h} \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$, $f_0, g_1, ..., g_{N_g}$ are Hermitian polynomials, and $a_1, ..., a_{N_f} \in \mathbb{C}$. The associated *moment relaxation of level* $t \in \mathbb{N} \cup \{\infty\}$ is

$$\xi_t := \inf \Big\{ L(f_0) : L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]_{2t}^* \text{ (Hermitian)}, \tag{3.2a}$$

$$L(f_i) = a_i \ (i \in [N_f]), \tag{3.2b}$$

$$L \geq 0 \text{ on } \mathcal{M}_{2t}(\{g_j : j \in [N_g]\}), \tag{3.2c}$$

$$L = 0 \text{ on } I_{2t}(\{h_k : k \in [N_h]\}) \Big\}. \tag{3.2d}$$

We refer to both the sequence of problems and their values $(\xi_t)_{t \in \mathbb{N}}$ as the moment hierarchy associated with GMP (3.1). Observe that the objective in (3.2a) and constraints in (3.2b), (3.2c), and (3.2d) only make sense if

$$\max_{i \in [0, N_f],\ j \in [N_g],\ k \in [N_h]} \{\deg(f_i),\ \deg(g_j),\ \deg(h_k)\} \leq 2t.$$

We will assume this technical condition implicitly holds whenever dealing with hierarchies and not mention it again. Clearly, we have

$$\xi_t \leq \xi_{t+1} \leq \xi_\infty,$$

since any feasible solution L to ξ_∞ (or ξ_{t+1}) induces a feasible solution for ξ_t by restricting L to $\mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]_{2t}^*$.

We now relate (3.2) to (3.1) by showing $\xi_\infty \leq \mathbf{val}$. Assume we are given a measure $\mu \in \mathcal{M}(\mathbb{C}^n)$ feasible for (3.1), let $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ be its induced linear functional. Then, L is feasible for (3.2) with $t = \infty$, and objective value $L(f_0) = \int f_0 d\mu$. We easily verify the three conditions. Firstly, $L(f_i) = \int f_i d\mu = a_i$ for all $i \in [N_f]$. Secondly, $L \geq 0$ on $\mathcal{M}(\{g_j : j \in [N_g]\})$ because any polynomial in $\mathcal{M}(\{g_j : j \in [N_g]\})$ is nonnegative on the set K containing the support of μ. Thirdly, $L = 0$ on the set $I(\{h_k : k \in [N_h]\})$ because every polynomial in $I(\{h_k : k \in [N_h]\})$ vanishes on the support of μ. Thus, ξ_∞ lower bounds $\mathbf{val}$, i.e.,

$$\xi_t \leq \xi_\infty \leq \mathbf{val} \text{ for all } t \in \mathbb{N}.$$

We refer to the above hierarchy of parameters $(\xi_t)_{t\in\mathbb{N}}$ as the *"dense" moment hierarchy.* This is the default hierarchy and should be understood in contradistinction to the "ideal-sparse" hierarchy, which we will introduce below in (3.9).

Moreover, under some mild assumptions (see Theorem 3.3 below), these bounds converge asymptotically to the optimum value **val** of (3.1), i.e.,

$$\lim_{t\to\infty} \xi_t = \xi_\infty = \mathbf{val}.$$

Practically, we can never compute the limit of the hierarchy. We often look for special cases when *finite convergence* occurs, this is when $\xi_r = \xi_\infty$ for some positive integer $r < \infty$ that is hopefully not too large. One such case is when the moment matrix satisfies a "flatness" condition. To this topic, we dedicate the next section, Section 3.2. For now, we continue with the general setting.

Boundedness of functionals. Next, we present a classical lemma that shows that any functional L that is nonnegative on $\mathcal{M}(H)$ is bounded, provided $\mathcal{M}(H)$ satisfies an "Archimedean type" condition (recall definition (2.19)). See, e.g., [**80**, Lemma 13] for a proof.

Lemma 3.1. *Let $H \subseteq \mathbb{C}[\mathbf{x},\overline{\mathbf{x}}]^h$ be such that $R - \sum_{i\in[n]} x_i x_i^* \in \mathcal{M}_2(H)$ for some $R > 0$. For each $t \in \mathbb{N}$, assume $L^{(t)} \in \mathbb{C}[\mathbf{x},\overline{\mathbf{x}}]_{2t}^*$ is nonnegative on $\mathcal{M}_{2t}(H)$. Then*

$$|L^{(t)}(w)| \leq R^t L^{(t)}(1) \text{ for all } w \in [\mathbf{x},\overline{\mathbf{x}}]_{2t}.$$

Moreover, if

$$\sup_{t\in\mathbb{N}} L^{(t)}(1) < \infty, \tag{3.3}$$

then $\{L^{(t)}\}_{t\in\mathbb{N}}$ has a point-wise converging subsequence in $\mathbb{C}[\mathbf{x},\overline{\mathbf{x}}]^*$.

3.1.2. An Archimedean condition for convergent complex hierarchies. We now state and prove a complex variant of a well-known fundamental result (see Theorem 3.3) that characterizes a sufficient condition for asymptotic convergence of the moment bounds. This result will be applied in Chapter 7 to show the convergence of a hierarchy of lower bounds for the separable rank.

Theorem 3.2. *Assume the following three conditions hold:*

(A) *problem (3.1) is feasible,*

(B) *there is an $R > 0$ such that $R - \sum_{i\in[n]} x_i x_i^* \in \mathcal{M}_2(\{g_i : i \in [N_g]\})^1$,*

(C) *there exists $z_i \in \mathbb{C}$ $(i \in [N_f])$ and $c > 0$ such that the polynomial $f := f_0 + \sum_{i\in[N_f]} z_i f_i - c$ is Hermitian and positive on K.*

[1] The asymptotic convergence result of Theorem 3.2 would still hold if we instead assumed that the (untruncated) quadratic module is Archimedean. However, this would require a more involved proof. The result, as stated, is sufficient for our purposes.

Then, for each $t \in \mathbb{N} \cup \{\infty\}$, the program (3.2) attains it optimum, and

$$\lim_{t\to\infty} \xi_t = \xi_\infty = \mathbf{val}.$$

Moreover, the GMP (3.1) has an optimal solution μ that is finite atomic and is supported on K.

Proof. We have shown that $\xi_\infty \leq \mathbf{val}$, so now we show

$$\mathbf{val} \leq \xi_\infty \text{ and } \lim_{t\to\infty} \xi_t = \xi_\infty.$$

($\mathbf{val} \leq \xi_\infty$) By assumption (A), $\mathbf{val} < \infty$, and thus, $\xi_\infty < \infty$. So, assume L is a feasible solution to ξ_∞. Then, $L \geq 0$ on $\mathcal{M}(\{g_j : j \in [N_g]\})$ and $L = 0$ on $\mathcal{M}(\{h_l : l \in [N_h]\})$. Since the quadratic module $\mathcal{M}(\{g_j : j \in [N_g]\})$ is Archimedean, we may now apply Theorem 2.9 (i) to conclude the existence of a representing measure μ for L supported on the set K. By Theorem 2.9 (ii) with

$$k = \max_{i\in[0,N_f],\ j\in[N_g],\ l\in[N_h]} \{\deg(f_i),\ \deg(g_j),\ \deg(h_l)\},$$

we may assume that there exist some $r \in \mathbb{N}$, weights $d_\ell > 0$ ($\ell \in [r]$), and atoms $\mathbf{b}_\ell \in K$ ($\ell \in [r]$) such that

$$L(p) = \sum_{\ell\in[r]} d_\ell L_{\mathbf{b}_\ell}(p) \text{ for all } p \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]_k.$$

Defining the measure $\widetilde{\mu} := \sum_{\ell\in[r]} d_\ell \delta_{\mathbf{b}_\ell}$, we thus have

$$\int f_0 d\widetilde{\mu} = L(f_0) \text{ and } \int f_i d\widetilde{\mu} = a_i \ (i \in [N_f]).$$

Hence, $\widetilde{\mu}$ is a finite atomic measure supported on K that is feasible for the GMP (3.1). This shows $\mathbf{val} \leq \xi_\infty$, and thus we have $\xi_\infty = \mathbf{val}$. Moreover, it shows that the GMP (3.1) has a finite atomic optimal solution supported on K, namely $\widetilde{\mu}$.

(Attainment of optimum) To show that problem (3.2) attains its optimum, we show that it optimizes a linear objective function over a compact set. By assumption (B), there is an $R > 0$ such that

$$R - \sum_{i\in[n]} x_i x_i^* \in \mathcal{M}_2\big(\{g_j : j \in [N_g]\}\big).$$

By assumption (A), $\xi_t \leq \mathbf{val} < \infty$, and hence ξ_t has a feasible solution, for any $t \in \mathbb{N}$. Moreover, we may, without changing the optimal value, restrict the optimization program (3.2) to linear functionals $L^{(t)}$ satisfying

$$L^{(t)}(f_0) \leq \mathbf{val}.$$

Let $L^{(t)}$ be feasible for ξ_t, then $L^{(t)}$ is nonnegative on $\mathcal{M}_{2t}(\{g_i : i \in [N_g]\})$.

Hence, we can apply Lemma 3.1 and conclude that

$$|L^{(t)}(w)| \leq R^t L^{(t)}(1) \quad \text{for any} \quad w \in [\mathbf{x}, \overline{\mathbf{x}}]_{2t}. \tag{3.4}$$

By assumption (C), there exist scalars $z_i \in \mathbb{C}$ $(i \in [N_f])$ and $c > 0$ such that the polynomial

$$f := f_0 + \sum_{i \in [N_f]} z_i f_i - c \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]$$

is Hermitian and $f > 0$ on K. By Theorem 2.11 we then have that f is in the (truncated) quadratic module $\mathcal{M}_{2r_0}(\{g_i : i \in [N_g]\})$ for some $r_0 \in \mathbb{N}$. Thus, for any integer $t \geq r_0$, we have

$$0 \leq L^{(t)}(f) = L^{(t)}(f_0 + \sum_{i \in [N_f]} z_i f_i - c) = L^{(t)}(f_0) + \sum_{i \in [N_f]} z_i a_i - cL^{(t)}(1),$$

and hence $L^{(t)}(1) \leq \Big(\mathbf{val} + \sum_{i \in [N_f]} z_i a_i\Big)/c < \infty$ because $L^{(t)}(f_0) \leq \mathbf{val}$. Thus, we obtain that there exists a constant $C > 0$ such that

$$\sup_{t \in \mathbb{N}} L^{(t)}(1) \leq C < \infty.$$

In combination with (3.4), we concluded, for all $t \in \mathbb{N}$, that

$$|L^{(t)}(w)| \leq R^t C \quad \text{for any} \quad w \in [\mathbf{x}, \overline{\mathbf{x}}]_{2t}.$$

Thus, we have shown that the feasible region of the problem (3.2) is bounded. Hence, we are optimizing a linear objective function over a compact set, and thus the optimum of (3.2) is attained.

$(\lim_{t \to \infty} \xi_t = \xi_\infty)$ We now show asymptotic convergence. For each integer $t \geq 1$, let $L^{(t)}$ be an optimum solution of problem (3.2) (which exists by the above argument). As $\sup_t L^{(t)}(1) \leq C < \infty$, we can use Lemma 3.1 to conclude that there exists an $L \in \mathbb{C}[\mathbf{x}, \overline{\mathbf{x}}]^*$ which is the limit of a subsequence of the sequence $(L^{(t)})_{t \in \mathbb{N}}$. Then, L is feasible for ξ_∞, which implies

$$\xi_\infty \leq L(f_0) = \lim_{t \to \infty} L^{(t)}(f_0) = \lim_{t \to \infty} \xi_t. \qquad \square$$

An Archimedean condition for convergent real hierarchies. We now consider (without proof) the real analog (Theorem 3.3 below) of the preceding result (Theorem 3.2 above). For a detailed exposition on a stronger result, where one assumes only that the associated quadratic module is Archimedean, we refer the reader to [**42, 106**]. Theorem 3.3 will be applied in both Chapter 5 and Chapter 6 to show the convergence of the respective heirarchies.

Consider the general (real) GMP in (2.1) and its associated semialgebraic domain in (2.2):

$$\begin{aligned} \mathbf{val} &:= \inf_{\mu \in \mathscr{M}(K)} \Big\{ \int f_0 d\mu : \int f_i d\mu = a_i \ (i \in [N_f]) \Big\}, \\ K &:= \{\mathbf{x} \in \mathbb{R}^n : g_j(\mathbf{x}) \geq 0 \ (j \in [N_g]), \ h_k(\mathbf{x}) = 0 \ (k \in [N_h])\}, \end{aligned} \tag{3.5}$$

where $f_0, f_1, ..., f_{N_f}, g_1, ..., g_{N_g}, h_1, ..., h_{N_h} \in \mathbb{R}[\mathbf{x}]$ and $a_1, ..., a_{N_f} \in \mathbb{R}$. The associated *moment relaxation of level* $t \in \mathbb{N} \cup \{\infty\}$ is

$$\begin{aligned} \xi_t := \inf \Big\{ L(f_0) : \;& L \in \mathbb{R}[\mathbf{x}]^*_{2t}, && \text{(3.6a)} \\ & L(f_i) = a_i \ (i \in [N_f]), && \text{(3.6b)} \\ & L \geq 0 \text{ on } \mathcal{M}_{2t}(\{g_j : j \in [N_g]\}), && \text{(3.6c)} \\ & L = 0 \text{ on } I_{2t}(\{h_k : k \in [N_h]\}) \Big\}. && \text{(3.6d)} \end{aligned}$$

Theorem 3.3. *Assume the following three conditions hold:*

(A) *problem (3.5) is feasible,*
(B) *there is an $R > 0$ such that $R - \sum_{i \in [n]} x_i^2 \in \mathcal{M}_2^{\mathbb{R}}(\{g_i : i \in [N_g]\})$,*
(C) *there exist $z_i \in \mathbb{R}$ $(i \in [N_f])$ and $c > 0$ such that the polynomial $f := f_0 + \sum_{i \in [N_f]} z_i f_i - c$ is positive on K.*

Then, for each $t \in \mathbb{N} \cup \{\infty\}$, the program (3.6) attains it optimum, and

$$\lim_{t \to \infty} \xi_t = \xi_\infty = \mathbf{val}.$$

Moreover, the GMP (3.5) has an optimal solution μ that is finite atomic and is supported on K.

We continue now in the real setting, as most of our applications in Part 2 are in the real setting. Moreover, the forthcoming sections on ideal sparsity are explained and used in the real setting.

Next, we show that each program (3.6) can be rewritten as an SDP.

3.1.3. SDP formulation of hierarchy.

Writing (3.6c) and (3.6d) as positive semidefinite constraints. The truncated quadratic module constraint (3.6c) and the truncated ideal constraint (3.6d) can both be recast as PSD or linear constraints on related moment matrices. The result is that (3.6) becomes a semidefinite program. Let $L \in \mathbb{R}[\mathbf{x}]^*_{2t}$ be a solution to (3.6), and define $g_0 := 1$. Using (1.11) and (1.12) from Chapter 1 and the linearity of L we get that

$$\begin{aligned} L \geq 0 \text{ on } \mathcal{M}_{2t}(\{g_j : j \in [N_g]\}) &\iff M_{t-d_{g_j}}(g_j L) \succeq 0 \ (j \in [0, N_g]), \\ L = 0 \text{ on } I_{2t}(\{h_k : k \in [N_h]\}) &\iff L(h_k [\mathbf{x}]_{2t-\deg(h_k)}) = 0 \ (k \in [N_h]). \end{aligned}$$

Hence, the parameter ξ_t can be expressed as the optimum value of the semidefinite program

$$\begin{aligned}
\xi_t = \inf\{L(f_0) : \; & L \in \mathbb{R}[\mathbf{x}]_{2t}^*, && (3.7a)\\
& L(f_i) = a_i \; (i \in [N_f]), && (3.7b)\\
& M_{t-d_{g_j}}(g_j L) \succeq 0 \; (j \in [0, N_g]), && (3.7c)\\
& L(h_i[\mathbf{x}]_{2t-\deg(h_k)}) = 0 \; (k \in [N_h])\}. && (3.7d)
\end{aligned}$$

Recall that $d_{g_j} = \lceil \frac{\deg(g_j)}{2} \rceil$ for each $j \in [0, N_g]$. Note that efficient algorithms exist for solving semidefinite programs up to any precision (under some mild assumptions). See, e.g., [**45**] and further references therein.

Polynomial matrix localizing constraints in SDPs. In relation (2.6) of Chapter 2, we saw an example where the semialgebraic set contains a polynomial matrix localizing constraint. To include a general real polynomial matrix localizing constraint $G(\mathbf{x}) \succeq 0$ to (3.7), where $G(\mathbf{x}) \in \mathbb{R}[x]^{m \times m}$, we can naively encode the constraint as follows:

$$L \geq 0 \text{ on } \mathcal{M}_{2t}(\{\mathbf{v}^T G(\mathbf{x})\mathbf{v} : \mathbf{v} \in \mathbb{R}^n\}).$$

However, this would require adding an infinite collection of moment matrices $M_{t-d_G}((\mathbf{v}^T G(\mathbf{x})\mathbf{v})L)$ to the SDP (3.7), which is not implementable in practice; recall $d_G = \max_{i,j \in [m]} \lceil \frac{\deg(G_{ij})}{2} \rceil$. However, as we saw in Corollary 1.6, the constraint

$$M_{t-d_G}(G \otimes L) = L(G \otimes [\mathbf{x}]_{t-d_G}[\mathbf{x}]_{t-d_G}^T) \succeq 0,$$

is both stronger and better suited to numerical implementation. This introduces a new PSD constraint involving a matrix of size $\binom{n+t-d_G}{t-d_G} \cdot m$, which costs more memory in terms of hardware, but is computationally feasible for moderate values of m.

It should be noted that convergence of the moment hierarchy for polynomial matrix inequality optimization problems was first studied by Herion and Lasserre in 2005 [**41**]. In terms of software, YALMIP [**90**] (a MATLAB [**89**] add-on) does provide support for these constraints.

Speaking of computational costs brings us to the next topic, the problem of growing (with the level t) matrix sizes in SDPs coming from the moment method.

Exponential growth of matrices in SDP hierarchies. The critical weakness of the (dense) hierarchy (3.7) is that it involves the matrices $M_{t-d_{g_i}}(g_i L)$ whose sizes rapidly grow beyond the memory capacity of most computers. Indeed, consider the moment matrix $M_t(g_0 L) = M_t(L)$ which is of size $\binom{n+t}{t}$. Alternatively, consider $M_{t-d_G}(G \otimes L)$, which is of size $\binom{n+t-d_G}{t-d_G} \cdot m$. Matrix size is often the bottleneck in numerical experimentation as most computers cannot meet the exponentially growing memory demands as the level t of the hierarchy increases. As a consequence, only modestly sized GMP instances are solved in practice. However, research into reducing the size of SDPs has been spurred on by the successful industrial applications of SDPs in recent years [**106**]. Ideal sparsity, which we introduced in Section 2.2, and will expand on shortly, is one such promising research direction. We now discuss some of the other classical techniques used for reducing the size of moment matrices in the SDP (3.7).

Existing schemes to improve the scalability of moment relaxations. Several schemes have been developed to overcome the scalability issue of the dense hierarchy (3.7). Without compromising the convergence guarantees, they aim to reduce the involved matrices' size by exploiting the input polynomials' specific structure. Usually, the new hierarchy is faster to compute but with weaker bounds (except in the case of ideal sparsity). We now discuss three such paradigms.

The first way is to use the properties of the input polynomials to define hierarchies involving smaller moment matrices. There are three situations where this is applicable.

- *Correlative sparsity* occurs when there are few correlations between the variables of the input polynomials, thereby allowing one to treat them somewhat independently and ignore interactions [**159, 104**]. Correlative sparsity has been extended to derive moment relaxations of polynomial problems in complex variables [**91**], noncommutative variables [**96**] and polynomial matrix inequalities [**172**]. We discuss the correlative sparsity approach for GMPs later in Section 3.1.5 and how it relates to ideal sparsity.
- *Term sparsity* occurs when there are few (in comparison to all possible) monomial terms involved in the input polynomials. For unconstrained polynomial optimization, one well-known solution is to eliminate the monomial terms which never appear among the support of sums of squares decompositions [**136**]. Term sparsity has recently been the focus of active research with extensions to constrained polynomial optimization [**161, 162**]. Note that term and correlative sparsity can be combined [**163**]. We refer to the recent surveys [**116, 173**] for a general exposition on sparse polynomial optimization.
- *Ideal sparsity* is our new contribution to the scalability of moment hierarchies, based on our work in [**100**]. We present the topic in detail in Sections 2.2.1, 3.1.4, and 3.2.3. The crux of ideal sparsity is that

one can reduce the size of the moment matrices in an SDP by observing that many entries are set to zero due to ideal constraints. In a similar vein, we consider a " block-diagonal reduction" in Section 7.2.

Secondly, one can sometimes decompose the input polynomials into a special structure. One can decompose a polynomial into a *sum of nonnegative circuits* by solving a geometric programming relaxation [**88**], or a second-order cone programming relaxation [**8, 160**]. Or, one can decompose it into a sum of arithmetic-geometric-mean-exponentials [**30**] with relative entropy programming relaxations.

The third approach is to exploit symmetries in the moment matrices [**138**], provided each input polynomial is invariant under the action of some subgroup of the general linear group.

3.1.4. Ideal-sparse moment hierarchy of relaxations. For the particular ideal-sparse GMP in (2.12) with semialgebraic domain K first described in (2.7), we can define a modified associated hierarchy. For convenience, we recall the GMP and its associated semialgebraic domain here:

$$\mathbf{val}^{\text{isp}} := \inf_{\substack{\mu_k \in \mathscr{M}(K_k)\\ k\in[p]}} \Big\{ \sum_{k\in[p]} \int f_{0|V_k} d\mu_k : \sum_{k\in[p]} \int f_{i|V_k} d\mu_k = a_i \ (i \in [N_f]) \Big\}, \tag{3.8a}$$

$$K := \{\mathbf{x} \in \mathbb{R}^n : g_j(\mathbf{x}) \geq 0 \ (j \in [N_g]),\ \mathbf{x}^S := \prod_{i\in S} x_i = 0 \ (S \in \mathcal{S})\}, \tag{3.8b}$$

$$K_k := \{\mathbf{y} \in \mathbb{R}^{|V_k|} : (\mathbf{y}, 0_{V\setminus V_k}) \in K\} \subseteq \mathbb{R}^{|V_k|} \ (k \in [p]). \tag{3.8c}$$

Here, $f_0, f_1, ..., f_{N_f}, g_1, ..., g_{N_g}, h_1, ..., h_{N_h} \in \mathbb{R}[\mathbf{x}]$, $a_1, ..., a_{N_f} \in \mathbb{R}$, $V_1, ..., V_p$ are $\subseteq$-maximal subsets of $[n]$ not containing any $S \in \mathcal{S} \subseteq \mathcal{P}([n])$. We consider the following parameters:

$$\begin{aligned} \xi_t^{\text{isp}} := \inf \Big\{ \textstyle\sum_{k\in[p]} \quad & L_k(f_{0|V_k}) : \\ & L_k \in \mathbb{R}[\mathbf{x}(V_k)]_{2t}^* \ (k \in [p]), \\ & \textstyle\sum_{k\in[p]} L_k(f_{i|V_k}) = a_i \ (i \in [N_f]), \\ & L_k \geq 0 \text{ on } \mathcal{M}_{2t}(\{g_{i|V_k} : i \in [N_g]\}) \ (k \in [p]) \Big\}, \end{aligned} \tag{3.9}$$

that we call the *ideal-sparse* hierarchy of moment approximations for problem (3.8a). Observe that there are no ideal constraints, as they have been encoded in the supports $V_1, ..., V_p$. Just as with the dense hierarchy in (3.6), there is an SDP formulation

$$\begin{aligned} \xi_t^{\text{isp}} = \inf \Big\{ \textstyle\sum_{k\in[p]} \quad & L_k(f_{0|V_k}) : \\ & L_k \in \mathbb{R}[\mathbf{x}(V_k)]_{2t}^* \ (k \in [p]), \\ & \textstyle\sum_{k\in[p]} L_k(f_{i|V_k}) = a_i \ (i \in [N_f]), \\ & M_{t-d_{g_{i|V_k}}}(g_{i|V_k} L_k) \succeq 0 \ (i \in [0, N_g],\ k \in [p]) \Big\}. \end{aligned} \tag{3.10}$$

If there is a matrix constraint of the form $G(\mathbf{x}) \succeq 0$ in the definition of K, then we add the constraints $M_{t-d_{G_{|V_k}}}(G_{|V_k} \otimes L_k) \succeq 0$ for all $k \in [p]$ to (3.10).

We next show that the ideal-sparse hierarchy $(\xi_t^{\text{isp}})_{t\in\mathbb{N}}$ provides bounds for **val** that are at least at good as the bounds $(\xi_t)_{t\in\mathbb{N}}$ from (3.6).

Theorem 3.4. *For any integer $t \geq 1$ we have*

$$\xi_t \leq \xi_t^{\text{isp}} \leq \mathbf{val}.$$

If in addition the assumptions of Theorem 3.3 holds, then

$$\lim_{t\to\infty} \xi_t^{\text{isp}} = \mathbf{val}.$$

Proof. By construction, $\xi_t^{\text{isp}} \leq \mathbf{val}^{\text{isp}}$, which, combined with Proposition 2.2, gives $\xi_t^{\text{isp}} \leq \mathbf{val}$. We now show $\xi_t \leq \xi_t^{\text{isp}}$. For this, assume $(L_1, ..., L_p)$ is feasible for (3.9). Define $L \in \mathbb{R}[\mathbf{x}]_{2t}^*$ by setting $L(p) = \sum_{k\in[p]} L_k(p_{|V_k})$ for any $p \in \mathbb{R}[\mathbf{x}]_{2t}$. By construction, $L(f_i) = \sum_k L_k(f_{i|V_k})$ for $i \in [0, N_f]$, so that $L(f_i) = a_i$ for $i \in [N_f]$, and $L \geq 0$ on $\mathcal{M}_{2t}(\{g_i : i \in [N_g]\})$. For each $S \in \mathcal{S}$ and $k \in [p]$, we have $S \not\subseteq V_k$ (because of how V_k was defined) and thus ${\mathbf{x}^S}_{|V_k}$ is identically zero; hence, for any $u \in \mathbb{R}[\mathbf{x}]_{2t-2}$, we have $L(u\mathbf{x}^S) = \sum_k L_k(u_{|V_k}{\mathbf{x}^S}_{|V_k}) = 0$. Hence, L is feasible for (3.6) with the same objective value as $(L_1, ..., L_p)$, which shows $\xi_t \leq \xi_t^{\text{isp}}$.

The asymptotic convergence of ξ_t^{isp} to **val** follows from the just proven fact that $\xi_t \leq \xi_t^{\text{isp}}$ and from Theorem 3.3, which implies $\lim_{t\to\infty} \xi_t = \mathbf{val}$ under the above-stated assumptions. □

Ideal sparsity shrinks matrices in SDP hierarchy. The whole appeal of ideal sparsity rests on the fact that the largest matrix size in (3.10) is now

$$\max_{k\in[p]} \binom{|V_k| + t}{t},$$

or $\max_{k\in[p]} \binom{|V_k|+t-d_G}{t-d_G} \cdot m$ if there are $m \times m$-sized matrix polynomial constraints $G(\mathbf{x}) \succeq 0$. Hence, the hope in applying ideal sparsity is that the quantity $\max_{k\in[p]} |V_k|$ is much smaller than n. Later in this section, we will elaborate more on the computational trade-off between a few large matrices vs. many smaller matrices in SDPs. For now, it suffices to say that most commercial and academic SDP solvers handle the latter situation better than the former. However, an excess of smaller matrices (i.e., large p) will still lead to most computers running out of memory. In anticipation of this shortfall, we propose to merge some of the sets $V_1, ..., V_p$, which we elaborate on in the next paragraph.

Merge maximal sets. Let $\widetilde{p} \leq p$, and $\widetilde{V}_1, ..., \widetilde{V}_{\widetilde{p}} \subseteq [n]$ denote sets such that every set V_k is contained in $V_{\widetilde{k}}$ for some $\widetilde{k} \in [\widetilde{p}]$. One can define the corresponding ideal-sparse moment hierarchy of bounds $\widetilde{\xi}_t^{\mathrm{isp}}$, which involves $\widetilde{p} \leq p$ measure variables, each supported on a set in $\widetilde{V}_1, ..., \widetilde{V}_{\widetilde{p}}$ (instead of the sets $V_1, ..., V_p$). However, these sets $\widetilde{V}_k$ may now contain some of the forbidden sets $S \in \mathcal{S}$ (from (2.7)) defining the ideal constraints. Hence, we must reimpose the ideal constraints to get a comparable hierarchy

$$\widetilde{\xi}_t^{\mathrm{isp}} := \inf \Big\{ \begin{array}{l} \sum_{h=1}^{\widetilde{p}} \widetilde{L}_h(f_{0|\widetilde{V}_h}) : \\ \widetilde{L}_h \in \mathbb{R}[\mathbf{x}(\widetilde{V}_h)]_{2t}^* \ (h \in [\widetilde{p}]), \\ \sum_{h=1}^{\widetilde{p}} \widetilde{L}_h(f_{i|\widetilde{V}_h}) = a_i \ (i \in [N_f]), \\ \widetilde{L}_h \geq 0 \text{ on } \mathcal{M}_{2t}(\{g_{i|\widetilde{V_h}} : i \in [N_g]\}) \ (h \in [\widetilde{p}]), \\ \widetilde{L}_h(\mathbf{x}^S \mathbf{x}^\alpha) = 0 \ (\mathrm{supp}(\alpha) \subseteq \widetilde{V}_h, \ S \subseteq \widetilde{V}_h, \ S \in \mathcal{S}) \Big\}. \end{array} \tag{3.11}$$

Note that this parameter interpolates between the dense and sparse parameters: indeed, $\widetilde{\xi}_t^{\mathrm{isp}} = \xi_t^{\mathrm{isp}}$ if $\widetilde{V}_1 = V_1, ..., \widetilde{V}_{\widetilde{p}} = V_p$, and $\widetilde{\xi}_t^{\mathrm{isp}} = \xi_t$ if $\widetilde{p} = 1$. Accordingly, we have the following inequalities among the parameters.

LEMMA 3.5. *Assume that $\widetilde{p} \leq p$ and that the sets $\widetilde{V}_1, ..., \widetilde{V}_{\widetilde{p}}$ contain the sets $V_1, ..., V_p$ (in the sense that, for each $k \in [p]$, $V_k \subseteq V_h$ for some $h \in [\widetilde{p}]$). Then*

$$\xi_t \leq \widetilde{\xi}_t^{\mathrm{isp}} \leq \xi_t^{\mathrm{isp}} \quad \text{for all } t \in \mathbb{N} \cap \{\infty\}.$$

PROOF. The proof for the inequality $\xi_t \leq \widetilde{\xi}_t^{\mathrm{isp}}$ is analogous to the proof of $\xi_t \leq \xi_t^{\mathrm{isp}}$ in Theorem 3.4. We now show $\widetilde{\xi}_t^{\mathrm{isp}} \leq \xi_t^{\mathrm{isp}}$. For this, assume $(L_1, ..., L_p)$ is feasible for the parameter ξ_t^{isp}. As each set V_k is contained in some set $\widetilde{V}_h$, there exists a partition $[p] = A_1 \cup ... \cup A_{\widetilde{p}}$ such that $V_k \subseteq \widetilde{V}_h$ for all $k \in A_h$ and $h \in [\widetilde{p}]$. For $h \in [\widetilde{p}]$, we define $\widetilde{L}_h \in \mathbb{R}[\mathbf{x}(\widetilde{V}_h)]_{2t}^*$ by setting $\widetilde{L}_h(q) = \sum_{k \in A_h} L_k(q_{|V_k})$ for $q \in \mathbb{R}[\mathbf{x}(\widetilde{V}_h)]_{2t}$. Then, one can easily verify that $(\widetilde{L}_1, ..., \widetilde{L}_{\widetilde{p}})$ provides a feasible solution for $\widetilde{\xi}_t^{\mathrm{isp}}$, with the same objective value as $(L_1, ..., L_p)$.

Now, we check the ideal constraints. Assume $S \cup \mathrm{supp}(\alpha) \subseteq \widetilde{V}_h$ and $S \in \mathcal{S}$. Then, as S is not contained in any maximal set V_k, we have $L_k((\mathbf{x}^S \mathbf{x}^\alpha)_{|V_k}) = 0$ for all $k \in [p]$, which directly implies $\widetilde{L}_h(\mathbf{x}^S \mathbf{x}^\alpha) = 0$. □

3.1.5. Links between ideal sparsity and correlative sparsity. Assume K is as defined in (3.8b) with $V_1, ..., V_p$ denoting all the $\subseteq$-maximal subsets of $[n]$ not containing any set $S \in \mathcal{S} \subseteq \mathcal{P}([n])$. These sets $V_1, ..., V_p$ now induce the graph

$$G = \Big(V = [n], \ E := \big\{\{i, j\} : \{i, j\} \subseteq V_k \text{ for some } k \in [p]\big\}\Big).$$

So, by construction, the maximal cliques of G are the sets $V_1, .., V_p$.

Consider a *chordal extension* $\widehat{G} = (V, \widehat{E})$ of G, i.e., such that $E \subseteq \widehat{E}$. Let $\widehat{V}_1, ..., \widehat{V}_{\widehat{p}}$ denote the maximal cliques of $\widehat{G}$. Notably, chordal graphs have at most n distinct maximal cliques, so $\widehat{p} \leq n$. Furthermore, a graph is chordal if and only if its maximal cliques satisfy the so-called *running intersection property* (RIP). See, e.g., [**54**] for details. Hence, the maximal cliques $\widehat{V}_1, ..., \widehat{V}_{\widehat{p}}$ satisfy (possibly after reordering) the RIP:

$$\forall\ k \in \{2, ..., \widehat{p}\}\ \exists\ j \in \{1, ..., k-1\} \text{ s.t. } \widehat{V}_k \cap (\widehat{V}_1 \cup ... \cup \widehat{V}_{k-1}) \subseteq \widehat{V}_j. \tag{3.12}$$

We proceed now in two steps.

Step one is to partition the index set $\mathbb{N}^n_t$ of the moment matrix $L([\mathbf{x}]_t[\mathbf{x}]_t^T)$ into $\mathcal{I}_t$ and $\mathbb{N}^n_t \setminus \mathcal{I}_t$ (we define $\mathcal{I}_t$ below in (3.14)). The set $\mathcal{I}_t$ is further partitioned $\mathcal{I}_t = \cup_{k \in [\widehat{p}]} \mathcal{I}_{k,t}$ with the special property (see Lemma 3.6) that

$$L(\mathbf{x}^\alpha \mathbf{x}^\beta) = 0\ (\{\alpha, \beta\} \not\subseteq \mathcal{I}_{k,t} \text{ for all } k \in [\widehat{p}]).$$

With respect to these partitions, we show that the moment matrix $L([\mathbf{x}]_t[\mathbf{x}]_t^T)$ has an "overlapping block-diagonal structure", with the blocks given by the sets $\mathcal{I}_{k,t}$ $(k \in [\widehat{p}])$.

Step two is to show that the support graph of the principal submatrix $L([\mathbf{x}]_t[\mathbf{x}]_t^T)[\mathcal{I}_t]$ is chordal (see Lemma 3.7). This allows us to use a known result (see Theorem 3.8) for characterizing the PSDness of $L([\mathbf{x}]_t[\mathbf{x}]_t^T)[\mathcal{I}_t]$ in terms of the PSDness of several smaller matrices.

Making the moment matrix "overlapping block-diagonal". Begin by recalling the definition of the hierarchy, now with ideal constraints of a special form:

$$\begin{aligned} \xi_t = \inf\{L(f_0) :\, & L \in \mathbb{R}[\mathbf{x}]^*_{2t}, \\ & L(f_i) = a_i\ (i \in [N_f]), \\ & L([\mathbf{x}]_t[\mathbf{x}]_t^T) \succeq 0, \\ & L(g_j[\mathbf{x}]_{t-d_j}[\mathbf{x}]_{t-d_j}^T) \succeq 0\ (j \in [N_g]), \\ & L(\mathbf{x}^S[\mathbf{x}]_{2t-|S|}) = 0\ (S \in \mathcal{S})\}. \end{aligned} \tag{3.13a}$$

Fix $t \in \mathbb{N}$, and define the sets

$$\begin{aligned} \mathcal{I}_t &= \bigcup_{k \in [\widehat{p}]} \mathcal{I}_{k,t} \subseteq \mathbb{N}^n_t, \\ \mathcal{I}_{k,t} &:= \left\{\alpha \in \mathbb{N}^n_t : \operatorname{supp}(\alpha) \subseteq \widehat{V}_k\right\}\ (k \in [\widehat{p}]). \end{aligned} \tag{3.14}$$

Lemma 3.6. *Assume $L \in \mathbb{R}[\mathbf{x}]^*_{2t}$ and L satisfies (3.13a), then*

$$L(\mathbf{x}^\alpha \mathbf{x}^\beta) = 0\ (\{\alpha, \beta\} \not\subseteq \mathcal{I}_{k,t} \textit{ for all } k \in [\widehat{p}]).$$

Proof. Assume there is no index $k \in [\widehat{p}]$ such that $\{\alpha, \beta\} \subseteq \mathcal{I}_{k,t}$. Then, $\operatorname{supp}(\alpha + \beta)$ is not in any set $\widehat{V}_k$ $(k \in [p])$, else $\operatorname{supp}(\alpha), \operatorname{supp}(\beta) \subseteq \widehat{V}_k$ and thus

$\alpha, \beta \in \mathcal{I}_{k,t}$, yielding a contradiction. As $\mathrm{supp}(\alpha+\beta)$ is not contained in any $\widehat{V}_k$ ($k \in [p]$) it must contain an $S \in \mathcal{S}$, and thus $L(\mathbf{x}^\alpha \mathbf{x}^\beta) = 0$. □

This result, in particular, implies that one may restrict the matrix $L([\mathbf{x}]_t[\mathbf{x}]_t^T)$ in (3.13) to its principal submatrix $L([\mathbf{x}]_t[\mathbf{x}]_t^T)[\mathcal{I}_t]$ indexed by $\mathcal{I}_t$ since any row/column indexed by $\alpha \in \mathbb{N}_t^n \setminus \mathcal{I}_t$ is identically zero.

The support graph of $L([\mathbf{x}]_t[\mathbf{x}]_t^T)[\mathcal{I}_t]$ ***is chordal.*** We now show that the support graph of the matrix $L([\mathbf{x}]_t[\mathbf{x}]_t^T)[\mathcal{I}_t]$ is chordal by showing that the sets $\mathcal{I}_{1,t}, ..., \mathcal{I}_{\widehat{p},t}$ are its maximal cliques, and that they inherit the RIP property. The sets $\mathcal{I}_{1,t}, \mathcal{I}_{2,t}, ..., \mathcal{I}_{\widehat{p},t}$ serve an analogous role to the sets $\widehat{V}_1, ..., \widehat{V}_{\widehat{p}}$.

By Lemma 3.6, $L(\mathbf{x}^\alpha \mathbf{x}^\beta) \neq 0$ implies $\{\alpha, \beta\} \subseteq \mathcal{I}_{k,t}$ for some $k \in [\widehat{p}]$. In other words, the support graph of the matrix $L([\mathbf{x}]_t[\mathbf{x}]_t^T)$ is contained in the graph with vertex set $\mathcal{I}_t$, whose maximal cliques are the sets $\mathcal{I}_{1,t}, ..., \mathcal{I}_{\widehat{p},t}$.

LEMMA 3.7. *The sets $\mathcal{I}_{1,t}, ..., \mathcal{I}_{\widehat{p},t}$ satisfy the RIP property:*

$$\forall q \in \{2, ..., \widehat{p}\} \exists\ k \in \{1, ..., q-1\} \text{ s.t. } \mathcal{I}_{q,t} \cap (\mathcal{I}_{1,t} \cup ... \cup \mathcal{I}_{q-1,t}) \subseteq \mathcal{I}_{k,t}. \quad (3.15)$$

PROOF. Let $q \in \{2, \dots, \widehat{p}\}$ and assume by way of contradiction that there exists no $k \in [q-1]$ for which $\mathcal{I}_{q,t} \cap (\mathcal{I}_{1,t} \cup \ldots \cup \mathcal{I}_{q-1,t}) \subseteq \mathcal{I}_{k,t}$ holds. Then, for each $k \in [q-1]$, there exists $\alpha^k \in \mathcal{I}_{q,t} \cap (\mathcal{I}_{1,t} \cup ... \cup \mathcal{I}_{q-1,t}) \setminus \mathcal{I}_{k,t}$ and thus there exists $i_k \in V \setminus \widehat{V}_k$ such that $\alpha^k_{i_k} \geq 1$. As $\alpha^k \in \mathcal{I}_{q,t}$ and $\alpha^k_{i_k} \geq 1$ it follows that $i_k \in \widehat{V}_q$. In addition, $\alpha^k \in \mathcal{I}_{j,t}$ for some $j \in [q-1]$. Again, as $\alpha^k_{i_k} \geq 1$ it follows that $i_k \in \widehat{V}_j$. This shows that

$$i_k \in \widehat{V}_q \cap (\widehat{V}_1 \cup ... \cup \widehat{V}_{q-1}) \quad \text{for all } k \in [q-1].$$

By the RIP property (3.12) for $\widehat{V}_1, ..., \widehat{V}_p$, there exists $q_0 \in [q-1]$ such that $\widehat{V}_q \cap (\widehat{V}_1 \cup ... \cup \widehat{V}_{q-1}) \subseteq \widehat{V}_{q_0}$. Therefore, $i_k \in \widehat{V}_{q_0}$ for all $k \in [q-1]$. As $i_k \notin \widehat{V}_k$ this implies that $q_0 \neq k$ for all $k \in [q-1]$, and thus we have contradicted the RIP of the sets $\widehat{V}_1, ..., \widehat{V}_p$. □

The above extends easily to the localizing matrices $L(g_j[\mathbf{x}]_{t-d_j}[\mathbf{x}]_{t-d_j}^T)$ for $j \in [N_g]$. In the same way, one may restrict the matrix $L(g_j[\mathbf{x}]_{t-d_j}[\mathbf{x}]_{t-d_j}^T)$ to its principal submatrix indexed by $\mathcal{I}_{t-d_j}$ and its support graph is contained in the graph with vertex set $\mathcal{I}_{t-d_j}$, whose maximal cliques are the sets $\mathcal{I}_{1,t-d_j}, ..., \mathcal{I}_{\widehat{p},t-d_j}$. Moreover, there is a correlative sparsity pattern on the matrix $L(g_j[\mathbf{x}]_{t-d_j}[\mathbf{x}]_{t-d_j}^T)$ ($0 \leq j \leq m$), which is inherited from the chordal structure of $\widehat{G}$.

Matrices with chordal support graphs. We can now invoke a classical result that relates a chordal PSD matrix (matrix with a chordal support graph) to the PSDness of the sub-matrices induced by the cliques of the support graph.

Theorem 3.8 ([**3**]). *Consider a positive semidefinite matrix* $X \in \mathcal{S}^n_+$ *whose support graph is contained in a chordal graph* $\widehat{G}$*, with maximal cliques* $\widehat{V}_1, ..., \widehat{V}_{\widehat{p}}$*. Then, there exist positive semidefinite matrices* $Y_k \in \mathcal{S}^{\widehat{V}_k}_+$ *(*$k \in [\widehat{p}]$*) such that* $X = \sum_{k=1}^{\widehat{p}} Z_k$*, where* $Z_k = Y_k \oplus 0_{V\setminus\widehat{V}_k, V\setminus\widehat{V}_k} \in \mathcal{S}^n_+$ *is obtained by padding* Y_k *with zeros.*

Therefore one may apply Theorem 3.8 to get a more economical reformulation of ξ_t. Indeed, by Theorem 3.8, one may write

$$L(g_j[\mathbf{x}]_{t-d_j}[\mathbf{x}]^T_{t-d_j}) = \sum_{k\in[\widehat{p}]} Z_{j,k},$$

where $Z_{j,k}$ is obtained from a matrix indexed by the set $\mathcal{I}_{k,t-d_j}$ by padding it with zero entries and replace the condition $L(g_j[\mathbf{x}]_{t-d_j}[\mathbf{x}]^T_{t-d_j}) \succeq 0$ by the conditions $Z_{j,1}, ..., Z_{j,\widehat{p}} \succeq 0$. The advantage is that requiring $Z_{j,k} \succeq 0$ boils down to checking positive semidefiniteness of a potentially much smaller matrix, indexed by $\mathcal{I}_{k,t-d_j}$.

Hence, this allows one to replace a single large positive semidefinite matrix with several smaller positive semidefinite matrices. While this method offers a more economical way of computing the dense parameter ξ_t, it is nevertheless inferior to the ideal-sparse approach described in the previous section.

As a final observation, another possibility to exploit the above correlative sparsity structure would be to replace in the definition of ξ_t in the program (3.6) each condition $L(g_j[x]_{t-d_j}[x]_{t-d_j}) \succeq 0$ by $\widehat{p}$ smaller matrix conditions $L(g_{j|\widehat{V}_k}[x(\widehat{V}_k)]_{t-d_j}[x(\widehat{V}_k)]_{t-d_j}) \succeq 0$ for $k \in [\widehat{p}]$. In other words, if $L_{|V_k}$ denotes the restriction of L to the polynomials in variables indexed by $\widehat{V}_k$, then we replace the condition $L \geq 0$ on $\mathcal{M}_{2t}(\mathbf{g})$ by the conditions $L_{|\widehat{V}_k} \geq 0$ on $\mathcal{M}_{2t}(\mathbf{g}_{|\widehat{V}_k})$ for each $k \in [\widehat{p}]$. In this way we obtain another parameter, denoted by ξ^{csp}_t, that is weaker than ξ_t and thus satisfies

$$\xi^{\text{csp}}_t \leq \xi_t \leq \widetilde{\xi}^{\text{isp}}_t \leq \xi^{\text{isp}}_t.$$

Recall $\widetilde{\xi}^{\text{isp}}_t$ is the parameter from (3.11) obtained when selecting an extension $\widetilde{G}$ of G, including, for instance, selecting a chordal extension $\widetilde{G} = \widehat{G}$.

3.2. Flatness and extraction of optimal solutions

This section deals with two intertwined topics: identifying if the moment hierarchy has converged to the optimal value of the original GMP at a finite level t and recovering optimizers when this occurs.

In Section 3.2.1, we present a classical result, Theorem 3.9, by Curto and Fialkow [**39**] that states a sufficient condition (on the ranks of successively bigger leading principal submatrices of the moment matrix of the hierarchy (3.5) at some fixed level) for finite convergence. Assuming that we are in the setting of Theorem 3.9, there is an algorithm, first proposed by Henrion and Lasserre [**85**], for extracting the optimizers of the GMP (3.5). We present this algorithm in Section 3.2.2. Finally, in Section 3.2.3, we say a few words relating the preceding two well-known topics to the newer topic of ideal sparsity and its application to matrix factorization ranks.

3.2.1. The flatness condition.

Recall the GMP in (3.5), its associated semialgebraic set K, and its moment hierarchy of approximating SDPs (3.7). By Theorem 3.3, if the quadratic module $\mathcal{M}(\mathbf{g})$ is Archimedean, then the bounds ξ_t converge asymptotically to ξ_∞. If, additionally, the conditions of Theorem 3.3 hold, then $\xi_\infty = \mathbf{val}$ and problem (3.5) has a finite atomic optimal solution μ. Now we show conditions under which the convergence is finite, and the resulting optimal (pseudo) measure is finite atomic.

Define the degree of the semialgebraic set K as follows:

$$d_K := \max\{\deg(f_i), \lceil \frac{\deg(g_j)}{2} \rceil, \deg(h_k) : i \in [0, N_f],\ j \in [0, N_g],\ k \in [N_h]\}.$$

THEOREM 3.9. [**39, 40**] *Let $t \in \mathbb{N}$ be such that $t \geq d_K$. Assume $L \in \mathbb{R}[\mathbf{x}]^*_{2t}$ is an optimal solution to program (3.7) and that it satisfies the following* flatness condition*:*

$$r := \text{rank } L([x]_s[x]_s^T) = \text{rank } L([x]_{s-d_K}[x]_{s-d_K}^T) \quad \textit{for some } s \in [d_K, t]. \quad (3.16)$$

Then, $\xi_t = \mathbf{val}$, and the GMP (3.5) has a finite atomic optimal solution μ supported by r points in K, i.e.,

$$\mu = \sum_{\ell \in [r]} c_\ell \delta_{\mathbf{x}^{(\ell)}},$$

for some weights $c_1, c_2, ..., c_r > 0$ and atoms $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, ..., \mathbf{x}^{(r)} \in K$.

To paraphrase, Theorem 3.9 says that if the level t is high enough for (3.7) to capture the "data" of the problem and subsequent leading principal submatrices of the moment matrix "do not exhibit an increase in information," i.e., (3.16) holds, then the hierarchy $(\xi_t)_{t\in\mathbb{N}}$ has converged finitely at level t. Furthermore, the r atoms of the finite atomic representing measure μ are often of great interest and have special application-specific interpretations associated with them.

Note that numerical noise and error can result in an incorrectly computed rank for the moment matrix $L([x]_s[x]_s^T)$. As such, verifying if flatness holds for a given problem is not always straightforward. However, most software options we mention later account for this with rigorous checks.

We saw in Section 2.1.1 that a polynomial optimization problem like (2.3) can be equivalently reformulated as a GMP. In this setting, the recovered atoms are optimizers of the original polynomial optimization problem. A proof for Theorem 3.9 can be found in [**39**] or [**110**].

3.2.2. A classical atom extraction algorithm. We now restate an algorithm to extract the r atoms from a solution $L \in \mathbb{R}[\mathbf{x}]_{2t}^*$ satisfying Theorem 3.9. Henrion and Lasserre first proposed this algorithm in [**85**]. We only state the main points as several detailed expositions already exist and can be found in [**85**] and [**106**].

By Theorem 3.9 we have that $\mu = \sum_{\ell \in [r]} c_\ell \delta_{\mathbf{x}^{(\ell)}}$, for weights $c_1, c_2, ..., c_r > 0$ and points $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, ..., \mathbf{x}^{(r)} \in K$. Hence, by construction

$$M_s(L) := L([x]_s [x]_s^T) = V_* C V_*^T, \text{ where}$$
$$C := \mathrm{Diag}([c_1, c_2, ..., c_r]) \in \mathbb{R}^{r \times r},\ V_* := [[\mathbf{x}^{(1)}]_s, [\mathbf{x}^{(2)}]_s, ..., [\mathbf{x}^{(r)}]_s] \in \mathbb{R}^{\mathbb{N}_s^n \times r}.$$

In practice, when we find a Cholesky factorization of the moment matrix, it is of the form

$$M_s(L) = VV^T, \text{ for some } V \in \mathbb{R}^{\mathbb{N}_s^n \times r}.$$

Note we use the notation V_* to distinguish the theoretical decomposition of the moment matrix from the decomposition V that we get in actual computations. A key insight is that the matrices V and V_* span the same linear subspace. The extraction algorithm looks at the column operations that transform V into V_*. Via Gaussian elimination with pivoting, we convert V into a matrix U, which is in reduced column echelon form, i.e., of the form

$$\begin{pmatrix}
1 & & & & & 0 \\
\star & & & & & \\
0 & 1 & & & & \\
0 & 0 & 1 & & & \\
\star & \star & \star & & & \vdots \\
\vdots & & & \ddots & & \\
0 & 0 & 0 & \cdots & 1 & \\
\star & \star & \star & \cdots & \star & \\
\vdots & & & & \vdots & \\
\star & \star & \star & \cdots & \star & 0
\end{pmatrix}.$$

Here $\star$ indicates an unknown entry without assigning a specific variable symbol. Three properties characterize the reduced column echelon form: The first nonzero (leading) entry in a column is 1; these entries are also called pivot elements. Every leading entry is to the right of the leading entries above it. Non-zero rows are all to the left of zero rows.

The reduced column echelon form is unique and can be obtained by Gaussian elimination. Note that Gaussian elimination is not numerically robust [**78**]. Fortunately, most problems considered in practice are well-conditioned and do not suffer instability. We refer to Henrion and Lasserre [**85**] for more on this.

Observe that the rows of V and U are indexed by the monomials $[\mathbf{x}]_s$. Denote by $\mathbf{w}(\mathbf{x}) := [\mathbf{x}^{\beta_1}, \mathbf{x}^{\beta_2}, ..., \mathbf{x}^{\beta_r}]^T$ the vector of monomials corresponding to row indices of pivot entries in U. We thus have the following system of polynomial equations to solve:

$$[\mathbf{x}]_s = U\mathbf{w}(\mathbf{x}). \tag{3.17}$$

Solving the above system is where most algorithms start to differ. We mention two approaches to solving (3.17).

The first is *homotopy continuation*, where one establishes a map that continuously deforms an “easy” polynomial system (with known roots) into a “difficult” polynomial system (with unknown roots). By careful numerical tracking of the smooth paths from the first system’s roots, one hopes to discover the roots of the second system. A comprehensive exposition on homotopy continuation would exceed the scope of this thesis. We refer to [**155**] for a dedicated treatment of the topic and further references therein.

The second approach is to look at the common eigenspaces of the so-called multiplication matrices. For each $i \in [n]$ define the $r \times r$ matrix

$$M_{x_i} := \begin{pmatrix} U_{x_i\mathbf{x}^{\beta_1},:} \\ \hline U_{x_i\mathbf{x}^{\beta_2},:} \\ \hline \vdots \\ \hline U_{x_i\mathbf{x}^{\beta_r},:} \end{pmatrix}$$

consisting of the rows of U indexed by the monomials $x_i\mathbf{x}^{\beta_\ell}$ ($\ell \in [r]$) obtained by multiplying $\mathbf{w}(\mathbf{x})$ by x_i. Then, the entries of the atoms $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, ..., \mathbf{x}^{(r)}$ are the eigenvalues of the matrices $M_{x_1}, M_{x_2}, ..., M_{x_n} \in \mathbb{R}^{r\times r}$, i.e.,

$$M_{x_i}\mathbf{w}(\mathbf{x}^{(\ell)}) = \mathbf{x}_i^{(\ell)}\mathbf{w}(\mathbf{x}^{(\ell)}), \text{ for all } i \in [n] \text{ and } \ell \in [r].$$

To recover the common eigenvalues, one considers a random convex combination of the multiplication matrices

$$M := \sum_{i\in[n]} \lambda_i M_{x_i}, \ \lambda \in \Delta^n.$$

With probability 1, M is non-degenerate (i.e., all of its eigenspaces are 1-dimensional). Find a Schur decomposition of $M = QTQ^T$, where $Q = [\mathbf{q}^{(1)}, \mathbf{q}^{(2)}, ... ,\mathbf{q}^{(r)}]$ is an orthogonal matrix and T is an upper-triangular matrix with the same eigenvalues as M sorted in increasing order. Then

$$\mathbf{x}_i^{(\ell)} = \mathbf{q}^{(\ell)}M_{x_i}\mathbf{q}^{(\ell)}$$

for all $i \in [n]$ and $\ell \in [r]$. For a hands-on example, we recommend the reader to consult Example 4.1 of [**106**].

Several open-source software implementations exist for atom extraction. These algorithms are often included in larger code packages that are used to solve optimization problems. For MatLab users, there is *GloptiPoly 3*[2], which is written and maintained by Didier Henrion. *Julia language* [**20**] users have a greater selection of packages, though we only mention two: *MomentTools.jl*[3] which Bernard Mourrain maintains, and *MultivariateMoments.jl* [4], which uses homotopy continuation and is maintained by Benoît Legat.

We now indicate how to apply the above-established procedure for the ideal-sparse setting.

3.2.3. Flatness, atom extraction, and ideal sparsity. Theorem 3.9 and the process described in Section 3.2.2 can be applied to the ideal-sparse setting. Indeed, it suffices to apply Theorem 3.9 independently to each linear functional L_k and check whether its moment matrix satisfies the corresponding flatness criterion (3.16). For each $k \in [p]$ define

$$d_{K_k} := \max\{\deg(f_{i|V_k}), \lceil \deg((g_j)_{|V_k})/2 \rceil : i \in [N_f],\ j \in [N_g]\}.$$

Corollary 3.10. *Assume that* $\max\{d_{K_k} : k \in [p]\} \leq t \in \mathbb{N}$. *Let* $(L_1, ..., L_p)$ *be an optimal solution to the SDP (3.10) with the property that for each* $k \in [p]$ *there exists an* $s_k \in [d_{K_k}, t]$ *such that*

$$\text{rank } L_k([\mathbf{x}(V_k)]_{s_k}[\mathbf{x}(V_k)]_{s_k}^T) = \text{rank } L_k([\mathbf{x}(V_k)]_{s_k - d_{K_k}}[\mathbf{x}(V_k)]_{s_k - d_{K_k}}^T). \quad (3.18)$$

Then, $\xi_t^{\text{isp}} = \mathbf{val}^{\text{isp}} (= \mathbf{val})$, *and problem (3.8a) has an optimal solution* $(\mu_1, ..., \mu_p)$, *where each* μ_k *is finite atomic and supported by* $r_k := \text{rank } L_k([\mathbf{x}(V_k)]_{s_k}[\mathbf{x}(V_k)]_{s_k}^T)$ *many atoms in* K_k.

Analogous to the classical dense setting, atom extraction is carried out independently on each of the p-many moment matrices $L_k([\mathbf{x}(V_k)]_{s_k}[\mathbf{x}(V_k)]_{s_k}^T)$.

[2] https://homepages.laas.fr/henrion/software/gloptipoly/

[3] https://gitlab.inria.fr/AlgebraicGeometricModeling/MomentTools.jl

[4] https://github.com/JuliaAlgebra/MultivariateMoments.jl

Discussion

We now collect some thoughts on the topics of Part 1.

A note on using multiple measures. The idea of optimizing over multiple measures (as we do in the ideal-sparse setting) has already appeared in several other contexts. In fact, it is quite routinely used in most computational methods, e.g., finite elements. In the context of analyzing dynamical systems involving polynomial data, a similar trick has been used to perform optimal control of piecewise-affine systems in [**2**], then later on to characterize invariant measures for piecewise polynomial systems (see [**115**, § 3.5]).

In the context of set estimation, one can also rely on a multi-measure approach to approximate the moments of Lebesgue measures supported on unions of basic semialgebraic sets [**108**].

The common idea consists in using the piecewise structure of the dynamics and/or the state-space partition to decompose the measure of interest into a sum of local measures supported on each partition cell. The advantage in our current setting is that these measures are supported on smaller dimensional spaces, which leads to potentially substantial computational benefits when considering the associated semidefinite programming relaxations.

Trade-off between few big matrices and many small matrices. One could conceive of a situation where computational costs are weighed between computing the ideal-sparse hierarchy (3.9) and computing a "merged hierarchy" (3.11), with $\widetilde{V}_1, ..., \widetilde{V}_{\widetilde{p}}$ somewhere "between" $V_1, ..., V_p$ and $V := [n]$. One would then endeavor to partially use ideal sparsity and ideal constraints to fully utilize (but not exceed) the available computational resources, thereby attaining a best computable (relative to capabilities) bound. However, finding such an optimal configuration appears to be a rather complicated problem. Regardless, the problem is susceptible to estimations and heuristics.

We now consider a special case with a natural heuristic for merging the sets $V_1, ..., V_p$. When $\mathcal{S} = \overline{E}$ is the set of non-edges of some graph $G = (V, E)$, the sets $V_1, ..., V_p$ are interpreted as the maximal cliques of G. In this setting, one can choose the sets $\widetilde{V}_1, ..., \widetilde{V}_{\widetilde{p}}$ to be the maximal cliques of a chordal extension $\widetilde{G}$ of G. That is to say; we add edges to G until the resulting graph

$\widetilde{G}$ is chordal; note this process is not unique in general, not even for minimal (in the number of edges) chordal extensions. Finding the minimal chordal extension of a graph is NP-complete [**7**], but heuristics exist for certain cases (see, e.g., [**21**]). In chordal graphs, the number of maximal cliques is, at most, the number of nodes, i.e., $\widetilde{p} \leq n$.

In our forthcoming Chapters 5 and 6 on matrix factorization ranks, we will consider the set of non-edges $\mathcal{S} = \overline{E}_A$ for the support graph $G_A = ([n], E_A)$ of some matrix A. However, we will only look at the two extreme cases of the dense and ideal-sparse parameters ξ_t and ξ_t^{isp}. For most of the matrices considered, the number of maximal cliques seems not to play a significant role. But, in Section 5.3.2, we do consider a case where $G = ([2n], E)$ is a complete graph with a perfect matching deleted, hence resulting in exponentially (2^n-many) many cliques.

Part 2

Matrix factorization ranks

In light of recent data science trends, new interest has fallen in alternative matrix factorizations. By this, we mean various ways of factorizing matrices (of a particular class) so that the factors have special properties and reveal insights into the original data. We are interested in the specialized ranks associated with these factorizations. As opposed to the familiar linear algebra notion of matrix rank, these specialized ranks are often very difficult to compute, raising the need for easier-to-compute bounds.

We begin this part of the thesis with a general introduction to matrix factorization ranks. Chapter 4 aims to motivate the reader for the topic before defining several different types of factorizations and their associated ranks. It does not contain any of the author's original research but significantly overlaps with a book chapter [**151**], to which the author had the privilege of contributing. We elected to superficially present the different factorization definitions and avoid technical discussions here to facilitate easier comparison among similar concepts. It is also easier to contextualize with previous research and literature. Each section of this chapter is punctuated with references for the erudite reader.

The subsequent three chapters are more technical and can each be read independently. The focus will be on the nonnegative rank (Chapter 5), completely positive rank (Chapter 6), and separable rank (Chapter 7). The content of these chapters is based on: our work with Sander Gribling and Monique Laurent in [**81**]; and our work with Milan Korda, Monique Laurent, and Victor Magron in [**100**]. In particular, we give two novel contributions to the field. The first one is the addition of new polynomial-matrix constraints to the moment hierarchies used to lower bound the completely positive rank and separable rank. Our second major contribution is the application of ideal sparsity to finding better lower bounds for the nonnegative rank, and the completely positive rank.

Each of the considered ranks is given a chapter, and each chapter follows a similar structure: We first construct a hierarchy of lower bounds via the moment method of Chapter 3. Then, we invoke a form of sparsity, ideal-sparsity (see Section 2.2) for nonnegative- and completely positive rank, and a block-diagonal reduction (see Section 7.2) for separable rank. Having built a hierarchy of bounds, we try to link the resulting parameters to other combinatorial bounds from the literature. Each chapter then ends with numerical results and examples.

CHAPTER 4

General theory

4.1. Nonnegative rank

A *nonnegative (NN) factorization* of an entry-wise nonnegative matrix $M \in \mathbb{R}_+^{n\times m}$ is a pair of nonnegative matrices $A \in \mathbb{R}_+^{n\times r}$ and $B \in \mathbb{R}_+^{r\times m}$ for some integer $r \in \mathbb{N}$ such that:

$$M = AB. \tag{4.1}$$

Our interest is in the inner dimension r of the factorization. One can always take $A = M$ and $B = I$, where I is the identity matrix, hence getting $r = m$. However, the interesting case is when $r < \frac{nm}{n+m}$. In this case, one has managed to express the $n \times m$ values of M in terms of the $(n \times r) + (r \times m)$ values of A and B, and as a result, using less storage. The smallest integer r for which this is possible is called the *nonnegative rank*, and is mathematically defined as follows:

$$\operatorname{rank}_+(M) := \min\{r \in \mathbb{N} : M = AB \text{ for some } A \in \mathbb{R}_+^{n\times r},\ B \in \mathbb{R}_+^{r\times m}\}. \tag{4.2}$$

It is not hard to see that the NN rank is sandwiched between the classical rank and the size of the matrix, i.e., $\operatorname{rank}(M) \leq \operatorname{rank}_+(M) \leq \min\{n, m\}$.

However, storage space efficiency is only part of the value of NN factorization. The true power of NN factorization comes from the fact that it is an easy-to-interpret *linear dimensionality reduction* technique. To understand what we mean by this, we first re-examine the relationship between the three matrices M, A, and B in (4.1).

Observe how the j^{th} column of M is given as a conic combination of the columns of A with weights given by the j^{th} column of B, i.e.,

$$M_{:,j} = \sum_{\ell\in[r]} B_{\ell,j} A_{:,\ell}. \tag{4.3}$$

Because all terms involved are nonnegative, zero entries in M force the corresponding entries of the factors to be zero. Formally, for any $i \in [n]$ and $j \in [m]$, $M_{i,j} = 0$ if and only if $B_{\ell,j} A_{i,\ell} = 0$ for all $\ell \in [r]$. Having no cancellation among factors will be useful for interpreting applications of nonnegative factorization. We will explain more with examples in Section 4.1.1. Furthermore, observe that the nonnegative factorization need not be unique. In fact, for any non-singular, nonnegative matrix $P \in \mathbb{R}_+^{r\times r}$ with nonnegative inverse P^{-1} one can produce another factorization

$$M = (AP^{-1})(PB).$$

An example of such a matrix P would be a permutation matrix.

Example 4.1. ***Example of a nonnegative factorization.*** *Consider the following example of a* 4×4 *nonnegative matrix and its nonnegative factorization from which we can deduce that* $\operatorname{rank}_+(M) = 2$ *because* $\operatorname{rank}(M) = 2$*:*

$$M = \begin{bmatrix} 35 & 38 & 41 & 44 \\ 79 & 86 & 93 & 100 \\ 123 & 134 & 145 & 156 \\ 167 & 182 & 197 & 212 \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \\ 7 & 8 \end{bmatrix} \begin{bmatrix} 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix} = AB.$$

4.1.1. Applications of nonnegative factorization. Having introduced the nonnegative rank, we now justify its importance with three applications. What we present here is but a small fraction of the whole body of literature on nonnegative factorization. The interested reader is highly encouraged to read a recent monograph of Gillis [**76**] for an in-depth study of the nonnegative rank with many applications and further references.

Image processing. When analyzing many images, it is natural to ask if the vast bulk is not just a combination of a few “basic images”. This raises two questions. First, how does one find or construct a set of basic images? Second, given this, hopefully small, set of basic images, how does one reproduce the original images? Lee and Seung answered both questions in [**111**], where they factorized a set of images of human faces into typical facial features and nonnegative weights. Combining the weights and features, one approximately recovers the original faces. In this setting, the matrix M has as columns the vectorized gray-scale images of human faces, hence $M_{i,j}$ is the i^{th} pixel of the j^{th} face, with a value between 0 and 1, with 0 corresponding to black and 1 to white.

Recalling the interpretation of (4.3), we can think of the columns of the matrix A as (vectorized) images of human facial features, like a mouth or pair of eyes. Hence, the j^{th} image $M_{:,j}$ is a weighted sum of feature-images $A_{:,\ell}$ ($\ell \in [r]$), where the (nonnegative) weight of feature $A_{:,\ell}$ is given by entry $B_{\ell,j}$ of the matrix B.

In contradistinction to techniques like *principal component analysis* (PCA), which possibly gives factors with negative entries, NN factorization saves us from the task of interpreting notions like “negative pixels” or “image cancellations.” By “negative pixels,” we mean negative factor values, i.e., $B_{\ell,j}A_{i,\ell} < 0$. This means that factor $B_{\ell,j}A_{:,\ell}$ does not just add features but also possibly erases the features added by other factors $B_{k,j}A_{:,k}$, where $k \neq \ell$. A fun by-product of these image factorizations is that one can generate new images by multiplying the matrix A with new weights different from B. However, the resulting images are not guaranteed to look like faces for a poor choice of weights.

Topic recovery and document classification. In text analysis, the matrix M is called the *word occurrence matrix*, and its entries $M_{i,j}$ are the number of times the i^{th} word occurs in the j^{th} document. This way of looking at a corpus of text is often called a "bag of words model," the sequence is ignored, and only the quantity is considered. Since word count is always nonnegative, M is nonnegative and has some NN factors A and B. The columns of matrix A take the meaning of "topics", and B gives the correct weights to recover M. Since there are no cancellations, we observe in the columns of A that certain words tend to occur together, at least within the original set of documents. Moreover, we see how the documents (columns of M) are composed of these base topics (columns of A), with the weight of each topic given by the entries of B. One can hence use these learned topics to group or classify documents.

Linear extension complexity. This third application is different from the above two. First, we define the linear extension complexity, then show how it relates to the nonnegative rank, and finally, we motivate its importance. The *linear extension complexity* of a polytope P is the smallest integer r for which P can be expressed as the linear image of an affine section of $\mathbb{R}^r_+$. Alternatively, the linear extension complexity can be defined as the smallest number of facets a higher dimensional polytope Q can have while still having P as a projection. In 1991 Yannakakis [**171**] proved that the linear extension complexity of P is equal to the nonnegative rank of the *slack matrix* associated with P. For a polytope P, the slack matrix is

$$(d_i - c_i^T v)_{v \in \mathcal{V}, i \in \mathcal{I}},$$

where $c_i \in \mathbb{R}^m,\ d_i \in \mathbb{R}$ come from the hyperplane representation of

$$P = \{x \in \mathbb{R}^m : c_i^T x \leq d_i \quad (i \in \mathcal{I})\},$$

and the vectors $v \in \mathbb{R}^m$ come from the extremal point representation of $P = \operatorname{conv}(\mathcal{V})$. This link between nonnegative rank and linear extension complexity was instrumental in showing why many combinatorial problems, like the traveling salesman problem, could not be efficiently solved simply by lifting the associated problem polytope to higher dimensions in some clever way, see [**68**]. Regarding lifting convex sets, we refer the reader to the survey [**66**].

4.1.2. On computing the nonnegative rank. Given the utility of computing nonnegative factorizations, it is natural to ask: is it difficult to compute the nonnegative rank for a given data matrix $M \geq 0$? This was answered in the affirmative in 2009 by Vavasis [**157**]. Despite being NP-hard to solve, good approximations are sometimes quite accessible. In Chapter 5, we show a general technique for approximating the nonnegative rank from below using the moment method described in Chapter 3.

An alternative, geometrically-motivated approach is to look for a minimal *rectangle cover* for the support of M, see Section 5.2 and [**77**]. Given a matrix $M \in \mathbb{R}^{n\times m}$, one seeks the smallest set of *rectangles*, sets of the form

$$R := \Big\{\{i,j\} : i \in I \subset [n],\ j \in J \subset [m]\Big\},$$

such that for each nonzero entry $M_{i,j} \neq 0$, $\{i,j\}$ belongs to at least one of these rectangles. The minimal number of rectangles needed to accomplish this is a lower bound on the nonnegative rank.

Finding the factorization rank does not necessarily give a factorization. The GMP method that we use next in Chapter 5 does not generally give a factorization, except in a particular case when flatness holds (recall Section 3.2), in which case it is possible to recover the factors. For NN factorization, several algorithms exist that iteratively compute A and B given a guessed value r. However, these algorithms only give approximate factorizations, that is, $M \approx AB$, with respect to some norm. A sufficiently good approximate NN factorization also implies an upper bound on the NN rank. For practical problems, an approximation is often sufficient. For a detailed account of NN factorization, we refer the reader again to the book of Gillis [**76**].

Above, we looked at NN factorization and some of its applications in data analysis and optimization theory. However, there are many more matrix factorization ranks, each having intricacies, applications, and interpretations. Next is the completely positive rank, which can be considered as a symmetric analog of the NN rank.

4.2. Completely positive rank

This factorization is similar to the nonnegative factorization given in (4.1) apart from the modification that we now require $B = A^T$. Formally, a nonnegative matrix $M \in \mathcal{S}^m$ is *completely positive* (CP) if there exists a nonnegative matrix $A \in \mathbb{R}_+^{n\times r}$, for some integer $r \in \mathbb{N}$, such that:

$$M = AA^T. \tag{4.4}$$

Clearly, it is necessary for a CP matrix to be *doubly nonnegative*, i.e., entry-wise nonnegative and *positive semi-definite* (PSD). However, these criteria are not in general sufficient for a matrix to be CP unless $n \leq 4$, see [**18**]. As an example of a 5×5 doubly nonnegative matrix that is not CP, we consider the following example from [**18**].

Example 4.2. ***A doubly nonnegative non-CP matrix*** [**18**, Example 2.9]

$$M = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 \\ 1 & 2 & 1 & 0 & 0 \\ 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & 1 & 2 & 1 \\ 1 & 0 & 0 & 1 & 6 \end{bmatrix}.$$

The nonnegativity is clear; the PSDness is checked via computing all the minors (or using a computer to check the eigenvalues). Non-CPness is harder to check; we refer to the explanation in Berman and Shaked-Monderer's monograph [**18**].

Deciding if a given matrix is CP is already an NP-hard problem; see [**53**]. Because the *completely positive factors* A and A^T are the same, up to transposition, CP factorization is called a *symmetric factorization*. We will see another example shortly at the end of this section. Similar to nonnegative rank, there is a *completely positive rank*, mathematically defined as the smallest inner dimension $r \in \mathbb{N}$ for which a CP factorization of M exists, i.e.,

$$\operatorname{rank}_{\text{cp}}(M) := \min\{r \in \mathbb{N} : M = AA^T \text{ for some } A \in \mathbb{R}_+^{n\times r}\}. \tag{4.5}$$

Clearly, a matrix M is CP if and only if M has finite CP rank; $\operatorname{rank}_{\text{cp}}(M) < \infty$. Hence, computing the CP rank can't be any easier than deciding if M is CP. That being said, the complexity status of computing $\operatorname{rank}_{\text{cp}}(M)$ for a given CP matrix M is unknown to the best of our knowledge. Some upper bounds are known for the CP rank [**146**]:

- $\operatorname{rank}_{\text{cp}}(M) \leq n$, when $n \leq 4$, and
- $\operatorname{rank}_{\text{cp}}(M) \leq \binom{n+1}{2} - 4$ if $n \geq 5$.

In 1994 it was conjectured by Drew, Johnson, and Loewy [**58**] that

$$\operatorname{rank}_{\text{cp}}(M) \leq \lfloor \frac{n^2}{4} \rfloor,$$

the bound being only attained for CP matrices M that have complete bipartite support graphs. This conjecture was disproved by Bomze et al. [**22, 23**] two decades later, using several specially constructed counter-examples. We show in (4.6) an example, namely $\widetilde{M_7}$ from [**22**], of size $n = 7$, with

$$\operatorname{rank}_{\text{cp}}(\widetilde{M_7}) = 14 > \lfloor \frac{49}{4} \rfloor = 12,$$

$$\widetilde{M_7} = \begin{bmatrix} 163 & 108 & 27 & 4 & 4 & 27 & 108 \\ 108 & 163 & 108 & 27 & 4 & 4 & 27 \\ 27 & 108 & 163 & 108 & 27 & 4 & 4 \\ 4 & 27 & 108 & 163 & 108 & 27 & 4 \\ 4 & 4 & 27 & 108 & 163 & 108 & 27 \\ 27 & 4 & 4 & 27 & 108 & 163 & 108 \\ 108 & 27 & 4 & 4 & 27 & 108 & 163 \end{bmatrix}. \tag{4.6}$$

On the applied side, CP matrices occur in the theory of *block designs*. We omit many details here, but essentially, block designs deal with arranging distinct objects into blocks so that the objects occur with certain regularity within and among the blocks. There is a direct application of block designs in designing experiments, where researchers wish to prevent the differences between test subjects from obfuscating the differences in outcome due to treatment; see [**83**] for more on block designs and see [**145**] for the link between block designs and CP matrices.

From another perspective, CP matrices are of great interest in optimization. Indeed, de Klerk and Pasechnik [**43**, Theorem 2.2] showed that computing the stability number of a graph could be recast as a linear optimization problem over the cone of CP matrices. Later, Burer [**28**] expanded on this result by showing that any nonconvex quadratic program with binary and continuous variables could be reformulated as a linear program over the cone of CP matrices. This effectively meant that many NP-hard problems could now be viewed as linear programs with CP membership constraints. This reformulation does not make the problems any easier to solve as the difficulty is now pushed into characterizing the cone of CP matrices. However, it does allow us to attack a large class of problems by understanding the unifying thread, complete positivity.

For a thorough account of completely positive and copositive matrices (the natural dual cone to CP matrices), we refer the inquisitive reader to the monograph by Berman and Shaked-Monderer [**145**].

4.3. Separable rank

In the quantum information theory setting, the state of a physical system is often characterized by a Hermitian PSD matrix $M \in \mathcal{H}^n \otimes \mathcal{H}^n$. A state M is said to be *separable* if there exists an integer $r \geq 1$ and vectors $\mathbf{a}_1, ..., \mathbf{a}_r, \mathbf{b}_1, ..., \mathbf{b}_r \in \mathbb{C}^n$ for which

$$M = \sum_{\ell \in [r]} \mathbf{a}_\ell \mathbf{a}_\ell^* \otimes \mathbf{b}_\ell \mathbf{b}_\ell^*. \tag{4.7}$$

We will not go into the quantum physics details. Instead, we refer the reader to [**129, 165**] and the references therein. It suffices to think of separable states as fully explained by classical physics, in contradistinction to non-separable states, a.k.a. *entangled states*, that have special non-classical properties of interest in quantum physics. For a rank-one state (a.k.a a *pure state*), i.e., if $\operatorname{rank}(M) = 1$, one can obtain a separable factorization by using *singular value decomposition (SVD)*. Non-rank-one states are called *mixed states*, and deciding whether a mixed state M is separable is generally NP-hard, see [**82, 72**].

Example 4.3. ***Example of an entangled state*** [**35**] *Consider the following mixed state of size 9×9, hence $n = 3$. We have omitted to show zeros for readability and drawn grid lines to highlight the block structure.*

$$M = \left[\begin{array}{ccc|ccc|ccc} 1 & & & & 1 & & & & 1 \\ & 2 & & 1 & & & & & \\ & & \frac{1}{2} & & & & 1 & & \\ \hline & 1 & & \frac{1}{2} & & & & & \\ 1 & & & & 1 & & & & 1 \\ & & & & & 2 & & 1 & \\ \hline & & 1 & & & & 2 & & \\ & & & & & 1 & & \frac{1}{2} & \\ 1 & & & & 1 & & & & 1 \end{array}\right].$$

See [**35**] *for a proof that M is entangled.*

Analogously to matrix ranks we considered thus far, there is also an associated notion of rank, namely, the *separable rank* [**46**] (a.k.a *optimal ensemble cardinality* [**55**]). For a separable matrix M, we define its SEP rank as

$$\operatorname{rank}_{\text{sep}}(M) := \min\left\{r \in \mathbb{N} : \exists\ \mathbf{a}_\ell, \mathbf{b}_\ell \in \mathbb{C}^n \text{ s.t. } M = \sum_{\ell \in [r]} \mathbf{a}_\ell \mathbf{a}_\ell^* \otimes \mathbf{b}_\ell \mathbf{b}_\ell^*\right\}. \quad (4.8)$$

A possible interpretation of the separable rank is that it gives a sense of how complex a classical system is, with the convention being that an entangled state has infinite separable rank. To the best of our knowledge, the complexity of computing the separable rank is still unknown. There are some crude bounds on the separable rank, namely:

$$\operatorname{rank}(M) \leq \operatorname{rank}_{\text{sep}}(M) \leq \operatorname{rank}(M)^2.$$

The left-most inequality can be strict (see [**55**]), and the right-most inequality follows from Caratheodory's theorem [**156**].

In addition to the above definition, there are several other variations on this notion of separability. One variation is to look for factorizations of the form $M = \sum_{\ell \in [r]} A_\ell \otimes B_\ell$, where $A_\ell, B_\ell \in \mathcal{H}^n$ are Hermitian PSD matrices (as opposed to rank-one Hermitian PSD matrices). From this, it is easy to define the associated *mixed separable rank* as the smallest r for which such a factorization is possible, i.e.,

$$\operatorname{rank}_{\text{mixsep}}(M) := \min\left\{r \in \mathbb{N} : \exists\ A^{(\ell)},\ B^{(\ell)} \in \mathcal{H}_+^n \text{ s.t. } M = \sum_{\ell \in [r]} A^{(\ell)} \otimes B^{(\ell)}\right\}.$$

When M is diagonal, its mixed separable rank equals the nonnegative rank of an associated $n \times n$ matrix consisting of the diagonal entries of M, see [**47**]. This shows that the mixed separable rank is hard to compute.

4.4. Tensor ranks

Tensors, or multi-way arrays, are natural generalizations of matrices that are commonly encountered in applied fields such as engineering, computer vision, and data science. It is to be expected, then, that matrix factorization generalizes to tensor factorization. A comprehensive introduction to tensor factorization ranks falls beyond the scope of this thesis. However, the separable factorization we considered prior is a good example. We build on the similarities with some remarks and references to further material.

Consider, for example, a three-way array $T \in \mathbb{R}^{n\times m\times p}$. Its *tensor rank* is the smallest number $r \in \mathbb{N}$ of rank-one tensors (tensors of the form $\mathbf{a}\otimes\mathbf{b}\otimes\mathbf{c}$ for some $\mathbf{a} \in \mathbb{R}^n$, $\mathbf{b} \in \mathbb{R}^m$, and $\mathbf{c} \in \mathbb{R}^p$) necessary to describe T, i.e.,

$$\operatorname{rank}_{\text{tensor}}(T) = \min\{r \in \mathbb{N} : T = \sum_{\ell\in[r]} \mathbf{a}_\ell\otimes\mathbf{b}_\ell\otimes\mathbf{c}_\ell,\ \mathbf{a}_\ell \in \mathbb{R}^n,\ \mathbf{b}_\ell \in \mathbb{R}^m,\ \mathbf{c}_\ell \in \mathbb{R}^p\}.$$

Similarly, one can define the *nonnegative tensor rank* by requiring the factors $\mathbf{a}_\ell, \mathbf{b}_\ell$, and $\mathbf{c}_\ell$ to be nonnegative. Moreover, one can define the *symmetric tensor rank* by requiring $n = m = p$ and forcing the factors to be equal, i.e., $\mathbf{a}_\ell = \mathbf{b}_\ell = \mathbf{c}_\ell$ for all $\ell \in [r]$. An interesting effect of going to tensors is that some decompositions become unique [**148**]. See [**36**] for an applications-centric monograph on tensor factorization. For a mathematical survey, see Kolda and Bader [**97**].

4.5. Non-commutative matrix ranks

This section examines two non-commutative analogs of NN factorization and CP factorization.

A *positive semidefinite (PSD) factorization* is when, for a nonnegative matrix $M \in \mathbb{R}^{n\times m}_+$ we look for an $r \in \mathbb{N}$, and PSD matrices

$$A_1, ..., A_n, B_1, ..., B_m \in \mathcal{S}^r_+$$

such that the matrix M is described entry-wise as follows:

$$M_{i,j} = \langle A_i, B_j\rangle, \text{ for } i \in [n] \text{ and } j \in [m].$$

If the matrices A_i ($i \in [n]$) and B_j ($j \in [m]$) are diagonal, then we recover a nonnegative factorization. Similar to nonnegative factorization, there is a substantial research interest in PSD factorization, largely due to its many appealing geometric interpretations, including semidefinite representations of polyhedra. We refer the reader to the survey by Fawzi et al. [**65**] for further study of PSD-factorizations.

A *completely positive semidefinite factorization* is the symmetric analog of PSD factorization and the non-commutative analog of CP factorization. Completely PSD factorization differs from PSD factorization only in that it requires $n = m$ and $B_i = A_i$ for all $i \in [n]$.

We refer the reader to [**80**] for a deeper treatment of these two non-commutative ranks and their commutative analogs.

CHAPTER 5

Nonnegative rank

This chapter focuses on the nonnegative rank of a nonnegative matrix. We begin with a quick recap of definitions. Then, we apply the results of Chapter 3 to build a hierarchy of lower bounds for the nonnegative rank (Section 5.1).

Our new contribution to this field is the exploitation of ideal sparsity (recall Section 2.2) to build a possibly stronger and easier-to-compute hierarchy (Section 5.1.2).

Having defined two hierarchies, we compare them to each other and other known combinatorial bounds from the literature in Section 5.2. Lastly, we present our numerical results comparing the dense and ideal-spare hierarchies in Section 5.3.

5.1. Hierarchies of lower bounds for the nonnegative rank

For a nonnegative matrix $M \in \mathbb{R}^{n\times m}_+$, its *nonnegative rank* is defined in two equivalent ways

$$\begin{aligned}\operatorname{rank}_+(M) :=& \min\{r \in \mathbb{N} : M = AB,\ A \in \mathbb{R}^{n\times r}_+,\ B \in \mathbb{R}^{r\times m}_+\},\\ =& \min\{r \in \mathbb{N} : M = \sum_{\ell\in[r]} \mathbf{a}_\ell \mathbf{b}_\ell^T,\ \mathbf{a}_\ell \in \mathbb{R}^n_+,\ \mathbf{b}_\ell \in \mathbb{R}^m_+\}.\end{aligned} \tag{5.1}$$

Here we used $A = (\mathbf{a}_1|...|\mathbf{a}_r)$ and $B = (\mathbf{b}_1|...|\mathbf{b}_r)^T$.

The nonnegative rank is a combinatorial parameter and it is NP-hard to compute in general [**157**]. To find good approximations for the nonnegative rank, one can consider parameters obtained via somehow relaxing the definition in (5.1). The following is a "natural convexification" of the parameter $\operatorname{rank}_+(M)$ proposed by Fang and Parrilo [**67**]:

$$\tau_+(M) = \inf\Big\{\lambda : \frac{1}{\lambda}M \in \operatorname{conv}\{\mathbf{x}\mathbf{y}^T : \mathbf{x} \in \mathbb{R}^n_+,\ \mathbf{y} \in \mathbb{R}^m_+,\ M \geq \mathbf{x}\mathbf{y}^T\}\Big\}. \tag{5.2}$$

Observe that this new parameter lower bounds the nonnegative rank, i.e.,

$$\tau_+(M) \leq \operatorname{rank}_+(M) \text{ for all } M \in \mathbb{R}^{n\times m}_+. \tag{5.3}$$

Indeed, any nonnegative factorization $M = \sum_{\ell\in[r]} \mathbf{a}_\ell \mathbf{b}_\ell^T$ induces a solution $\lambda = \frac{1}{r}$ to (5.2) because

$$\frac{1}{r}M = \sum_{\ell\in[r]} \frac{1}{r}\mathbf{a}_\ell \mathbf{b}_\ell^T \in \operatorname{conv}\{\mathbf{x}\mathbf{y}^T : \mathbf{x} \in \mathbb{R}^n_+,\ \mathbf{y} \in \mathbb{R}^m_+,\ M \geq \mathbf{x}\mathbf{y}^T\}.$$

The optimization problem (5.2) may seem more challenging to solve than (5.1), but it has the advantage that it admits a GMP formulation.

GMP formulation of the parameter $\tau_+(M)$***.*** Before formulating the GMP, we need two technicalities. We need to define a semialgebraic set, which requires specifying the support graph of a nonnegative matrix M, and we assume a particular scaling on the negative factors.

Let $V := [n+m] = U \cup W$, with

- $U := [n] = \{1, \ldots, n\}$ corresponding to the row indices of M, and
- $W := \{n+1, \ldots, n+m\}$ corresponding (up to shifting) to the column indices of M.

Define two sets of pairs of indices

$$\begin{aligned} E^M &:= \Big\{\{i,j\} \in U \times W : M_{i,j-n} \neq 0\Big\}, \\ \overline{E}^M &:= \Big\{\{i,j\} \in U \times W : M_{i,j-n} = 0\Big\}. \end{aligned} \tag{5.4}$$

The set E^M is the edge set of the (bipartite) support graph $G^M := (V, E^M)$ of the matrix M. The set of non-edges of G^M is $\overline{E}^M$.

As observed in [**80**], one may assume without loss of generality (after rescaling) that the nonnegative factors $\mathbf{a}_\ell$, $\mathbf{b}_\ell$ ($\ell \in [r]$) in (5.1) satisfy

$$\|\mathbf{a}_\ell\|_\infty, \|\mathbf{b}_\ell\|_\infty \leq \sqrt{M_{\max}}.$$

Here, $M_{\max} := \max_{i\in[n],\ j\in[m]} M_{i,j}$ denotes the largest entry of M.

Because it is convenient to think of the indices ($i \in [n]$ and $j \in [m]$) of the matrix M as corresponding to vertices in a graph, we henceforth use the following renaming of variables:

$$y_1, ..., y_m \to x_{n+1}, ..., x_{n+m}.$$

Now, we have all the prerequisites to define the following semialgebraic domain:

$$\begin{aligned} K^M := \Big\{\mathbf{x} \in \mathbb{R}^{m+n} :& \sqrt{M_{\max}}x_i - x_i^2 \geq 0 \ (i \in [m+n]), && (5.5\text{a}) \\ & M_{i,j-n} - x_i x_j \geq 0 \ (\{i,j\} \in E^M), && (5.5\text{b}) \\ & x_i x_j = 0 \ (\{i,j\} \in \overline{E}_M)\Big\}. && (5.5\text{c}) \end{aligned}$$

Lemma 5.1. *The parameter* $\tau_+(M)$ *is equal to the optimal value of the following generalized moment problem:*

$$\mathbf{val}_+(M) := \inf_{\mu \in \mathscr{M}(K^M)} \Big\{ \int 1 d\mu : \int x_i x_j d\mu = M_{i,j-n} \ (i \in U,\ j \in W) \Big\}. \tag{5.6}$$

Proof. **($\mathbf{val}_+(M) \leq \tau_+(M)$)** Any feasible solution to $\tau_+(M)$, i.e., a decomposition of the form $M = \lambda \sum_{\ell \in [s]} c_\ell \mathbf{a}_\ell \mathbf{b}_\ell^T$, with $\lambda > 0$, $\sum_{\ell \in [s]} c_\ell = 1$, $c_\ell > 0$, $(\mathbf{a}_\ell, \mathbf{b}_\ell) \geq 0$, and $M \geq \mathbf{a}_\ell \mathbf{b}_\ell^T$, corresponds to a finite atomic measure

$$\mu := \lambda \sum_{\ell \in [s]} c_\ell \delta_{(\mathbf{a}_\ell, \mathbf{b}_\ell)}$$

that is feasible for $\mathbf{val}_+(M)$ with objective value λ. Hence, $\mathbf{val}_+(M) \leq \tau_+(M)$.

($\mathbf{val}_+(M) \geq \tau_+(M)$) Assume the GMP (5.6) is feasible, else $\mathbf{val}_+(M) = \infty$, and we have nothing to prove. Let $\mu \in \mathscr{M}(K^M)$ be a feasible solution to (5.6). In view of Theorem 2.8 (ii), we may assume that μ is a finite atomic measure, i.e.,

$$\mu := \lambda \sum_{\ell \in [r]} c_\ell \delta_{(\mathbf{a}_\ell, \mathbf{b}_\ell)},$$

with $\lambda > 0$, $\sum_{\ell \in [r]} c_\ell = 1$, $c_\ell > 0$, and $(\mathbf{a}_\ell, \mathbf{b}_\ell) \in K^M$. This measure then induces a decomposition

$$M = \lambda \sum_{\ell \in [r]} c_\ell \mathbf{a}_\ell \mathbf{b}_\ell^T$$

corresponding to a solution to (6.3), with value λ. Hence, $\mathbf{val}_+(M) \geq \tau_+(M)$. If $\mathbf{val}_+$ is infeasible, then both parameters $\tau_+(M)$ and $\mathbf{val}_+(M)$ are infeasible and thus equal to ∞. □

The quadratic module is Archimedean. Consider the quadratic module $\mathcal{M}(H)$ generated by the positivity constraints (5.5a) defining K^M,

$$H := \Big\{ \sqrt{M_{\max}} x_i - x_i^2 \ (i \in [m+n]) \Big\}.$$

Observe that

$$\begin{aligned} (n+m)M_{\max} - \sum_{i \in [n+m]} x_i^2 &= \sum_{i \in [n+m]} \Big(M_{\max} - x_i^2 \Big) \\ = \sum_{i \in [n+m]} \Big((\sqrt{M_{\max}} - x_i)^2 + 2(\sqrt{M_{\max}} x_i - x_i^2) \Big) &\in \mathcal{M}_2(H). \end{aligned} \tag{5.7}$$

Here, the last line of the equation is clearly in the quadratic module $\mathcal{M}(H)$, and the first line contains the Archimedean certificate of $\mathcal{M}(H)$. Our argument here is paraphrased from Section 2 of [**80**].

5.1.1. A (dense) hierarchy of lower bounds for $\tau_+(M)$.

We now build a hierarchy of semidefinite programs using the moment method described in Section 3.1. For each $t \in \mathbb{N} \cup \{\infty\}$ define the following parameter that provides a lower bound for $\tau_+(M)$:

$$\begin{aligned}\xi_t^+(M) := \min \Big\{ & L(1) : \\ & L \in \mathbb{R}[x_1, ..., x_{m+n}]_{2t}^*, \\ & L(x_i x_j) = M_{i,j-n} \ (i \in U, j \in W), && (5.8a) \\ & L([\mathbf{x}]_t [\mathbf{x}]_t^T) \succeq 0, && (5.8b) \\ & L((\sqrt{M_{\max}} x_i - x_i^2)[\mathbf{x}]_{t-1}[\mathbf{x}]_{t-1}^T) \succeq 0 \ (i \in V), && (5.8c) \\ & L((M_{i,j-n} - x_i x_j)[\mathbf{x}]_{t-1}[\mathbf{x}]_{t-1}^T) \succeq 0 \ (\{i,j\} \in E^M), && (5.8d) \\ & L(x_i x_j [\mathbf{x}]_{2t-2}) = 0 \ \text{ for } \{i,j\} \in \overline{E}^M \Big\}. && (5.8e)\end{aligned}$$

If we omit the (ideal) constraint (5.8e) and require the constraint (5.8d) to hold also for pairs $\{i, j\} \in \overline{E}^M$, then we obtain a (possibly weaker) parameter, first introduced in [**80**]. In [**80**], this parameter was also denoted by $\xi_t^+(M)$; to not conflict with our current notation, we now denote it by $\xi_t^{+(2019)}(M)$. Hence, we have

$$\xi_t^{+(2019)}(M) \leq \xi_t^+(M) \leq \tau_+(M) \leq \text{rank}_+(M).$$

Moreover, asymptotic convergence of $\xi_t^{+(2019)}(M)$ to $\tau_+(M)$, i.e.,

$$\lim_{t\to\infty} \xi_t^{+(2019)}(M) = \xi_\infty^{+\,(2019)}(M) = \tau_+(M),$$

was already shown in [**80**]. Thus we can conclude

$$\lim_{t\to\infty} \xi_t^+(M) = \xi_\infty^+(M) = \tau_+(M). \tag{5.9}$$

The core tool underlying these convergence results is Theorem 3.3, where assumption (A) is satisfied because of (5.3), assumption (B) because of (5.7), and assumption (C) holds by taking $z_{i,j} = 0$ $(i \in U, j \in W)$ and $c = \frac{1}{2}$.

For completeness, we state the conclusion of Theorem 3.3 for the hierarchy $\xi_t^+(M)$ and parameter $\tau_+(M)$. For each $t \in \mathbb{N} \cup \{\infty\}$, the program (5.8) attains it optimum, and

$$\lim_{t\to\infty} \xi_t^+(M) = \xi_\infty^+(M) = \tau_+(M).$$

Moreover, the GMP (5.6) has an optimal solution μ that is finite atomic and is supported on K^M.

5.1.2. An ideal-sparse hierarchy of lower bounds for $\tau_+(M)$. Observe how the sparsity pattern of M naturally gives rise to the ideal constraints in (5.5c). Given the special form of these ideal constraints, we may apply ideal sparsity from Section 3.1.4 to construct an ideal-sparse hierarchy that we denote by $\xi_t^{+,\text{isp}}(M)$.

Because the support graph G^M is bipartite, the maximal subsets $V_1, ..., V_p$ of $V = [n + m]$ that do not contain any pair $\{i, j\} \in \overline{E}^M$ (recall the general definition in Section 2.2.1) can now be interpreted as the vertex sets of all the maximal bicliques of G^M. A *biclique* in G^M corresponds to a complete bipartite subgraph, and it is thus given by a pair (A, B) with $A \subseteq U$ and $B \subseteq W$ such that $\{i, j\} \in E^M$ for all $(i, j) \in A \times B$; it is called maximal if $A \cup B$ is maximal in the vertex-set inclusion sense. For any $t \in \mathbb{N} \cup \{\infty\}$, define the parameter

$$\begin{aligned}
\xi_t^{+,\mathrm{isp}}(M) = \min \Big\{ & \sum_{k \in [p]} L_k(1) : \\
& L_k \in \mathbb{R}[\mathbf{x}(V_k)]_{2t}^* \ (k \in [p]), \\
& \sum_{k \in [p] : \{i,j\} \subseteq V_k} L_k(x_i x_j) = M_{i,j-n} \ (i \in U, \ j \in W), && \text{(5.10a)} \\
& L_k([\mathbf{x}(V_k)]_t [\mathbf{x}(V_k)]_t^T) \succeq 0 \ (k \in [p]), && \text{(5.10b)} \\
& L_k((\sqrt{M_{\max}} x_i - x_i^2)[\mathbf{x}(V_k)]_{t-1}[\mathbf{x}(V_k)]_{t-1}^T) \succeq 0 \ (i \in V_k, \ k \in [p]), && \text{(5.10c)} \\
& L_k((M_{i,j-n} - x_i x_j)[\mathbf{x}(V_k)]_{t-1}[\mathbf{x}(V_k)]_{t-1}^T) \succeq 0 \ (\{i,j\} \subseteq V_k, \ k \in [p]) \Big\}. && \text{(5.10d)}
\end{aligned}$$

Though the definition of $\xi_t^{+,\mathrm{isp}}(M)$ looks much more cumbersome than its dense counterpart $\xi_t^+(M)$, they both largely follow the same structure. As noted in Section 2.2.1, the ideal constraints from (5.8e) are captured in the supports $V_1, .., V_p$.

By direct application of Theorem 3.4 we have, for any $t \in \mathbb{N} \cup \{\infty\}$, the following inequalities among the above parameters:

$$\xi_t^+(M) \leq \xi_{t+1}^+(M) \leq \xi_{t+1}^{+,\mathrm{isp}}(M) \leq \tau_+(M) \leq \mathrm{rank}_+(M).$$

Moreover, we have asymptotic convergence of $\xi_t^{+,\mathrm{isp}}(M)$ to $\tau_+(M)$, which follows from the convergence of the dense hierarchy $\xi_t^+(M)$.

Adding scalar localizing constraints based on nonnegativity. More constraints may be added to the above programs to strengthen the bounds. In [**80**], the authors propose to exploit the nonnegativity of the variables and add the linear constraints

$$L((M_{i,j-n} - x_i x_j)[\mathbf{x}]_{2t-2}) \geq 0 \ ((i,j) \in U \times W), \tag{5.11}$$

$$L((\sqrt{M_{\max}} x_i - x_i^2)[\mathbf{x}]_{2t-2}) \geq 0 \ (i \in V), \tag{5.12}$$

$$L([\mathbf{x}]_{2t}) \geq 0. \tag{5.13}$$

Adding constraint (5.11) to the parameter $\xi_t^+(M)$ results in a possibly stronger parameter we denote by $\xi_{t,\dagger}^+(M)$. If we add all the constraints (5.11), (5.12), and (5.13) to $\xi_t^+(M)$, then we get a possibly even stronger parameter $\xi_{t,\ddagger}^+(M)$.

We can mutatis mutandis define ideal-sparse parameters $\xi_{t,\dagger}^{+,\mathrm{isp}}(M)$ and $\xi_{t,\ddagger}^{+,\mathrm{isp}}(M)$ by using the following ideal-sparse analogs of the constraints (5.11), (5.12), and (5.13):

$$\begin{gathered}L_k((M_{i,j-n} - x_i x_j)[\mathbf{x}]_{2t-2}) \geq 0 \ (\{i,j\} \subseteq V_k,\ k \in [p]),\\ L_k((\sqrt{M_{\max}} x_i - x_i^2)[\mathbf{x}(V_k)]_{2t-2}) \geq 0 \ (i \in V_k,\ k \in [p]),\\ L([\mathbf{x}(V_k)]_{2t}) \geq 0 \ (k \in [p]).\end{gathered}$$

Thus, we have

$$\begin{array}{ccccc}\xi_t^{+,\mathrm{isp}}(M) & \leq & \xi_{t,\dagger}^{+,\mathrm{isp}}(M) & \leq & \xi_{t,\ddagger}^{+,\mathrm{isp}}(M)\\ \vee\!\mid & & \vee\!\mid & & \vee\!\mid\\ \xi_t^{+}(M) & \leq & \xi_{t,\dagger}^{+}(M) & \leq & \xi_{t,\ddagger}^{+}(M).\end{array}$$

Observe that the constraints are all linear. Hence, they are not as costly to implement as PSD constraints, which are often the computational bottleneck in SDP hierarchies.

5.2. Links to other lower bounds on the nonnegative rank

We now recall two lower bounds on the nonnegative rank from existing literature. The first is due to Fawzi and Parrilo, who proposed an SoS-based relaxation $\tau_+^{\mathrm{sos}}(M)$ of the parameter $\tau_+(M)$. The second is a more classical result that looks at minimal edge covers for the support graph G^M of M. At the end of this section, we relate and summarize all the parameters we have introduced thus far in the chapter.

The bound $\tau_+^{\mathrm{sos}}(M)$. Fawzi and Parrilo [**67**] introduced a semidefinite bound $\tau_+^{\mathrm{sos}}(M)$ and showed that it satisfies $\tau_+^{\mathrm{sos}}(M) \leq \tau_+(M)$. In [**80**] it is shown that the parameter $\xi_{2,\dagger}^{+}(M)$ possibly improves on this bound [1]

$$\tau_+^{\mathrm{sos}}(M) \leq \xi_{2,\dagger}^{+}(M) \leq \tau_+(M).$$

Edge biclique-cover bound. We now define a well-known combinatorial lower bound on the nonnegative rank, called the edge biclique-cover number. Recall that the support graph $G^M = (U \cup W, E^M)$ of $M \in \mathbb{R}_+^{m\times n}$ is a bipartite graph. Define the *edge biclique-cover number* of G^M, denoted $\mathrm{bc}(G^M)$, as the smallest number of bicliques whose union covers every edge in E^M. Observe

[1] This follows from the proof of [**80**, Proposition 15], since it only uses the relation $L((M_{i,j-n} - x_i x_j)x_i x_j) \geq 0$ for any $(i,j) \in U \times W$ in addition to the constraints defining the basic parameter $\xi_2^{+}(M)$.

that a nonnegative factorization $M = \sum_{\ell\in[r]} \mathbf{a}_\ell \mathbf{b}_\ell^T$ induces an edge biclique-cover $\{A^\ell \cup B^\ell\}_{\ell\in[r]}$ of G^M, where the sets are given by $A^\ell := \operatorname{supp}(\mathbf{a}_\ell)$ and $B^\ell := \operatorname{supp}(\mathbf{b}_\ell)$. Hence we have

$$\mathrm{bc}(G^M) \leq \mathrm{rank}_+(M).$$

Any biclique in G^M corresponds to a pair $(A, B) \subseteq U \times W$ for which the *rectangle* $A \times B$ is entirely contained in the support of M. Because of the bijective correspondence between bicliques of G^M and rectangles supported on $\operatorname{supp}(M)$, the parameter $\mathrm{bc}(G^M)$ is also known as the *rectangle covering number* of M (see, e.g., [**67, 77**]). Relaxing the integrality constraint in $\mathrm{bc}(G^M)$, we can define a fractional analog, called the *fractional rectangle covering number* as follows:

$$\mathrm{bc}_{\mathrm{frac}}(G^M) := \min\Big\{ \sum_{k\in[p]} \lambda_k : \lambda \in \mathbb{R}^p_+, \sum_{k:\{i,j\}\subseteq V_k} \lambda_k \geq 1 \ (\{i,j\} \in E^M)\Big\}. \quad (5.14)$$

If we require that λ be integer valued, then (5.14) becomes $\mathrm{bc}(G^M)$. Hence we have

$$\mathrm{bc}_{\mathrm{frac}}(G^M) \leq \mathrm{bc}(G^M).$$

We now show that the first level of the ideal-sparse hierarchy is at least as good as the fractional rectangle covering number.

LEMMA 5.2. *For any $M \in \mathbb{R}^{n\times m}_+$ we have*

$$\xi_1^{+,\mathrm{isp}}(M) \geq \mathrm{bc}_{\mathrm{frac}}(G^M).$$

PROOF. Let $(L_1, ..., L_p)$ be an optimal solution for the program (5.10) defining $\xi_1^{+,\mathrm{isp}}(M)$. Observe that by (5.10d) we have $L_k(M_{i,j-n} - x_i x_j) \geq 0$ and hence $M_{i,j-n} \cdot L_k(1) \geq L_k(x_i x_j) \geq 0$ for all $\{i,j\} \in E^M$, $\{i,j\} \subseteq V_k$, $k \in [p]$. Hence, by condition (5.10a) we have, for every fixed $\{i,j\} \in E^M$, that

$$M_{i,j-n} \sum_{k\in[p]:\{i,j\}\subseteq V_k} L_k(1) \geq \sum_{k\in[p]:\{i,j\}\subseteq V_k} L_k(x_i x_j) = M_{i,j-n}.$$

Because $M_{i,j-n} \neq 0$ for all $\{i,j\} \in E^M$, it follows that $\sum_{k\in[p]:\{i,j\}\subseteq V_k} L_k(1) \geq 1$ for each edge $\{i,j\} \in E^M$. Hence, the vector $\lambda := (L_k(1))_{k\in[p]}$ provides a feasible solution to program (5.14), implying that

$$\xi_1^{+,\mathrm{isp}}(M) = \sum_{k\in[p]} L_k(1) \geq \mathrm{bc}_{\mathrm{frac}}(G^M). \qquad \square$$

Separation between ideal-sparse and dense bounds. The ideal-sparse bounds can be arbitrarily better than the dense bounds, even at level $t = 1$. To demonstrate this claim, we consider the matrix $M = I_n$.

Example 5.3. ***Identity matrices separate ideal-sparse and dense bounds.*** *Consider the identity matrix $M = I_n \in \mathcal{S}^n$. Clearly, we have* $\mathrm{rank}_+(I_n) = \mathrm{rank}(I_n) = n$. *As the support graph G^M is the disjoint union of n edges, its fractional edge biclique-cover number is equal to n, and thus, in view of Lemma 5.2, we have the equalities $\xi_1^{+,\mathrm{isp}}(I_n) = n = \mathrm{rank}_+(I_n)$. We now show that for the dense bound, we have $\xi_1^+(I_n) < 8$ for any $n \geq 4$. For this, recall that $\xi_1^+(I_n)$ is given by*

$$\begin{aligned}\xi_1^+(I_n) = \min\Big\{L(1) : \ & L \in \mathbb{R}[\mathbf{x}]_2^*,\\ & L(x_i) \geq L(x_i^2) \ (i \in [2n]),\\ & L(x_i x_{n+j}) = \delta_{i,j} \ (i,j \in [n]),\\ & L([\mathbf{x}]_1[\mathbf{x}]_1^T) \succeq 0\Big\},\end{aligned} \tag{5.15}$$

where $\mathbf{x} = (x_1, ..., x_{2n})$. Consider the linear functional $L \in \mathbb{R}[\mathbf{x}]_2^*$ defined by $L(1) = 8\frac{n-2}{n}$, $L(x_i) = L(x_i^2) = 2\frac{n-2}{n}$ for $i \in [2n]$, $L(x_i x_j) = L(x_{n+i}x_{n+j}) = \frac{n-4}{n}$ for $i \neq j \in [n]$, and $L(x_i x_{n+j}) = \delta_{i,j}$ for $i, j \in [n]$. Then, one can check that

$$L([x]_1[x]_1^T) = \begin{pmatrix} 8\frac{n-2}{n} & 2\frac{n-2}{n}e^T & 2\frac{n-2}{n}e^T \\ 2\frac{n-2}{n}e & I_n + \frac{n-4}{n}J_n & I_n \\ 2\frac{n-2}{n}e & I_n & I_n + \frac{n-4}{n}J_n \end{pmatrix} \succeq 0.$$

Hence, L is feasible for the program defining $\xi_1^+(I_n)$, which shows the upper bound $\xi_1^+(I_n) \leq L(1) = 8\frac{n-2}{n} < 8$.

Summary of lower bounds on the nonnegative rank. For the reader's convenience, we summarize the parameters of this chapter and their relations to each other. For any $t \in \{2, 3, ...\} \cup \{\infty\}$ we have the following:

$$\begin{array}{lllll}
\mathrm{bc}_{\mathrm{frac}}(G^M) & \leq \xi_1^{+,\mathrm{isp}}(M) & \leq \xi_t^{+,\mathrm{isp}}(M) & \leq \xi_{t,\dagger}^{+,\mathrm{isp}}(M) & \leq \xi_{t,\ddagger}^{+,\mathrm{isp}}(M) \leq \tau_+(M) \leq \mathrm{rank}_+(M) \\
\quad \underline{\wedge} & \quad \underline{\vee} & \quad \underline{\vee} & \quad \underline{\vee} & \\
\mathrm{bc}(G^M) & \xi_1^+(M) & \leq \xi_2^+(M) & \leq \xi_{2,\dagger}^+(M) & \\
\quad \underline{\wedge} & & & \quad \underline{\vee} & \\
\mathrm{rank}_+(M) & & & \tau_+^{\mathrm{sos}}(M). &
\end{array}$$

5.3. Numerical results and examples

In this section, we test the ideal-sparse and dense hierarchies on two classes of nonnegative matrices. The first class consists of size 4 × 4 matrices that depend continuously on a single variable. The second class we consider is the Euclidean distance matrices (EDMs).

5.3.1. Matrices related to the nested rectangles problem. The nonnegative matrices we will consider have an interesting link between their nonnegative rank and the geometric nested rectangles problem (see [**25**]). Bounds for their nonnegative rank were investigated by Fawzi and Parrilo [**67**] and Gribling et al. [**80**]. Consider the matrices

$$S(a,b) := \begin{pmatrix} 1-a & 1+a & 1-b & 1+b \\ 1+a & 1-a & 1-b & 1+b \\ 1+a & 1-a & 1+b & 1-b \\ 1-a & 1+a & 1+b & 1-b \end{pmatrix} \qquad \text{for } 0 \le a, b \le 1.$$

If $a, b < 1$ then $S(a,b)$ is fully dense and no improvement can be expected from our new bounds. Thus we consider the case $b = 1$ and $0 \le a \le 1$. We have computed the bounds $\xi^{+}_{t,\ddagger}(M)$ and $\xi^{+,\mathrm{isp}}_{t,\ddagger}(M)$ at level $t = 1, 2, 3$ for $M = S(a,1)$ with a ranging from 0 to 1 in increments of 0.01. The results are displayed in Figure 1 below. We can make the following two observations about Figure 1. First, the ideal-sparse hierarchy is much stronger at level $t = 1$, but at level $t = 2$ the dense and ideal-sparse hierarchies give comparable bounds. Second, for $a = 1$, all bounds (except the dense bound of level 1) are equal to $4 = \mathrm{rank}_+(S(1,1))$ (as is expected for the ideal-sparse hierarchy given Lemma 5.2).

Bounds $\xi^{+}_{t,\ddagger}(M)$ and $\xi^{+,\mathrm{isp}}_{t,\ddagger}(M)$ for $M = S(a,1)$ and $t = 1, 2, 3$ vs. $0 \le a \le 1$

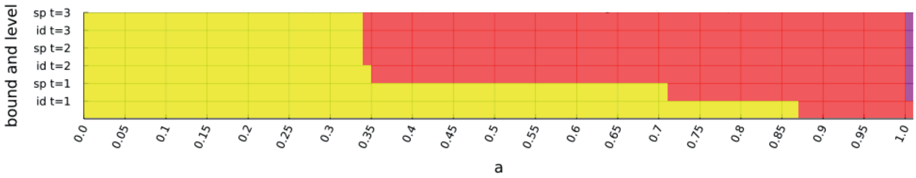


FIGURE 1. This figure shows $\xi^{+}_{t,\dagger}(S(a,1))$ and $\xi^{+,\mathrm{isp}}_{t,\dagger}(S(a,1))$ computed at levels $t = 1, 2, 3$ with a ranging from 0 to 1 in increments of 0.01. The color indicates a lower bound on the obtained numerical value: yellow, red, and purple show the bound is at least 2, 3, and 4, respectively. So a red square at $a = 0.35$ and "sp t=2" means $\xi^{+,\mathrm{isp}}_{2,\dagger}(M) \ge 3$.

5.3.2. Euclidean distance matrices. The second class of examples we consider are the *Euclidean distance matrices* $M_n = ((i-i)^2)_{i,j=1}^{n} \in \mathbb{R}^{n\times n}_{+}$, known to have a large separation between their rank (in the linear algebra sense) and their nonnegative rank. Indeed, $\mathrm{rank}(M_n) = 3$, see [**14**], and their bipartite support graph G^{M_n} is $K_{n,n}$ with a deleted perfect matching (known as a *crown graph*), whose edge biclique-cover number satisfies $\mathrm{bc}(G^{M_n}) = \Theta(\log n)$ [**52**]. So we have $\mathrm{rank}(M_n) = 3$ and $\mathrm{rank}_+(M_n) \ge \mathrm{bc}(G^{M_n}) = \Theta(\log n)$. In addition, it is known that $\mathrm{rank}_+(M_n) \le 2 + \lceil \frac{n}{2} \rceil$, see [**77**, Theorem 9]. The numerical results are shown in Table 1. In these examples, the ideal-sparse bound of level $t = 2$ is more difficult to compute since the support graph G^{M_n} has 2^{n-1} maximal bicliques, each with n vertices. For this reason, we could compute $\xi^{+,\mathrm{isp}}_{2,\dagger}$ only until $n = 7$ before running out of memory. So this example illustrates the limitations of the ideal sparsity approach when the number of maximal cliques is too large. Note that this difficulty (of large numbers of maximal bicliques) remains even if we would replace the support graph G^{M_n} with a supergraph $\widetilde{G}$, obtained by adding to G^{M_n} (say) s edges from the missing perfect matching. Indeed, such $\widetilde{G}$ still has 2^{n-s-1} maximal bicliques, each with $n+s$ vertices.

TABLE 1. Bounds for the matrices $M_n = ((i-j)^2)_{i,j=1}^n$.

n	bc	$\xi^+_{1,\dagger}$	$\xi^+_{2,\dagger}$	$\xi^{+,\mathrm{isp}}_{1,\dagger}$	$\xi^{+,\mathrm{isp}}_{2,\dagger}$	$2+\lceil\frac{n}{2}\rceil$
4	4	2	3.46	3	3.63	4
5	4	2	3.73	3.35	4.19	5
6	4	2	3.96	3.41	4.53	5
7	5	2	4.17	3.55	4.85	6
8	5	2	4.35	3.59	-	6
9	5	2	4.51	3.66	-	7

CHAPTER 6

Completely positive rank

This chapter focuses on the completely positive rank of a matrix. We begin with a quick recap of definitions. Then, we apply the results of Chapter 3 to build a hierarchy of lower bounds for the completely positive rank (Section 6.1). We explore the progressive improvements over time that were made in order to create stronger hierarchies.

Our two new contributions to this field are the strengthening of the hierarchy by adding a polynomial matrix localizing constraint (recall Section 1.3), and the exploitation of ideal sparsity (recall Section 2.2) to build a possibly stronger and easier-to-compute ideal-sparse hierarchy (Section 6.1.3). We also explore a weak-ideal-sparse hierarchy that sacrifices some bound strength for even faster computation.

Having defined several hierarchies, we compare them to each other and other known combinatorial bounds from the literature in Section 6.2. Lastly, we present our numerical results comparing the different hierarchies in Section 6.3.

The cone of completely positive matrices. Recall that, for a given integer $n \in \mathbb{N}$, the *cone of completely positive $n \times n$ matrices* is defined as

$$\mathcal{CP}_n := \text{cone}\{\mathbf{x}\mathbf{x}^T : \mathbf{x} \in \mathbb{R}^n_+\}.$$

The cone of completely positive matrices and its dual, the *cone of copositive matrices*, are well-known for their expressive power in modeling optimization problems. For example, many NP-hard problems can be formulated as linear optimization problems over these cones [**43, 28**]. Checking whether a given matrix A is completely positive is itself a computational hard problem (see [**53**]). The reader may be tempted to think that $\mathcal{CP}_n$ is characterized by entrywise nonnegativity and PSDness. This has been shown to hold for the particular setting of $n \in \{1, 2, 3, 4\}$ (see [**18**]), but this is not the case in general, as was shown in Example 4.2.

The moment approach has been applied to test whether $A \in \mathcal{CP}_n$ and to find a CP factorization (see (6.1)), in particular, by Nie [**126**], who formulates it as testing the existence of a representing measure (over the standard simplex) for the sequence of entries of A.

We refer to the monograph [**18**] for a deeper insight into the structural properties of the cone $\mathcal{CP}_n$.

The completely positive rank. Given a matrix $A \in \mathcal{CP}_n$, one can ask what is the smallest integer $r \in \mathbb{N}$ such that A admits a decomposition of the form

$$A = \sum_{\ell \in [r]} \mathbf{a}_\ell \mathbf{a}_\ell^T, \; \mathbf{a}_\ell \in \mathbb{R}^n_+. \tag{6.1}$$

The smallest such r is called the *completely positive rank* of A and is defined as

$$\mathrm{rank}_{\mathrm{cp}}(A) := \min \Big\{ r \in \mathbb{N} : A = \sum_{\ell \in [r]} \mathbf{a}_\ell \mathbf{a}_\ell^T, \; \mathbf{a}_\ell \in \mathbb{R}^n_+ \Big\}. \tag{6.2}$$

Because this definition only holds for completely positive matrices, we assign to all other matrices a CP rank of ∞. No efficient algorithms are known for exactly finding the CP rank. Thus we are motivated to search for efficient methods of approximating the CP rank. In particular, there is an interest in finding lower bounds on the CP rank, as, e.g., in [**67, 80, 81**].

The lower bound $\tau_{\mathrm{cp}}(A)$. In [**67**], Fawzi and Parrilo defined the parameter

$$\tau_{\mathrm{cp}}(A) := \inf \Big\{ \lambda > 0 : \frac{1}{\lambda} A \in \mathrm{conv}\{\mathbf{x}\mathbf{x}^T : \mathbf{x} \in \mathbb{R}^n_+, \; \mathbf{x}\mathbf{x}^T \leq A, \; \mathbf{x}\mathbf{x}^T \preceq A\} \Big\}. \tag{6.3}$$

This parameter can be seen as a natural "convexification" of the completely positive rank, and it satisfies

$$\tau_{\mathrm{cp}}(A) \leq \mathrm{rank}_{\mathrm{cp}}(A). \tag{6.4}$$

As a quick argument, observe that any given CP factorization like $A = \sum_{\ell \in [r]} \mathbf{a}_\ell \mathbf{a}_\ell^T$ induces a solution $\lambda = r$ to (6.3). This is because we can write $\frac{1}{r}A$ as the following convex combination:

$$\frac{1}{r} A = \sum_{\ell \in [r]} \frac{1}{r} \mathbf{a}_\ell \mathbf{a}_\ell^T,$$

with $\mathbf{a}_\ell \in \mathbb{R}^n_+$, $\mathbf{a}_\ell \mathbf{a}_\ell^T \preceq A$, and $\mathbf{a}_\ell \mathbf{a}_\ell^T \leq A$ for each $\ell \in [r]$.

As presented here, the parameter $\tau_{\mathrm{cp}}(A)$ does not immediately seem easy to compute. Hence, we look to lower bound $\tau_{\mathrm{cp}}(A)$. Fawzi and Parrilo also introduced in [**67**] the SDP-based lower bound $\tau^{\mathrm{sos}}_{\mathrm{cp}}(A) \leq \tau_{\mathrm{cp}}(A)$. We do not elaborate further on $\tau^{\mathrm{sos}}_{\mathrm{cp}}(A)$ here, but we will relate it to forthcoming parameters when appropriate.

The parameter $\tau_{\mathrm{cp}}(A)$ can be reformulated as an instance of a GMP of the form (2.1) from Chapter 2. To state the GMP in question, we must first establish some preliminaries. Then we give the result in Lemma 6.2.

To avoid trivialities, assume there are no zeros on the diagonal, i.e., $A_{ii} > 0$ for all $i \in [n]$. Indeed, if A is a CP matrix with $A_{ii} = 0$, then its i^{th} row/column is identically zero, and thus it can be removed without altering the CP rank.

We define the *support graph* $G_A := (V := [n], E_A)$ of A, with edge set and non-edge set respectively given by

$$\begin{aligned} E_A &:= \Big\{ \{i,j\} : A_{ij} \neq 0, \, i,j \in V, \, i \neq j \Big\}, \\ \overline{E}_A &:= \Big\{ \{i,j\} : A_{ij} = 0, \, i,j \in V, \, i \neq j \Big\}. \end{aligned} \tag{6.5}$$

Defining a semi-algebraic set. Using the edges and non-edges, we can define the following semialgebraic set

$$\begin{aligned} K_A := \Big\{\mathbf{x} \in \mathbb{R}^n : \sqrt{A_{ii}} x_i - x_i^2 &\geq 0 \ (i \in [n]), && \text{(6.6a)} \\ A_{ij} - x_i x_j &\geq 0 \ (\{i,j\} \in E_A), && \text{(6.6b)} \\ x_i x_j &= 0 \ (\{i,j\} \in \overline{E}_A), && \text{(6.6c)} \\ A - \mathbf{x}\mathbf{x}^T &\succeq 0 \Big\}. && \text{(6.6d)} \end{aligned}$$

Alternatively, following [**80**], we could have defined the set K_A as

$$\begin{aligned} K_A = \Big\{\mathbf{x} \in \mathbb{R}^n : & \ x_i \geq 0 \ (i \in [n]), \\ & A_{ij} - x_i x_j \geq 0 \ (i, j \in [n]), \\ & A - \mathbf{x}\mathbf{x}^T \succeq 0 \Big\}, \end{aligned}$$

which is more closely modeled on the definition of $\tau_{\mathrm{cp}}(A)$ in (6.3). We now explain why we will adopt the particular algebraic description (6.6a) - (6.6d) for the set K_A.

As was observed in [**80**], the constraints $A \geq \mathbf{x}\mathbf{x}^T$ and $\mathbf{x} \geq 0$ are equivalent to $\sqrt{A_{ii}} x_i - x_i^2 \geq 0$ $(i \in [n])$, $A_{ij} - x_i x_j \geq 0$ $(\{i,j\} \in E_A)$, and $x_i x_j = 0$ $(\{i,j\} \in \overline{E}_A)$. However, the associated truncated quadratic modules (defined in (2.18)) are not. Indeed, for each $t \in \mathbb{N}$, we have

$$\mathcal{M}_{2t}(\widetilde{H}) \subseteq \mathcal{M}_{2t}(H),$$

where

$$\begin{aligned} \widetilde{H} &:= \Big\{ x_i \ (i \in [n]), \ A_{ij} - x_i x_j \ (i, j \in [n]) \Big\}, \\ H &:= \Big\{ \sqrt{A_{ii}} x_i - x_i^2 \ (i \in [n]), \ A_{ij} - x_i x_j \ (\{i,j\} \in E_A), \ \pm x_i x_j \ (\{i,j\} \in \overline{E}_A) \Big\}. \end{aligned}$$

The inclusion follows from the following two polynomial identities:

$$\begin{aligned} A_{ii} - x_i^2 &= (\sqrt{A_{ii}} - x_i)^2 + 2(\sqrt{A_{ii}} x_i - x_i^2) \ (i \in [n]), && \text{(6.8)} \\ x_i &= ((\sqrt{A_{ii}} x_i - x_i^2) + x_i^2)/\sqrt{A_{ii}} \ (i \in [n]). \end{aligned}$$

Remark 6.1. *Observe that (6.8) implies that*

$$\mathrm{Tr}(A) - \sum_{i \in [n]} x_i^2 \in \mathcal{M}_2(H).$$

In particular, we conclude that any quadratic module that contains H in its generators is Archimedean.

Observe how the ideal constraints (6.6c) are of a form that ideal sparsity can be exploited (see Section 2.2). The ideal-sparse structure is thusly inherited from the sparsity of the matrix A. Later, in equation (6.20), we will use this fact to define a stronger and possibly faster hierarchy of lower bounds.

GMP formulation of the parameter $\tau_{\mathrm{cp}}(A)$**.** We now show that the parameter $\tau_{\mathrm{cp}}(A)$ is the optimal value of a GMP of the form (3.5).

LEMMA 6.2. *The parameter* $\tau_{\mathrm{cp}}(A)$ *is equal to the optimal value of the following generalized moment problem:*

$$\mathbf{val}_{\mathrm{cp}}(A) := \inf_{\mu \in \mathscr{M}(K_A)} \Big\{ \int 1 d\mu : \int x_i x_j d\mu = A_{ij} \ (i, j \in V) \Big\}. \tag{6.9}$$

PROOF. **(**$\mathbf{val}_{\mathrm{cp}}(A) \le \tau_{\mathrm{cp}}(A)$**)** Any feasible solution to $\tau_{\mathrm{cp}}(A)$, i.e., any decomposition of the form $A = \lambda \sum_{\ell \in [r]} c_\ell \mathbf{a}_\ell \mathbf{a}_\ell^T$, with $\lambda > 0$, $\sum_{\ell \in [r]} c_\ell = 1$, $c_\ell > 0$, $\mathbf{a}_\ell \ge 0$, $A \ge \mathbf{a}_\ell \mathbf{a}_\ell^T$, and $A \succeq \mathbf{a}_\ell \mathbf{a}_\ell^T$, corresponds to a finite atomic measure

$$\mu = \lambda \sum_{\ell \in [r]} c_\ell \delta_{\mathbf{a}_\ell}$$

that is feasible for $\mathbf{val}_{\mathrm{cp}}(A)$, with objective value λ. Hence, $\mathbf{val}_{\mathrm{cp}}(A) \le \tau_{\mathrm{cp}}(A)$.

($\mathbf{val}_{\mathrm{cp}}(A) \ge \tau_{\mathrm{cp}}(A)$**)** Assume the GMP (6.9) is feasible, else $\mathbf{val}_{\mathrm{cp}}(A) = \infty$ and we have nothing to prove. Let $\mu \in \mathscr{M}(K_A)$ be a feasible solution to (6.9). In view of Theorem 2.8 (ii), we may assume that μ is a finite atomic measure, i.e.,

$$\mu = \lambda \sum_{\ell \in [r]} c_\ell \delta_{\mathbf{a}_\ell},$$

with $\lambda > 0$, $\sum_{\ell \in [r]} c_\ell = 1$, $c_\ell > 0$, and $\mathbf{a}_\ell \in K_A$. This measure then induces a decomposition

$$A = \lambda \sum_{\ell \in [r]} c_\ell \mathbf{a}_\ell \mathbf{a}_\ell^T$$

corresponding to a solution to (6.3), with value λ. Hence, $\mathbf{val}_{\mathrm{cp}}(A) \ge \tau_{\mathrm{cp}}(A)$. □

6.1. Hierarchies of lower bounds for the completely positive rank

With $\tau_{\mathrm{cp}}(A)$ recast as a GMP instance (6.9), we now apply the tools from Chapter 3 to create a hierarchy of SDP programs, each lower bounding $\tau_{\mathrm{cp}}(A)$, and with asymptotic convergence to $\tau_{\mathrm{cp}}(A)$. Over time, several hierarchies have been developed, each improving on the last in some way. We first present an initial hierarchy from [**80**] and then three subsequent improved hierarchies based on our work [**81, 100**]. The running theme is that each improvement is attained by encoding another property of CP factorizations into the hierarchy.

6.1.1. The first hierarchy of lower bounds for $\tau_{\mathrm{cp}}(A)$**.** In [**80**], the authors derived a hierarchy of SDP bounds for the CP rank of a CP matrix $A \in \mathbb{R}_+^{n \times n}$, which we denote here, for any $t \in \mathbb{N} \cup \{\infty\}$, as

$$
\begin{aligned}
\xi_t^{\mathrm{cp},(2019)}(A) := \inf \Big\{ L(1) :\ & \\
& L \in \mathbb{R}[\mathbf{x}]_{2t}^*, \\
& L(\mathbf{x}\mathbf{x}^T) = A, && \text{(6.10a)} \\
& L([\mathbf{x}]_t[\mathbf{x}]_t^T) \succeq 0, && \text{(6.10b)} \\
& L((\sqrt{A_{ii}}x_i - x_i^2)[\mathbf{x}]_{t-1}[\mathbf{x}]_{t-1}^T) \succeq 0 \ (i \in V), && \text{(6.10c)} \\
& L((A_{ij} - x_i x_j)[\mathbf{x}]_{t-1}[\mathbf{x}]_{t-1}^T) \succeq 0 \ (\{i,j\} \in E_A \cup \overline{E}_A), && \text{(6.10d)} \\
& L((\mathbf{x}\mathbf{x}^T)^{\otimes \ell}) \preceq A^{\otimes \ell} \ (\ell \in [t]) \Big\}. && \text{(6.10e)}
\end{aligned}
$$

These constraints warrant some explanation. The constraint (6.10a) comes directly from the GMP (6.9), and the matrix (localizing) constraints (6.10b), (6.10c), and (6.10d) come from the definition of the semialgebraic domain K_A in (6.6). Note that this hierarchy does not take the zero entries in the matrix A into special consideration. As a result, there are no explicit ideal constraints. This will be important later when we use the sparsity in the matrix A to create an improved hierarchy. The last constraint (6.10e) is the defining feature of $\xi_t^{\mathrm{cp},(2019)}(A)$, and is motivated by the following argument.

If $\mathbf{x}\mathbf{x}^T \preceq A$ (as is imposed by the constraint (6.6d) in the definition of K_A), then it must follow that $(\mathbf{x}\mathbf{x}^T)^{\otimes \ell} \preceq A^{\otimes \ell}$ for all $\ell \in \mathbb{N}$. Via an argument in the proof of [**80**, Proposition 6] it can be shown that

$$L((\mathbf{x}\mathbf{x}^T)^{\otimes \ell}) \preceq A^{\otimes \ell} \ (\ell \in \mathbb{N}) \tag{6.11}$$

is valid for all linear functionals arising from an atomic decomposition as in the definition (6.3) of $\tau_{\mathrm{cp}}(A)$. Gribling et al. [**80**] showed asymptotic convergence, i.e,

$$\lim_{t\to\infty} \xi_t^{\mathrm{cp},(2019)}(A) = \tau_{\mathrm{cp}}(A).$$

In [**80**], the authors showed that the same convergence result holds if we replace constraint (6.10e) with

$$L(\mathbf{v}^T(A - \mathbf{x}\mathbf{x}^T)\mathbf{v}[\mathbf{x}]_{t-1}[\mathbf{x}]_{t-1}^T) \succeq 0 \ (\mathbf{v} \in \mathbb{R}^n). \tag{6.12}$$

The core tool underlying these convergence results is Theorem 3.3, where assumption (A) is satisfied because of (6.4), assumption (B) because of Remark 6.1, and assumption (C) holds by taking $z_{i,j} = 0$ $(i, j \in [n])$ and $c = \frac{1}{2}$. We restate the conclusion of Theorem 3.3 in the completely positive rank setting for completeness.

For each $t \in \mathbb{N} \cup \{\infty\}$, the program (6.10) attains it optimum, and

$$\lim_{t\to\infty} \xi_t^{\mathrm{cp},(2019)}(A) = \xi_\infty^{\mathrm{cp},(2019)}(A) = \tau_{\mathrm{cp}}(A).$$

Moreover, the GMP (6.2) has an optimal solution μ that is finite atomic and is supported on K_A.

Note that this result holds analogously for the subsequently improved hierarchies (e.g., (6.13) and (6.15)) that we consider later in this chapter. When we consider flatness in Section 6.3.3, the existence of optimal solutions for the hierarchies will be necessary.

Unfortunately, we could find no results comparing the constraints (6.10e) and (6.12). As such, we cannot make any claims about which is better regarding the bounds of their associated hierarchies. However, there are clear distinctions between (6.10e) and (6.12) when it comes to the number of constraints and the sizes of the involved matrices. The constraint (6.10e) requires t-many PSD constraints, the largest of which contains a matrix of size n^t. On the other hand, the constraint (6.12) involves matrices of size $\binom{n+t-1}{t-1}$, but there are infinitely many of them because $\mathbf{v}$ ranges over all of $\mathbb{R}^n$. By invariance to scaling, one can take $\mathbf{v} \in \mathbb{S}^{n-1}$ (the unit sphere in $\mathbb{R}^n$). The authors of [**80**] considered using (6.12) with the vectors $\mathbf{v}$ restricted to some finite set $T \subset \mathbb{S}^{n-1}$. Doing so, one can partially involve the constraints (6.12) in computations. Next, we define a new hierarchy that replaces (6.10e) with a constraint stronger than (6.12), which involves only one PSD matrix of size $\binom{n+t-1}{t-1} \cdot n$.

6.1.2. A polynomial matrix localizing constraint hierarchy for $\tau_{\mathrm{cp}}(A)$. Consider, for any $t \in \mathbb{N} \cup \{\infty\}$, the parameter

$$\begin{aligned}\xi_t^{\mathrm{cp},(2022)}(A) := \inf \Big\{ & L(1) : \\ & L \in \mathbb{R}[\mathbf{x}]_{2t}^*, \\ & L(\mathbf{x}\mathbf{x}^T) = A, \\ & L([\mathbf{x}]_t[\mathbf{x}]_t^T) \succeq 0, \\ & L((\sqrt{A_{ii}}x_i - x_i^2)[\mathbf{x}]_{t-1}[\mathbf{x}]_{t-1}^T) \succeq 0 \ (i \in V), \\ & L((A_{ij} - x_ix_j)[\mathbf{x}]_{t-1}[\mathbf{x}]_{t-1}^T) \succeq 0 \ (\{i,j\} \in E_A \cup \overline{E}_A), \\ & L((A - \mathbf{x}\mathbf{x}^T) \otimes [\mathbf{x}]_{t-1}[\mathbf{x}]_{t-1}^T) \succeq 0 \Big\}. \end{aligned} \tag{6.13a}$$

Observe that via Corollary 1.6, the constraints (6.12) are implied by the stronger constraints (6.13a). We now show (see Lemma 6.3 below) that the polynomial matrix localizing constraint (6.13a) implies the tensor positivity constraint in (6.10e). As a result, it follows that

$$\xi_t^{\mathrm{cp},(2019)}(A) \leq \xi_t^{\mathrm{cp},(2022)}(A) \text{ for all } t \in \mathbb{N} \cup \{\infty\}.$$

We further substantiate this theoretical result with numerical examples later in Table 1.

Lemma 6.3. [**81**, Lemma 19] *Consider $t \in \mathbb{N}$, $A \in \mathbb{R}_+^{n\times n}$ and $L \in \mathbb{R}[\mathbf{x}]_{2t}^*$. If $L(\mathbf{x}\mathbf{x}^T) = A$ and $L((A - \mathbf{x}\mathbf{x}^T) \otimes [\mathbf{x}]_{t-1}[\mathbf{x}]_{t-1}^T) \succeq 0$, then $L((\mathbf{x}\mathbf{x}^T)^{\otimes \ell}) \preceq A^{\otimes \ell}$ ($\ell \in [t]$).*

Proof. Observe that

$$L((A - \mathbf{x}\mathbf{x}^T) \otimes [\mathbf{x}]_{t-1}[\mathbf{x}]_{t-1}^T) \succeq 0 \iff L((A - \mathbf{x}\mathbf{x}^T) \otimes \langle \mathbf{x}\rangle_{t-1}\langle \mathbf{x}\rangle_{t-1}^T) \succeq 0, \tag{6.14}$$

where $\langle \mathbf{x}\rangle_{t-1}$ denotes the vector of noncommutative monomials of degree less than t in the variables $x_1, \ldots, x_n$. This equivalence holds because the latter matrix is obtained by duplicating rows/columns of the former matrix.

Note that, for each $\ell \in [t]$, $L((A - \mathbf{x}\mathbf{x}^T) \otimes \langle \mathbf{x}\rangle_{t-1}\langle \mathbf{x}\rangle_{t-1}^T)$ contains the matrix $L((A - \mathbf{x}\mathbf{x}^T) \otimes (\mathbf{x}\mathbf{x}^T)^{\otimes(\ell-1)})$ as a principal submatrix. To see this, observe that, for each $\ell \in \mathbb{N}$, $(\mathbf{x}\mathbf{x}^T)^{\otimes \ell} = (\mathbf{x}^{\otimes \ell})(\mathbf{x}^{\otimes \ell})^T$ is a principal submatrix of $\langle \mathbf{x}\rangle_{=\ell}\langle \mathbf{x}\rangle_{=\ell}^T$ (where $\langle \mathbf{x}\rangle_{=\ell}$ are all the noncommutative monomials with degree exactly ℓ), because all the entries of $\mathbf{x}^{\otimes \ell}$ are contained in $\langle \mathbf{x}\rangle_{=\ell}$.

Since $L((A-\mathbf{x}\mathbf{x}^T)\otimes\langle\mathbf{x}\rangle_{t-1}\langle\mathbf{x}\rangle_{t-1}^T) \succeq 0$, we obtain $L((A-\mathbf{x}\mathbf{x}^T)\otimes(\mathbf{x}\mathbf{x}^T)^{\otimes(\ell-1)}) \succeq 0$, and thus $L((\mathbf{x}\mathbf{x}^T)^{\otimes\ell}) \preceq A\otimes L((\mathbf{x}\mathbf{x}^T)^{\otimes(\ell-1)})$ for all $\ell \in [t]$. Combined with $L(\mathbf{x}\mathbf{x}^T) = A$ this permits us to show:

$$L((\mathbf{x}\mathbf{x}^T)^{\otimes\ell}) \preceq A \otimes L((\mathbf{x}\mathbf{x}^T)^{\otimes(\ell-1)} \preceq \cdots \preceq A^{\otimes(\ell-1)} \otimes L(\mathbf{x}\mathbf{x}^T) = A^{\otimes\ell}. \qquad \square$$

Adding ideal constraints to the hierarchy. We now use the observation that $\mathbf{x} \in K_A$ satisfies $x_ix_j = 0$ for all $\{i,j\} \in \overline{E}_A$, to further improve upon the hierarchy in (6.13). For $t \in \mathbb{N}\cup\{\infty\}$ consider the following SDP:

$$\begin{aligned}\xi_t^{\mathrm{cp}}(A) := \min \Big\{ L(1) : L \in \mathbb{R}[\mathbf{x}]_{2t}^*,\\ & L(\mathbf{x}\mathbf{x}^T) = A,\\ & L([\mathbf{x}]_t[\mathbf{x}]_t^T) \succeq 0,\\ & L((\sqrt{A_{ii}}x_i - x_i^2)[\mathbf{x}]_{t-1}[\mathbf{x}]_{t-1}^T) \succeq 0 \ (i \in V),\\ & L((A_{ij} - x_ix_j)[\mathbf{x}]_{t-1}[\mathbf{x}]_{t-1}^T) \succeq 0 \ (\{i,j\} \in E_A), && (6.15a)\\ & L(x_ix_j[\mathbf{x}]_{2t-2}) = 0 \ (\{i,j\} \in \overline{E}_A), && (6.15b)\\ & L((A - \mathbf{x}\mathbf{x}^T) \otimes [\mathbf{x}]_{t-1}[\mathbf{x}]_{t-1}^T) \succeq 0 \Big\}.\end{aligned}$$

This hierarchy now explicitly uses ideal constraints induced by zero entries in the matrix A. Hence, the constraints (6.10d), which run over all off-diagonal entries of the matrix A, are now split into two constraints (6.15a) and (6.15b). Thus, it clearly follows that

$$\xi_t^{\mathrm{cp},(2022)}(A) \le \xi_t^{\mathrm{cp}}(A) \text{ for all } t \in \mathbb{N}\cup\{\infty\}.$$

Adding scalar localizing constraints based on nonnegativity. Exploiting the fact that the variables x_i should be nonnegative, one may add localizing constraints like the following:

$$L([\mathbf{x}]_{2t}) \ge 0, \qquad (6.16)$$

$$L((\sqrt{A_{ii}}x_i - x_i^2)[\mathbf{x}]_{2t-2}) \ge 0 \ (i \in V), \qquad (6.17)$$

$$L(A_{ij} - x_ix_j)[\mathbf{x}]_{2t-2}) \ge 0 \ (\{i,j\} \in E_A), \qquad (6.18)$$

$$L(x_ix_j[\mathbf{x}]_{t-1}[\mathbf{x}]_{t-1}^T) \succeq 0 \ (\{i,j\} \in E_A). \qquad (6.19)$$

Note that the constraints (6.19) are redundant at the smallest level $t = 1$. One could add a similar constraint to (6.19) by replacing x_ix_j with any monomial. We use the notation $\xi_{t,\dagger}^{\mathrm{cp}}(A)$ to denote the parameter obtained by adding (6.18) to the program defining $\xi_t^{\mathrm{cp}}(A)$. Define analogously $\xi_{t,\dagger}^{\mathrm{cp},(2019)}(A)$ by adding (6.18) to $\xi_t^{\mathrm{cp},(2019)}(A)$, so that we have

$$\xi_{t,\dagger}^{\mathrm{cp},(2019)}(A) \le \xi_{t,\dagger}^{\mathrm{cp}}(A).$$

As we will see in relation (6.27) below, the bound $\xi_{2,\dagger}^{\mathrm{cp},(2019)}(A)$ is at least as good as $\mathrm{rank}(A)$, which is an obvious lower bound on $\mathrm{rank}_{\mathrm{cp}}(A)$. Let $\xi_{t,\ddagger}^{\mathrm{cp}}(A)$ denote the further strengthening of $\xi_{t,\dagger}^{\mathrm{cp}}(A)$ by adding constraints (6.16), (6.17), and (6.19), so that we have

$$\xi_t^{\mathrm{cp}}(A) \le \xi_{t,\dagger}^{\mathrm{cp}}(A) \le \xi_{t,\ddagger}^{\mathrm{cp}}(A) \text{ for any } t \in \mathbb{N}\cup\{\infty\}.$$

6.1.3. An ideal-sparse hierarchy of lower bounds for $\tau_{\rm cp}(A)$. The ideal constraints (6.15b) are of a form susceptible to the technique of ideal sparsity described in Section 3.1.4. Hence, we follow the approach described in Section 3.1.4 to create a new ideal-sparse hierarchy for the parameter $\tau_{\rm cp}(A)$, as characterized by the GMP (6.9). The process will be similar to what was done for the nonnegative rank in Section 5.1.2.

Begin by considering the support graph $G_A := (V = [n], E_A)$ of the matrix A, and let $V_1, ..., V_p$ denote all the maximal cliques of the graph G_A. For $t \in \mathbb{N} \cup \{\infty\}$, define the *ideal-sparse moment bounds*:

$$\xi_t^{\rm cp,isp}(A) :=$$

$$\min\Big\{ \sum_{k\in[p]} L_k(1) : \ L_k \in \mathbb{R}[\mathbf{x}(V_k)]_{2t}^* \ (k \in [p]),$$

$$\sum_{k\in[p]:i,j\in V_k} L_k(x_ix_j) = A_{ij} \ (i,j \in V), \tag{6.20a}$$

$$L_k([\mathbf{x}(V_k)]_t[\mathbf{x}(V_k)]_t^T) \succeq 0 \ (k \in [p]),$$

$$L_k((\sqrt{A_{ii}}x_i - x_i^2)[\mathbf{x}(V_k)]_{t-1}[\mathbf{x}(V_k)]_{t-1}^T) \succeq 0 \ (i \in V_k, \ k \in [p]),$$

$$L_k((A_{ij} - x_ix_j)[\mathbf{x}(V_k)]_{t-1}[\mathbf{x}(V_k)]_{t-1}^T) \succeq 0 \ (i \neq j \in V_k, \ k \in [p]), \tag{6.20b}$$

$$L_k((A - \mathbf{x}\mathbf{x}^T)_{|V_k} \otimes [\mathbf{x}(V_k)]_{t-1}[\mathbf{x}(V_k)]_{t-1}^T) \succeq 0 \ (k \in [p]). \tag{6.20c}$$

Here, in equation (6.20c), it is understood that, for a given $k \in [p]$, in the matrix $A - \mathbf{x}\mathbf{x}^T$ one sets the entries of $\mathbf{x}$ indexed by $V \setminus V_k$ to zero.

Analogous to the constraints (6.16), (6.17), (6.18) and (6.19) in the dense hierarchy, we can add the following constraints that exploit the nonnegativity of the variables:

$$L_k([\mathbf{x}(V_k)]_{2t}) \geq 0 \ (k \in [p]), \tag{6.21}$$

$$L_k((\sqrt{A_{ii}}x_i - x_i^2)[\mathbf{x}(V_k)]_{2t-2}) \geq 0 \ (i \in V_k, \ k \in [p]), \tag{6.22}$$

$$L_k((A_{ij} - x_ix_j)[\mathbf{x}(V_k)]_{2t-2}) \geq 0 \ (\{i,j\} \subseteq V_k, \ k \in [p]), \tag{6.23}$$

$$L_k(x_ix_j[\mathbf{x}(V_k)]_{t-1}[\mathbf{x}(V_k)]_{t-1}^T) \succeq 0 \ (i \neq j \in V_k, \ k \in [p]). \tag{6.24}$$

Define $\xi_{t,\dagger}^{\rm cp,isp}(A)$ by adding constraint (6.23) to $\xi_t^{\rm cp,isp}(A)$, and $\xi_{t,\ddagger}^{\rm cp,isp}(A)$ by adding the constraints (6.21), (6.22) and (6.24) to $\xi_{t,\dagger}^{\rm cp,isp}(A)$, so that

$$\xi_t^{\rm cp,isp}(A) \leq \xi_{t,\dagger}^{\rm cp,isp}(A) \leq \xi_{t,\ddagger}^{\rm cp,isp}(A).$$

Weak-ideal-sparse hierarchies for $\tau_{\rm cp}(A)$. Observe that, if, in equation (6.20c), we replace the matrix $A - \mathbf{x}\mathbf{x}^T$ by its principal submatrix indexed by V_k, then one also gets a lower bound on $\tau_{\rm cp}(A)$, possibly weaker than $\xi_t^{\rm cp,isp}(A)$, but potentially easier to compute. We let $\xi_t^{\rm cp,wisp}(A)$ denote the parameter obtained in this way by replacing, in the definition of $\xi_t^{\rm cp,isp}(A)$, equation (6.20c) by

$$L_k((A[V_k] - \mathbf{x}(V_k)\mathbf{x}(V_k)^T) \otimes [\mathbf{x}(V_k)]_{t-1}[\mathbf{x}(V_k)]_{t-1}^T) \succeq 0 \ (k \in [p]), \tag{6.25}$$

so that we have

$$\xi_t^{\mathrm{cp,wisp}}(A) \leq \xi_t^{\mathrm{cp,isp}}(A).$$

Note that by relaxing constraint (6.20c) to (6.25), we can no longer claim that $\xi_t^{\mathrm{cp,isp}}(A)$ is at least as good as the dense hierarchy $\xi_t^{\mathrm{cp}}(A)$. In some numerical instances, it is strictly worse. Indeed, in our numerical experiments, we frequently observe the strict inequality $\xi_t^{\mathrm{cp,wisp}}(A) < \xi_t^{\mathrm{cp}}(A)$ for randomly generated matrices A (see Section 6.3.1 for details). For example, the matrix (with entries rounded for presentation)

$$A = \begin{pmatrix} 1.0 & 0.578 & 0.0 & 0.0 & 0.225 \\ 0.578 & 1.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.0 & 0.0 & 0.656 \\ 0.0 & 0.0 & 0.0 & 1.0 & 0.526 \\ 0.225 & 0.0 & 0.656 & 0.526 & 1.0 \end{pmatrix}$$

has the following parameters at level $t = 2$:

$$\left(\xi_2^{\mathrm{cp,wisp}}(A) = 4\right) < \left(\xi_2^{\mathrm{cp}}(A) = 5\right) \leq \left(\xi_2^{\mathrm{cp,isp}}(A) = 5\right) \leq \left(\mathrm{rank}_{\mathrm{cp}}(A) = 5\right).$$

6.2. Links to other lower bounds on the completely positive rank

Here, we indicate links to other known lower bounds on the CP rank. Clearly, the rank is a lower bound:

$$\mathrm{rank}(A) \leq \mathrm{rank}_{\mathrm{cp}}(A).$$

Edge clique-cover bound. A combinatorial lower bound arises naturally from the edge clique-cover number of the support graph G_A. Given a graph $G = (V, E)$, its *edge clique-cover number*, denoted $\mathrm{c}(G)$ (following [**67**]), is defined as the smallest number of (maximal) cliques in G whose union covers every edge of G. This parameter is NP-hard to compute [**70**]. Clearly, $\mathrm{c}(G) = |E|$ if G is a triangle-free graph (i.e., $\omega(G) = 2$, where $\omega(G)$ denotes the maximum cardinality of a clique in G). As observed in [**67**], the edge clique-cover parameter gives a lower bound on the CP rank:

$$\mathrm{c}(G_A) \leq \mathrm{rank}_{\mathrm{cp}}(A).$$

Indeed, if $A = \sum_{\ell \in [r]} \mathbf{a}_\ell \mathbf{a}_\ell^T$ with $\mathbf{a}_\ell \in \mathbb{R}_+^n$ and $r = \mathrm{rank}_{\mathrm{cp}}(A)$, then the supports of $\mathbf{a}_1, ..., \mathbf{a}_r$ are (not necessarily distinct) cliques that provide an edge clique-cover of G_A by at most r cliques.

The bound $\tau_{\mathrm{cp}}^{\mathrm{sos}}(A)$. In [**67**], a semidefinite parameter $\tau_{\mathrm{cp}}^{\mathrm{sos}}(A)$ is introduced, which is shown to be at least as good as $\mathrm{rank}(A)$. Moreover, $\tau_{\mathrm{cp}}^{\mathrm{sos}}(A)$ is also at least as good as $\mathrm{c}_{\mathrm{frac}}(G_A)$, the *fractional edge clique-cover number*, which is defined by

$$\mathrm{c}_{\mathrm{frac}}(G_A) := \min\Big\{ \sum_{k \in [p]} \lambda_k : \lambda \in \mathbb{R}_+^p, \quad \sum_{k:\{i,j\} \subseteq V_k} \lambda_k \geq 1 \text{ for } \{i,j\} \in E_A \Big\}. \tag{6.26}$$

The parameter $\mathrm{c}_{\mathrm{frac}}$ is the natural linear relaxation of $\mathrm{c}(G_A)$. If we require that λ in (6.26) be integer-valued, then we recover $\mathrm{c}(G_A)$. Thus, we have

$$\max\{\mathrm{rank}(A), \mathrm{c}_{\mathrm{frac}}(G_A)\} \leq \tau_{\mathrm{cp}}^{\mathrm{sos}}(A) \leq \tau_{\mathrm{cp}}(A).$$

In [**80**], it is shown[1] that the bounds $\xi^{\mathrm{cp}}_{2,(2019),\dagger}(A)$ are at least as strong as $\tau^{\mathrm{sos}}_{\mathrm{cp}}(A)$. Hence we have a chain of inequalities

$$\mathrm{c}_{\mathrm{frac}}(G_A) \leq \tau^{\mathrm{sos}}_{\mathrm{cp}}(A) \leq \xi^{\mathrm{cp}}_{2,(2019),\dagger}(A) \leq \xi^{\mathrm{cp}}_{2,\dagger}(A) \leq \xi^{\mathrm{cp,isp}}_{2,\dagger}(A) \leq \tau_{\mathrm{cp}}(A). \tag{6.27}$$

Observe now that the (weak) ideal-sparse bound $\xi^{\mathrm{cp,wisp}}_1(A)$ at level $t = 1$ is at least as good as the parameter $\mathrm{c}_{\mathrm{frac}}(G_A)$.

LEMMA 6.4. *If $A \in \mathcal{CP}_n$ with support graph G_A, then*

$$\mathrm{c}_{\mathrm{frac}}(G_A) \leq \xi^{\mathrm{cp,wisp}}_1(A).$$

PROOF. Let $(L_1, ..., L_p)$ be an optimal solution for the parameter $\xi^{\mathrm{cp,wisp}}_1(A)$. Using (6.20b), we have $L_k(A_{ij} - x_i x_j) \geq 0$ for all $i \neq j$ with $\{i, j\} \subseteq V_k$ and $k \in [p]$, which gives $A_{ij} L_k(1) \geq L_k(x_i x_j)$. Summing over k we get

$$A_{ij} = \sum_{k \in [p] : \{i,j\} \subseteq V_k} L_k(x_i x_j) \leq A_{ij} \sum_{k \in [p] : \{i,j\} \subseteq V_k} L_k(1),$$

using (6.20a) for the first equality. As $A_{ij} > 0$, this gives $\sum_{k:\{i,j\} \subseteq V_k} L_k(1) \geq 1$ for every edge $\{i, j\} \in E_A$. Hence, the vector $\lambda = (L_k(1))_{k=1}^p \in \mathbb{R}^p_+$ is feasible for program (6.26), which implies $\mathrm{c}_{\mathrm{frac}}(G_A) \leq \sum_{k \in [p]} L_k(1) = \xi^{\mathrm{cp,wisp}}_1(A)$, as desired. □

Known upper bounds on the CP rank. General upper bounds on the CP rank are

- $\mathrm{rank}_{\mathrm{cp}}(A) \leq n$ if $n \leq 4$ [**146**],
- $\mathrm{rank}_{\mathrm{cp}}(A) \leq \binom{n+1}{2} - 4$ if $n \geq 5$ [**146**], and
- $\mathrm{rank}_{\mathrm{cp}}(A) \leq \binom{r+1}{2} - 1$ if $r = \mathrm{rank}(A) \geq 2$ [**12**].

It is known that $\mathrm{c}(G_A) \leq n^2/4$ [**63**]. It has been a long-standing conjecture by Drew et al. [**58**] that the CP rank of an $n \times n$ completely positive matrix is at most $n^2/4$. This conjecture, however, was disproved in [**22, 23**] for any $n \geq 7$. In particular, it is shown in [**23**] that the maximum CP rank of an $n \times n$ CP matrix is of the order $n^2/2 + O(n^{3/2})$.

If the support graph G_A is triangle-free, then $|E_A| \leq \mathrm{rank}_{\mathrm{cp}}(A) \leq \max\{n, |E_A|\}$. Moreover, if G_A is connected, triangle-free, and not a tree, then $\mathrm{rank}_{\mathrm{cp}}(A) = |E_A|$ [**58**]. Hence, if G_A is a tree, then $n - 1 = |E_A| \leq \mathrm{rank}_{\mathrm{cp}}(A) \leq n$, with $\mathrm{rank}_{\mathrm{cp}}(A) = n$ if A is nonsingular. By Lemma 6.4, we know that $\xi^{\mathrm{cp,wisp}}_1(A) \geq |E_A|$ if G_A is triangle-free. Hence, the bound $\xi^{\mathrm{cp,wisp}}_1(A)$ gives the exact value of the CP rank when G_A is connected, triangle-free, and not a tree. On the other hand, if G_A is a tree and A is nonsingular, then the bound $\xi^{\mathrm{cp}}_{2,\dagger}(A)$ gives the exact value (equal to n) of the CP rank since it is at least $\tau^{\mathrm{sos}}_{\mathrm{cp}}(A) \geq \mathrm{rank}(A)$ by relation (6.27).

Separation between ideal-sparse and dense bounds. We now give a class of CP matrices that exhibit an arbitrarily large separation between the dense and ideal-sparse bounds at level $t = 1$; these matrices A have size $n = 2m$ and

$$\xi^{\mathrm{cp}}_1(A) < m + 1 \leq m^2 = \xi^{\mathrm{cp,wisp}}_1(A) = \mathrm{rank}_{\mathrm{cp}}(A).$$

[1] Indeed the proof for the relevant result [**80**, Proposition 7] only uses the relation $L((A_{ij} - x_i x_j) x_i x_j) \geq 0$ from (6.18) and the relation $L((\mathbf{x}\mathbf{x}^T)^{\otimes 2}) \preceq A^{\otimes 2}$ in (6.10e).

Example 6.5. ***Complete bipartite support graph matrices.*** *For $n = 2m$, consider the matrix*

$$A = \begin{pmatrix} (m+1)I_m & J_m \\ J_m & (m+1)I_m \end{pmatrix} \in \mathcal{S}^n,$$

where I_m is the identity matrix and J_m the all-ones matrix. Then, A is a CP matrix because it is nonnegative and diagonally dominant. Because the support graph of A is the complete bipartite graph $K_{m,m}$, which is triangle-free, we know from [**58**] *that* $\mathrm{rank}_{\mathrm{cp}}(A) = |E_A| = m^2$. *Using Lemma 6.4, we obtain*

$$\mathrm{c}_{\mathrm{frac}}(K_{m,m}) = \xi_1^{\mathrm{cp,isp}}(A) = \xi_1^{\mathrm{cp,wisp}}(A) = m^2 = \mathrm{rank}_{\mathrm{cp}}(A).$$

Next, we claim $\xi_1^{\mathrm{cp}}(A) < m+1$. For this, observe that $\xi_1^{\mathrm{cp}}(A)$ can be reformulated as

$$\begin{aligned} \xi_1^{\mathrm{cp}}(A) = \min \Big\{ L(1) : & L \in \mathbb{R}[\mathbf{x}]_2^*, \\ & L(1) \geq 1, \\ & L(x_i) \geq \sqrt{A_{ii}} \ (i \in [n]), \\ & L(\mathbf{x}\mathbf{x}^T) = A, \\ & L([\mathbf{x}]_1[\mathbf{x}]_1^T) \succeq 0 \Big\}. \end{aligned}$$

Consider the linear functional $L \in \mathbb{R}[\mathbf{x}]_2^*$ defined by $L(\mathbf{x}\mathbf{x}^T) = A$, $L(x_i) = \sqrt{m+1}$ for $i \in [n]$ and $L(1) = \frac{2m(m+1)}{2m+1}$. We show that L is feasible for the above program, which implies $\xi_1^{\mathrm{cp}}(A) \leq L(1) < m+1$. For this it suffices to show that $L([\mathbf{x}]_1[\mathbf{x}]_1^T) \succeq 0$. By taking the Schur complement with respect to the upper left corner, this boils down to checking that $L(1)A - (m+1)J_m \succeq 0$. As the all-ones vector is an eigenvector of A (with eigenvalue $2m+1$), it is an eigenvector of $L(1)A - (m+1)J_m$ with eigenvalue $L(1)(2m+1) - 2m(m+1) = 0$. Since the matrix A is positive semidefinite, the proof is complete.

6.3. Numerical results and examples

In this section, we present the computed bounds for three classes of matrices: high-CP rank matrices from the literature, randomly generated CP matrices, and doubly-nonnegative matrices that are not CP.

We aim to demonstrate improved bounds due to adding a polynomial matrix localizing constraint (see (6.13a)) and using ideal sparsity (see (6.20)). To this end, we show the following.

First, we show the improvement of $\xi_t^{\mathrm{cp},(2022)}(A)$ over $\xi_t^{\mathrm{cp},(2019)}(A)$ (see Section 6.3.1) due to replacing the constraint (6.10e) with (6.13a).

Second, we show the improvement due to ideal sparsity (see Section 6.3.2) in better bounds (see Table 2) and computation times (see Figures 1 and 2). As a side note, we explore flatness and atom extraction for the dense and ideal-sparse settings (see Table 3).

Thirdly, we show that the ideal sparsity is also better at detecting non-membership in the cone of CP matrices (see Section 6.3.4).

6.3.1. Improvement due to the polynomial matrix localizing constraint. To demonstrate the impact of the constraint (6.13a), we compare our bounds $\xi_3^{\mathrm{cp},2022}(A)$ from [**81**] to the bounds $\xi_3^{\mathrm{cp},(2019)}(A)$ from [**80**] on the CP rank of some matrices A known to have a high CP rank, taken from [**22**]. The boldface entries in Table 1 show a strict improvement in the bounds. For these computations, we used the high precision solver SDPA-GMP [**124**] because MOSEK [**5**] and SDPA [**169, 170**] could not certify solutions. [2]

TABLE 1. Bounds for completely positive rank at level $t = 3$.

A	$\mathrm{rank}(A)$	n	$\lfloor \frac{n^2}{4} \rfloor$	$\xi_3^{\mathrm{cp},(2019)}(A)$	$\xi_3^{\mathrm{cp},(2022)}(A)$	$\mathrm{rank}_{\mathrm{cp}}(A)$
M_7	7	7	12	10.5	**11.4**	14
$\widetilde{M}_7$	7	7	12	10.5	10.5	14
$\widetilde{M}_8$	8	8	16	13.82	**14.5**	18
$\widetilde{M}_9$	9	9	20	17.74	**18.4**	26

6.3.2. Improvement due to ideal sparsity. We consider a class of randomly generated sparse CP matrices. The exact construction is given below. In all numerical examples we considered, the bounds $\xi_t^{\mathrm{cp}}(A)$ and $\xi_t^{\mathrm{cp,isp}}(A)$ obtained for these matrices were always at most $\mathrm{rank}(A) + 2$. So we do not list the numerical bounds for these examples as little insight is gained from them. However, random examples allow us to compare the computation times amongst hierarchies and across various matrix sizes, non-zero densities, and levels. In what follows, the *non-zero density* of a symmetric matrix $A \in \mathcal{S}^n$, denoted nzd(A), is defined as the proportion of non-zero entries above the main diagonal, i.e., $\mathrm{nzd(A)} = |\mathrm{E_A}|/\binom{\mathrm{n}}{2}$. Hence a diagonal matrix has nzd=0, and a dense matrix has nzd=1.

The second class contains examples from the literature whose CP rank is known from theory. However, recall the moment hierarchies provide lower bounds on τ_{cp}, whose value is often unknown and could be strictly less than the CP rank. Regardless these examples give an interesting testbed to evaluate the quality of the new bounds.

The third class of examples consists of doubly nonnegative matrices, which are known not to be completely positive. In running these examples, the hope is to obtain an infeasibility certificate from the solver. Thereby obtaining a numerical certificate that the matrix is not completely positive. In this context, one hierarchy performs better than another if it returns the infeasibility certificate at a lower level or uses less run time.

The size of the matrices involved in the semidefinite programs snowballs with the level t in the hierarchy (roughly, as $\binom{n+t}{t}$), so these problems become quickly too big for the solver (in particular, due to limited memory). We will consider matrices up to

[2] The code is available at: `https://github.com/JAndriesJ/ju-CPrank`

size $n = 12$ for the dense and ideal-sparse hierarchies at level $t = 2$. At level $t = 3$ and for matrices of size $n = 12$, we can only compute bounds for the weak-ideal-sparse hierarchy.

All computations shown were run on Windows 11 Home 64-bit with an 11th Gen Intel(R) Core(TM) i7-11800H @ 2.30GHz Processor and 16GB of RAM. The software we use was custom coded in Julia [**20**] utilizing the JuMP [**59**] package for problem formulation and MOSEK [**5**] as the semidefinite programming solver. [3]

Randomly generated sparse CP matrices. We first describe how we construct random sparse CP matrices. Given integers $n \in \mathbb{N}$ and $n - 1 \leq m \leq \binom{n}{2}$, we create a symmetric $n \times n$ binary matrix M with exactly m ones above the diagonal, whose positions are selected uniformly at random. Let G be the graph with M as its adjacency matrix. We only keep the instances where G is a connected graph. We enumerate the maximal cliques $V_1, ..., V_p$ of G (using, e.g., the Bron-Kerbosch algorithm [**26**]). Then, we select a subset of maximal cliques $V_{q_1}, ..., V_{q_l}$ whose union covers every edge of G (e.g., using a greedy algorithm). For each $k \in [l]$, generate $m_k \geq 1$ vectors $(\mathbf{a}^{(k,i)})_{i \in [m_k]} \subseteq \mathbb{R}^n_+$ with uniformly random entries following $\mathcal{U}[0, 1]$ and supported by V_{q_k}. We will choose $m_k = 2$ by default. Then consider the matrix $\sum_{k \in [l]} \sum_{i \in [m_k]} \mathbf{a}^{(k,i)} (\mathbf{a}^{(k,i)})^T$ and scale it so that all diagonal entries are equal to 1, call A the resulting matrix. By construction, A is completely positive with connected support $G_A = G$, and non-zero density $\mathrm{nzd} = \mathrm{m}/\binom{\mathrm{n}}{2}$.

We generate random examples for varying matrix size ($n = 5, 6, 7, 8, 9$) and incrementing nzd in ascending order. To not include examples with disconnected graphs, we need $\mathrm{nzd} \geq (\mathrm{n} - 1)/\binom{\mathrm{n}}{2}$. To account for different graph configurations with the same non-zero density, we generate 10 examples per matrix size and nzd value. For all of them, we compute the dense- and weak-ideal-sparse bounds of level $t = 2$ and $t = 3$. Here we are not interested in the numerical bounds but rather in their computation times. This numerical experiment allows us to show the differences in computation time between the ideal-sparse and dense hierarchies. It turns out that the computation times for the parameters ξ^{cp}_t, $\xi^{\mathrm{cp}}_{t,\dagger}$, and $\xi^{\mathrm{cp}}_{t,\ddagger}$ are all comparable at level $t = 2, 3$, likewise for the ideal-sparse analogs. For this reason, we only plot the results for the "†" variant, i.e., for the parameters $\xi^{\mathrm{cp}}_{t,\dagger}$, $\xi^{\mathrm{cp,isp}}_{t,\dagger}$, $\xi^{\mathrm{cp,wisp}}_{t,\dagger}$. The results are shown in Figure 1 (for $t = 2$) and in Figure 2 (for $t = 3$).

We can make the following observations about the results in Figure 1. As expected, the ideal-sparse hierarchy is faster to compute than the dense hierarchy for matrices with non-zero density nzd ≤ 0.8. The computation of the weak-ideal-sparse hierarchy is even faster. Moreover, the speed-up increases with the matrix size and the hierarchy level, as seen across Figures 1 and 2. At level $t = 3$, some hierarchies can no longer be computed for specific matrix sizes and non-zero densities. This is particularly evident in the case of the dense hierarchy for matrices of size seven and larger. The ideal-sparse hierarchies can be computed up to size nine depending on

[3] See the code repository https://github.com/JAndriesJ/ju-cp-rankju-cp-rank.

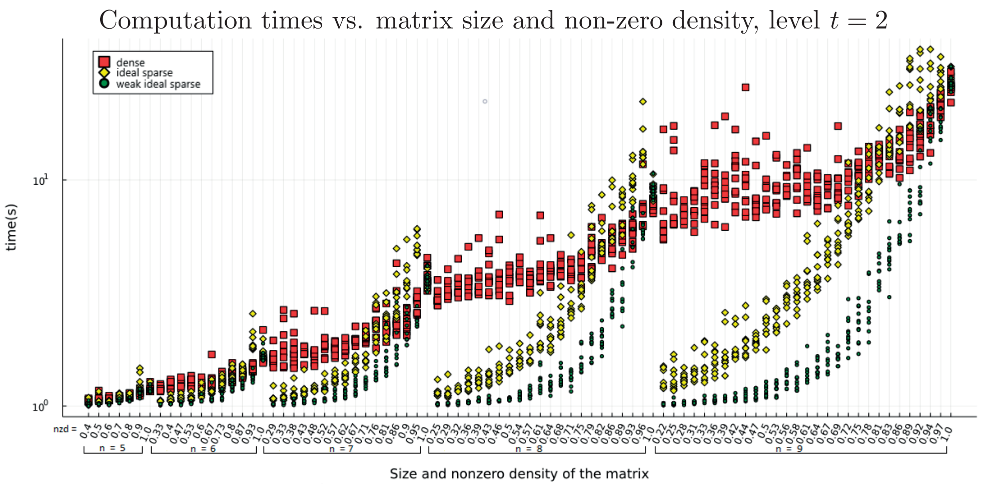


FIGURE 1. Scatter plot of the computation times (in seconds) for the three hierarchies $\xi^{\mathrm{cp}}_{2,\dagger}$ (indicated by a red square), $\xi^{\mathrm{cp,isp}}_{2,\dagger}$ (indicated by a yellow losange), $\xi^{\mathrm{cp,wisp}}_{2,\dagger}$ (indicated by a green circle) against matrix size and non-zero density for 850 random matrices, generated using the above-described procedure. The matrices are arranged in ascending size ($n = 5, 6, 7, 8, 9$) and then ascending non-zero density, ranging from the minimal density needed to have a connected support graph up to a fully dense matrix (nzd = 1).

the non-zero density. We only show examples we could compute in less than 10^3 seconds. The parameters that either took longer than 10^3 seconds or exceeded memory constraints can be inferred by the absence of their respective markers in Figure 2.

Selected sparse CP matrices. Here we compute the dense and (weak) ideal-sparse parameters for a few selected CP matrices from the literature. We first briefly discuss the four example matrices we will consider, denoted ex1, ex2, ex3, ex4, and shown below.

$$\mathrm{ex1} = \begin{pmatrix} 3 & 2 & 0 & 0 & 1 \\ 2 & 5 & 6 & 0 & 0 \\ 0 & 6 & 14 & 4 & 0 \\ 0 & 0 & 4 & 9 & 1 \\ 1 & 0 & 0 & 1 & 2 \end{pmatrix}, \quad \mathrm{ex2} = \begin{pmatrix} 2 & 0 & 0 & 1 & 1 \\ 0 & 2 & 0 & 1 & 1 \\ 0 & 0 & 2 & 1 & 1 \\ 1 & 1 & 1 & 3 & 0 \\ 1 & 1 & 1 & 0 & 3 \end{pmatrix},$$

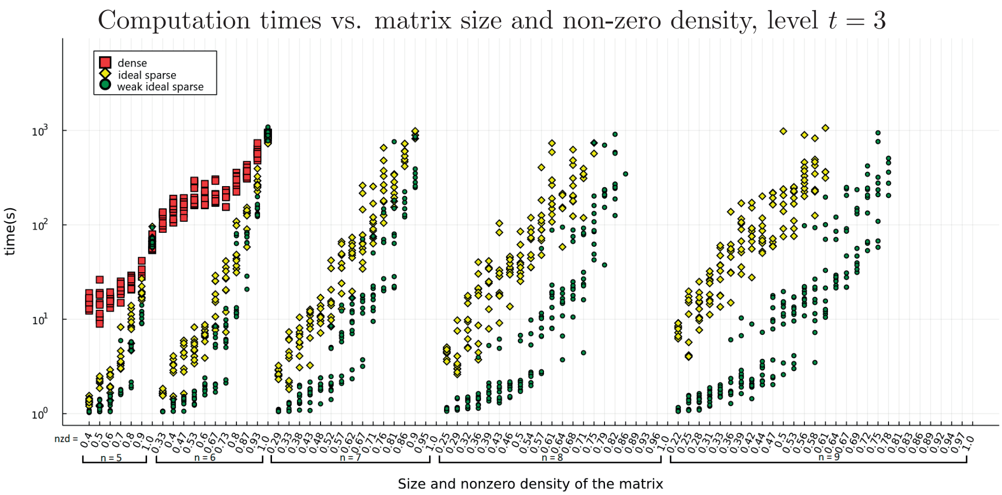


FIGURE 2. This is a similar plot to Figure 1 but now for level t=3 of each of the hierarchies. By omitting markers, we indicate that the corresponding computations either exceeded memory constraints or took longer than 10^3 seconds.

$$\mathrm{ex3} = \begin{pmatrix}
781 & 0 & 72 & 36 & 228 & 320 & 240 & 228 & 36 & 96 & 0 \\
0 & 845 & 0 & 96 & 36 & 228 & 320 & 320 & 228 & 36 & 96 \\
72 & 0 & 827 & 0 & 72 & 36 & 198 & 320 & 320 & 198 & 36 \\
36 & 96 & 0 & 845 & 0 & 96 & 36 & 228 & 320 & 320 & 228 \\
228 & 36 & 72 & 0 & 781 & 0 & 96 & 36 & 228 & 240 & 320 \\
320 & 228 & 36 & 96 & 0 & 845 & 0 & 96 & 36 & 228 & 320 \\
240 & 320 & 198 & 36 & 96 & 0 & 745 & 0 & 96 & 36 & 228 \\
228 & 320 & 320 & 228 & 36 & 96 & 0 & 845 & 0 & 96 & 36 \\
36 & 228 & 320 & 320 & 228 & 36 & 96 & 0 & 845 & 0 & 96 \\
96 & 36 & 198 & 320 & 240 & 228 & 36 & 96 & 0 & 745 & 0 \\
0 & 96 & 36 & 228 & 320 & 320 & 228 & 36 & 96 & 0 & 845
\end{pmatrix},$$

$$\mathrm{ex4} = \begin{pmatrix}
91 & 0 & 0 & 0 & 19 & 24 & 24 & 24 & 19 & 24 & 24 & 24 \\
0 & 42 & 0 & 0 & 24 & 6 & 6 & 6 & 24 & 6 & 6 & 6 \\
0 & 0 & 42 & 0 & 24 & 6 & 6 & 6 & 24 & 6 & 6 & 6 \\
0 & 0 & 0 & 42 & 24 & 6 & 6 & 6 & 24 & 6 & 6 & 6 \\
19 & 24 & 24 & 24 & 91 & 0 & 0 & 0 & 19 & 24 & 24 & 24 \\
24 & 6 & 6 & 6 & 0 & 42 & 0 & 0 & 24 & 6 & 6 & 6 \\
24 & 6 & 6 & 6 & 0 & 0 & 42 & 0 & 24 & 6 & 6 & 6 \\
24 & 6 & 6 & 6 & 0 & 0 & 0 & 42 & 24 & 6 & 6 & 6 \\
19 & 24 & 24 & 24 & 19 & 24 & 24 & 24 & 91 & 0 & 0 & 0 \\
24 & 6 & 6 & 6 & 24 & 6 & 6 & 6 & 0 & 42 & 0 & 0 \\
24 & 6 & 6 & 6 & 24 & 6 & 6 & 6 & 0 & 0 & 42 & 0 \\
24 & 6 & 6 & 6 & 24 & 6 & 6 & 6 & 0 & 0 & 0 & 42
\end{pmatrix}.$$

The matrix ex1 (from [**11**]) is supported by the 5-cycle C_5 and the matrix ex2 (from [**168**]) is supported by the bipartite graph $K_{3,2}$. In both cases, we have that $\xi_1^{\text{cp,isp}}(A) = \text{rank}_{\text{cp}}(A) = |E_A|$ (combining Lemma 6.4 and the results of [**58**] mentioned earlier at the end of Section 6.2). The matrices ex3 and ex4 were constructed, respectively, in [**22, 23**] as examples of matrices having a large CP rank exceeding the value $n^2/4$ (thus refuting the conjecture by Drew et al. [**58**]). The matrix ex3 is supported by $\overline{C_{11}}$, the complement of an 11-cycle, and matrix ex4 is supported by the complete tripartite graph $K_{4,4,4}$. One can verify that the edge clique-cover number is equal to 8 for $\overline{C_{11}}$ and to 16 for $K_{4,4,4}$.

The numerical results for these four examples are presented in Table 2, where we also show other parameters for the matrix (size n, rank r, CP rank r_{cp}) and its support graph (number p of maximal cliques, edge clique-cover number c). Here are some comments about Table 2.

TABLE 2. Dense and ideal-sparse bounds for literature selected sparse CP matrices

A	n	p	c	r	bounds			r_{cp}	times (seconds)		
					ξ_1^{cp}	$\xi_1^{\text{cp,isp}}$	$\xi_1^{\text{cp,wisp}}$		dense	isp	wisp
ex1	5	5	5	5	2.71	5	5	5	< 1	< 1	< 1
ex2	5	6	6	4	3	6	6	6	< 1	< 1	< 1
ex3	11	22	8	11	4.24	8.53	8.53	32	< 1	< 1	< 1
ex4	12	64	16	10	4.85	29.66	29.63	37	< 1	< 1	< 1
					$\xi_{2,\ddagger}^{\text{cp}}$	$\xi_{2,\ddagger}^{\text{cp,isp}}$	$\xi_{2,\ddagger}^{\text{cp,wisp}}$				
ex1	5			5	5	5	5	5	< 1	< 1	< 1
ex2	5			4	6	6	6	6	< 1	< 1	< 1
ex3	11			11	21.93	22.32	22.32	32	123.86	54.89	8.14
ex4	12			10	29.57	29.66	29.66	37	238.94	33.78	1.28
					$\xi_{3,\ddagger}^{\text{cp}}$	$\xi_{3,\ddagger}^{\text{cp,isp}}$	$\xi_{3,\ddagger}^{\text{cp,wisp}}$				
ex3	11			11	-	-	22.33	32	-	-	2648.69
ex4	12			10	-	-	29.66	37	-	-	28.69

n = size of A, p = number of maximal cliques of G_A,
c = edge clique-cover number of G_A, $r = \text{rank}(A)$, $r_{\text{cp}} = \text{rank}_{\text{cp}}(A)$
- : computations that failed due to memory constraints

These results substantiate the claims in Lemma 6.4: the ideal-sparse bound of level $t = 1$ is equal to the number of edges for ex1 and ex2 (and matches the CP rank); moreover, it gives a strong improvement on the dense bound of level 1. The bounds of level $t = 2$ all exceed the rank of the matrix (as expected in view of (6.27)). At level $t = 3$, only the weak-ideal-sparse bound can be computed for the matrices ex3 and ex4.

In Table 2, the values of the bounds at level $t = 3$ are close to those at level $t = 2$ for matrices ex3 and ex4. However, the tests for the flatness condition (3.18) fail, so that one cannot claim that the bounds are equal to τ_{cp} at this stage.

6.3.3. Flatness and atom extraction. For the application to the CP rank, if the flatness condition holds for an optimal solution for the parameter $\xi_t^{\mathrm{cp}}(A)$ (resp., $\xi_t^{\mathrm{cp,isp}}(A)$), then the parameter is equal to $\tau_{\mathrm{cp}}(A)$ and one can extract a CP factorization of A. In this way, one finds an explicit factorization of A and thus an upper bound on its CP rank. If the computed value of $\tau_{\mathrm{cp}}(A)$ equals the number of recovered atoms, this certifies that $\tau_{\mathrm{cp}}(A)$ equals the CP rank and the recovered CP decomposition of A is an optimal one.

We tested whether the flatness conditions (3.16) and (3.18) hold for matrices ex1 and ex2 at level $t = 2$ and whether one can extract atoms and construct a CP factorization.

The results are summarized in Table 3, where we indicate the number of atoms (corresponding to a CP factorization with that many factors) when the extraction procedure is successful. We indicate that the extraction procedure fails by reporting "# atoms=0". As mentioned in [**85**], one may apply the extraction procedure even if flatness does not hold.

For the dense bounds of level $t = 2$, flatness does not hold for the matrices ex1 and ex2. However, while one does not succeed in extracting atoms for matrix ex1, the extraction is successful for matrix ex2 and returns six atoms. Interestingly, flatness holds for the ideal-sparse bounds, and the atom extraction is successful. However, the number of extracted atoms is 10 for matrix ex1, thus twice the CP rank. To verify that the extracted atoms are (approximatively) correct, we use them to construct a CP matrix A_{rec}, which we then compare to the original matrix A. In all cases we obtain $\|A_{\mathrm{rec}} - A\|_1 \leq 10^{-8}$, which shows that a correct factorization has been constructed.

Note that for the ideal-sparse parameter, since one splits the problem over the maximal cliques and has a distinct linear functional L_k for each clique V_k, it may be more difficult to satisfy the flatness condition (3.18) (since each L_k must satisfy it), as happens for matrices ex3 and ex4.

Table 3. Testing flatness and atom extraction

A	$\xi_{2,\ddagger}^{\mathrm{cp}}$		$\xi_{2,\ddagger}^{\mathrm{cp,isp}}$		$\xi_{2,\ddagger}^{\mathrm{cp,wisp}}$	
	flat. (3.16)	# atoms	flat. (3.18)	# atoms	flat. (3.18)	# atoms
ex1	false	0	true	10	false	0
ex2	false	6	true	6	true	6
	$\xi_{3,\ddagger}^{\mathrm{cp}}$		$\xi_{3,\ddagger}^{\mathrm{cp,isp}}$		$\xi_{3,\ddagger}^{\mathrm{cp,wisp}}$	
ex1	false	10	true	10	false	0
ex2	true	6	true	6	true	6

6.3.4. Doubly nonnegative matrices that are not completely positive. We consider the following three matrices that are known to be doubly nonnegative but not completely positive (taken from [**140, 126, 11**]):

$$\text{ex5} = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 \\ 1 & 2 & 1 & 0 & 0 \\ 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & 1 & 2 & 1 \\ 1 & 0 & 0 & 1 & 3 \end{pmatrix}, \quad \text{ex6} = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 \\ 1 & 2 & 1 & 0 & 0 \\ 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & 1 & 2 & 1 \\ 1 & 0 & 0 & 1 & 6 \end{pmatrix},$$

$$\text{ex7} = \begin{pmatrix} 7 & 1 & 2 & 2 & 1 & 1 \\ 1 & 12 & 1 & 3 & 3 & 5 \\ 2 & 1 & 2 & 3 & 0 & 0 \\ 2 & 3 & 3 & 5 & 0 & 0 \\ 1 & 3 & 0 & 0 & 2 & 4 \\ 1 & 5 & 0 & 0 & 4 & 10 \end{pmatrix}$$

The objective is to see whether the hierarchies are able to detect that the matrix is not CP. This can be achieved in two ways: when the solver returns an infeasibility certificate, or when it returns a bound that exceeds a known upper bound on the CP rank. We test this for the bounds at levels $t = 1$ and $t = 2$. At level $t = 2$ we try different variants by adding the constraints (6.16),(6.17), (6.18), and (6.19) and their ideal-sparse analogs. The results are presented in Tables 4 and 5. There we indicate one of three possible outcomes. The first outcome is indicated with a question mark "?", which means that the solver could not reach a decision within the default MOSEK solver parameters. The second possible outcome is when the solver returns an infeasibility certificate (indicated with "*"), or when it returns a value that exceeds a known upper bound for the CP rank (in which case the bound is marked again with "*"). The last column in both tables, labeled $r_{\text{cp}} \leq$, provides such an upper bound on the CP rank of a CP matrix with the given support graph. The third possible outcome is when the solver returns a value that does not violate the upper bound, in which case no conclusion can be drawn. All computations took less than a second and hence computation times are not shown.

TABLE 4. Detecting non-CP matrices for $t = 1$.

A	n	r	ξ_1^{cp}	$\xi_1^{\text{cp,isp}}$	$\xi_1^{\text{cp,wisp}}$	$r_{\text{cp}} \leq$
ex5	5	4	2.47	*	*	5
ex6	5	5	2.59	*	*	5
ex7	6	6	2.4	3.02	3.02	17

* = infeasibility certificate

We make three observations about Tables 4-5. The first is that the ideal-sparse hierarchies show infeasibility at level $t = 1$ already for examples ex5 and ex6, while the dense hierarchy shows the same only at level $t = 2$ with all additional constraints imposed. Secondly, the ideal-sparse hierarchy correctly identifies ex7 as not CP at level $t = 2$ while the dense hierarchy does not succeed even at level $t = 3$. The third observation is that adding additional constraints helps prevent the solver from returning an "unknown result status" but this seems to be less needed in the case of the ideal-sparse hierarchies. It should be noted that increasing the level of the

Table 5. Detecting non CP matrices for $t = 2, 3$.

A	n	r	ξ_2^{cp}	$\xi_2^{\mathrm{cp,isp}}$	$\xi_2^{\mathrm{cp,wisp}}$	$\xi_{2,\dagger}^{\mathrm{cp}}$	$\xi_{2,\dagger}^{\mathrm{cp,isp}}$	$\xi_{2,\dagger}^{\mathrm{cp,wisp}}$	$\xi_{2,\ddagger}^{\mathrm{cp}}$	$\xi_{2,\ddagger}^{\mathrm{cp,isp}}$	$\xi_{2,\ddagger}^{\mathrm{cp,wisp}}$	r_{cp}
ex5	5	4	?	*	?	?	*	*	*	*	*	≤ 5
ex6	5	5	13.56*	?	*	13.56*	*	*	16.11*	*	*	≤ 5
ex7	6	6	?	34.88*	34.01*	12.94	*	*	13.89	*	*	≤ 17
			ξ_3^{cp}	$\xi_3^{\mathrm{cp,isp}}$	$\xi_3^{\mathrm{cp,wisp}}$	$\xi_{3,\dagger}^{\mathrm{cp}}$	$\xi_{3,\dagger}^{\mathrm{cp,isp}}$	$\xi_{3,\dagger}^{\mathrm{cp,wisp}}$	$\xi_{3,\ddagger}^{\mathrm{cp}}$	$\xi_{3,\ddagger}^{\mathrm{cp,isp}}$	$\xi_{3,\ddagger}^{\mathrm{cp,wisp}}$	
ex5	5	4	?	*	?	*	*	*	*	*	*	≤ 5
ex6	5	5	?	*	*	?	*	*	194.2*	*	*	≤ 5
ex7	6	6	?	?	?	?	*	*	?	*	*	≤ 17

* = infeasibility certificate, ? = unknown result status

hierarchy creates more opportunities for numerical errors in the computations, as seen in Table 5.

CHAPTER 7

Separable rank

This chapter considers the separable rank of a complex-valued matrix. We introduce separable bipartite states, which are complex-valued matrices $\rho \in \mathcal{H}_+^n \otimes \mathcal{H}_+^n$. Our interest in these states comes from their use in quantum information, which we briefly touch upon in the introduction of this chapter. Our objective here is not to introduce the reader to physics theory but rather to show that the separable rank, as a parameter, can be lower bounded by a hierarchy of SDPs using the tools of Section 3.1, similar to what we did for the nonnegative rank (Chapter 5) and the completely positive rank (Chapter 6). In Section 7.1, we construct a hierarchy of lower bounds for $\tau_{\text{sep}}(\rho)$, the optimal value of a GMP that lower bounds the separable rank of ρ.

Our contribution to this topic is three-fold: First, we are the first to construct hierarchies of lower bounds for the separable rank. Second, we investigate three (incomparable) hierarchies based on different scalings of the factors. Third, we incorporate a polynomial matrix localizing constraint into the hierarchies, similar to constraint (6.13a) in Section 6.1, to get improved bounds.

Unlike the nonnegative rank and the completely positive rank, the entries of ρ are not nonnegative, and as a result, we do not get ideal constraints of a form where ideal sparsity can be exploited. However, we do consider a block-diagonal reduction technique (Section 7.2) for removing redundant variables, thereby reducing the moment matrix sizes in the associated hierarchy.

Our approach for lower bounding the separable rank extends naturally to the multipartite setting, the mixed separable rank, and the real separable rank (Section 7.3).

We conclude with examples and numerical experiments to substantiate our theoretical results (Section 7.4).

The cone of separable states. Consider the following matrix cone:

$$\mathcal{SEP}_n := \text{cone}\{\mathbf{x}\mathbf{x}^* \otimes \mathbf{y}\mathbf{y}^* : \mathbf{x} \in \mathbb{C}^n,\ \mathbf{y} \in \mathbb{C}^n,\ \|\mathbf{x}\| = \|\mathbf{y}\| = 1\} \subseteq \mathcal{H}_+^n \otimes \mathcal{H}_+^n, \quad (7.1)$$

which is also sometimes denoted as $\mathcal{SEP}$ when the dimension n is not important. Recall that $\mathcal{H}^n$ denotes the cone of complex Hermitian $n \times n$ matrices, and $\mathcal{H}_+^n$ is the subcone of Hermitian positive semidefinite matrices. The cone $\mathcal{SEP}_n$ is of particular interest in quantum information theory; its elements are known as the *(unnormalized, bipartite) separable states* on $\mathcal{H}^n \otimes \mathcal{H}^n$. If a PSD matrix $\rho \in \mathcal{H}^n \otimes \mathcal{H}^n$ does not belong to $\mathcal{SEP}_n$, then it is said to be *entangled*.

Entangled states can be used to observe quantum, non-classical behaviors that two physically separated quantum systems may display, as already pointed out in the early work [**62**]. Entanglement is now recognized as a vital resource used in quantum information theory to carry out various tasks such as quantum computation, quantum communication, quantum cryptography, and teleportation (see, e.g., [**129, 165**] and references therein). Therefore, deciding whether a state is separable or entangled is of fundamental interest in quantum information theory.

Gurvits [**82**] has shown that the (weak) membership problem for the set

$$\mathcal{SEP}_n \cap \{\rho : \mathrm{Tr}(\rho) = 1\}$$

is an NP-hard problem. In addition, the problem was shown to be strongly NP-hard in [**72**]. This is our motivation for finding tractable criteria for the separability or entanglement of quantum states.

The separable rank. Consider a separable state $\rho \in \mathcal{SEP}_n$. Then, its *separable rank*, denoted $\mathrm{rank}_{\mathrm{sep}}(\rho)$, is the smallest integer $r \in \mathbb{N}$ for which there exist vectors $\mathbf{a}_1, ..., \mathbf{a}_r, \mathbf{b}_1, ..., \mathbf{b}_r \in \mathbb{C}^n$ such that

$$\rho = \sum_{\ell \in [r]} \mathbf{a}_\ell \mathbf{a}_\ell^* \otimes \mathbf{b}_\ell \mathbf{b}_\ell^*. \tag{7.2}$$

In other words,

$$\mathrm{rank}_{\mathrm{sep}}(\rho) := \min \Big\{ r \in \mathbb{N} : \rho = \sum_{\ell \in [r]} \mathbf{a}_\ell \mathbf{a}_\ell^* \otimes \mathbf{b}_\ell \mathbf{b}_\ell^*; \ \ \mathbf{a}_\ell, \mathbf{b}_\ell \in \mathbb{C}^n \Big\}. \tag{7.3}$$

If $\rho \in \mathcal{H}^n \otimes \mathcal{H}^n \setminus \mathcal{SEP}_n$, then we set $\mathrm{rank}_{\mathrm{sep}}(\rho) = \infty$. The separable rank has been previously studied, e.g., in [**156, 55, 32**], where it is called the optimal ensemble cardinality or the length of ρ. It can be seen as a 'complexity measure' of the state, with an infinite rank for entangled states.

The lower bound $\tau_{\mathrm{sep}}(\rho)$. Consider the following parameter, which was first introduced in our work [**81**]:

$$\tau_{\mathrm{sep}}(\rho) := \inf \Big\{ \lambda : \lambda > 0, \frac{1}{\lambda}\rho \in \mathrm{conv}\{\mathbf{x}\mathbf{x}^* \otimes \mathbf{y}\mathbf{y}^* : (\mathbf{x}, \mathbf{y}) \in K_\rho\} \Big\}. \tag{7.4}$$

Here, the semialgebraic set K_ρ is defined as

$$K_\rho := \Big\{ (\mathbf{x}, \mathbf{y}) \in \mathbb{C}^n \times \mathbb{C}^n \mid \rho \succeq \mathbf{x}\mathbf{x}^* \otimes \mathbf{y}\mathbf{y}^*; \ \|\mathbf{x}\|_\infty, \ \|\mathbf{y}\|_\infty \leq \rho_{\max}^{1/4} \Big\}. \tag{7.5}$$

The second constraint in (7.5) is motivated by a particular scaling of the separable (SEP) factors. The SEP factors $\mathbf{a}_\ell, \mathbf{b}_\ell$ in (7.2) clearly satisfy the PSD condition

$$\rho - \mathbf{a}_\ell \mathbf{a}_\ell^* \otimes \mathbf{b}_\ell \mathbf{b}_\ell^* \succeq 0 \ (\ell \in [r]). \tag{7.6}$$

Looking at the diagonal, this means that entrywise $|(\mathbf{a}_\ell)_i|^2 |(\mathbf{b}_\ell)_j|^2 \leq \rho_{ij,ij}$ $(i, j \in [n])$, which implies the following bounds on the norms of the SEP factors:

$$\|\mathbf{a}_\ell\|_\infty^2 \cdot \|\mathbf{b}_\ell\|_\infty^2 \leq \rho_{\max} \ (\ell \in [r]), \tag{7.7}$$

$$\|\mathbf{a}_\ell\|_2^2 \cdot \|\mathbf{b}_\ell\|_2^2 \leq \mathrm{Tr}(\rho) \ (\ell \in [r]), \tag{7.8}$$

where

$$\rho_{\max} := \max_{i,j \in [n]} \rho_{ij,ij}$$

denotes the maximum diagonal entry of ρ. Via rescaling, we may assume without loss of generality that $\|\mathbf{a}_\ell\|_\infty = \|\mathbf{b}_\ell\|_\infty$ and

$$\|\mathbf{a}_\ell\|_\infty^2,\ \|\mathbf{b}_\ell\|_\infty^2 \le \sqrt{\rho_{\max}} \quad (\ell \in [r]). \tag{7.9}$$

LEMMA 7.1. *For any $n \in \mathbb{N}$ and $\rho \in \mathcal{SEP}_n$, we have*

$$\tau_{\text{sep}}(\rho) \le \text{rank}_{\text{sep}}(\rho).$$

Moreover, if $\rho \notin \mathcal{SEP}_n$ then $\tau_{\text{sep}}(\rho) = \text{rank}_{\text{sep}}(\rho) = \infty$.

PROOF. Begin with a SEP factorization of $\rho \in \mathcal{SEP}_n$ like the one in (7.2). Assume that we have applied the rescaling (7.9) so that each SEP factor $(\mathbf{a}_\ell, \mathbf{b}_\ell)$ belongs to the set K_ρ. Then,

$$\frac{1}{r}\rho = \sum_{\ell\in[r]} \frac{1}{r}\mathbf{a}_\ell\mathbf{a}_\ell^* \otimes \mathbf{b}_\ell\mathbf{b}_\ell^* \in \text{conv}\{\mathbf{x}\mathbf{x}^* \otimes \mathbf{y}\mathbf{y}^* : (\mathbf{x}, \mathbf{y}) \in K_\rho\},$$

is a feasible solution to (7.4) with objective value $\lambda = r$. Hence, $\tau_{\text{sep}}(\rho) \le \text{rank}_{\text{sep}}(\rho)$.

If $\rho \notin \mathcal{SEP}_n$, then ρ does not have a factorization (7.2) and hence $\text{rank}_{\text{sep}}(\rho) = \infty$. Similarly, there can be no $\lambda > 0$ for which $\frac{1}{\lambda}\rho \in \text{conv}\{\mathbf{x}\mathbf{x}^* \otimes \mathbf{y}\mathbf{y}^* : (\mathbf{x}, \mathbf{y}) \in K_\rho\}$ if $\rho \notin \mathcal{SEP}_n$, and thus $\tau_{\text{sep}}(\rho) = \infty$. □

The parameter $\tau_{\text{sep}}(\rho)$ does not seem any easier to compute than the separable rank. However, it enjoys an additional convexity property that the combinatorial parameter $\text{rank}_{\text{sep}}(\rho)$ does not have. Moreover, $\tau_{\text{sep}}(\rho)$ can be reformulated as the optimal value of a GMP in the form of (3.1).

GMP formulation of the parameter $\tau_{\text{sep}}(\rho)$.

LEMMA 7.2. *The parameter $\tau_{\text{sep}}(\rho)$ is equal to the optimal value of the following generalized moment problem:*

$$\mathbf{val}_{\text{sep}}(\rho) := \inf_{\mu\in\mathscr{M}(K_\rho)} \Big\{ \int 1 d\mu : \int x_i\overline{x}_j y_k\overline{y}_l d\mu = \rho_{ij,kl}\ (i,j,k,l \in [n]) \Big\}. \tag{7.10}$$

PROOF. $(\mathbf{val}_{\text{sep}}(\rho) \le \tau_{\text{sep}}(\rho))$ Any feasible solution to $\tau_{\text{sep}}(\rho)$, i.e., any decomposition of the form $\rho = \lambda \sum_{\ell\in[r]} c_\ell \mathbf{a}_\ell\mathbf{a}_\ell^* \otimes \mathbf{b}_\ell\mathbf{b}_\ell^*$, with $\lambda > 0$, $c_\ell > 0$, $\sum_{\ell\in[r]} c_\ell = 1$, $\mathbf{a}_\ell\mathbf{a}_\ell^* \otimes \mathbf{b}_\ell\mathbf{b}_\ell^* \preceq \rho$, and $\|\mathbf{a}_\ell\|_\infty,\ \|\mathbf{b}_\ell\|_\infty \le \rho_{\max}^{1/4}$, corresponds to a finite atomic measure

$$\mu := \lambda \sum_{\ell\in[r]} c_\ell \delta_{(\mathbf{a}_\ell, \mathbf{b}_\ell)}$$

that is feasible for $\mathbf{val}_{\text{sep}}(\rho)$, with objective value r. Hence, $\mathbf{val}_{\text{sep}}(\rho) \le \tau_{\text{sep}}(\rho)$.

$(\mathbf{val}_{\text{sep}}(\rho) \ge \tau_{\text{sep}}(\rho))$ Assume (7.10) is feasible, else $\mathbf{val}_{\text{sep}}(\rho) = \infty$ and there is nothing to prove. Let $\mu \in \mathscr{M}(K_\rho)$ be a feasible for (7.10). In view of Theorem 2.9 (ii), we may assume that μ is a finite atomic measure, i.e., $\mu := \lambda \sum_{\ell\in[r]} c_\ell \delta_{(\mathbf{a}_\ell, \mathbf{b}_\ell)}$, with $\lambda > 0$, $c_\ell > 0$, and $(\mathbf{a}_\ell, \mathbf{b}_\ell) \in K_\rho$. This measure then induces a SEP decomposition

$$\rho = \lambda \sum_{\ell\in[r]} c_\ell \mathbf{a}_\ell\mathbf{a}_\ell^* \otimes \mathbf{b}_\ell\mathbf{b}_\ell^*$$

corresponding to a solution for (7.4), with objective value λ. Hence, we have shown $\mathbf{val}_{\text{sep}}(\rho) \ge \tau_{\text{sep}}(\rho)$.

□

In the next section, we will present a hierarchy of lower bounds on $\tau_{\rm sep}(\rho)$, constructed using tools from Section 3.1. Moreover, these lower bounds will asymptotically converge to $\tau_{\rm sep}(\rho)$.

7.1. Hierarchies of lower bounds for the separable rank

For $t \in \mathbb{N} \cup \{\infty\}$ with $t \geq 2$, define the parameter

$$\begin{aligned}
\xi_t^{\rm sep}(\rho) := \inf \Big\{ L(1) \mid \\
& L \in \mathbb{C}[\mathbf{x},\mathbf{y},\overline{\mathbf{x}},\overline{\mathbf{y}}]_{2t}^* \text{ (Hermitian)}, \\
& L(\mathbf{x}\mathbf{x}^* \otimes \mathbf{y}\mathbf{y}^*) = \rho, \\
& L([\mathbf{x},\mathbf{y},\overline{\mathbf{x}},\overline{\mathbf{y}}]_t[\mathbf{x},\mathbf{y},\overline{\mathbf{x}},\overline{\mathbf{y}}]_t^*) \succeq 0, \\
& L((\sqrt{\rho_{\max}} - x_i\overline{x}_i)[\mathbf{x},\mathbf{y},\overline{\mathbf{x}},\overline{\mathbf{y}}]_{t-1}[\mathbf{x},\mathbf{y},\overline{\mathbf{x}},\overline{\mathbf{y}}]_{t-1}^*) \succeq 0 \ (i \in [n]), \quad (7.11a) \\
& L((\sqrt{\rho_{\max}} - y_i\overline{y}_i)[\mathbf{x},\mathbf{y},\overline{\mathbf{x}},\overline{\mathbf{y}}]_{t-1}[\mathbf{x},\mathbf{y},\overline{\mathbf{x}},\overline{\mathbf{y}}]_{t-1}^*) \succeq 0 \ (i \in [n]), \quad (7.11b) \\
& L((\rho - \mathbf{x}\mathbf{x}^* \otimes \mathbf{y}\mathbf{y}^*) \otimes [\mathbf{x},\mathbf{y},\overline{\mathbf{x}},\overline{\mathbf{y}}]_{t-2}[\mathbf{x},\mathbf{y},\overline{\mathbf{x}},\overline{\mathbf{y}}]_{t-2}^*) \succeq 0 \Big\}. \quad (7.11c)
\end{aligned}$$

As explained in Section 3.1.1, this does yield a hierarchy, i.e.,

$$\xi_2^{\rm sep}(\rho) \leq \xi_3^{\rm sep}(\rho) \leq \cdots \leq \xi_\infty^{\rm sep}(\rho). \tag{7.12}$$

Archimedean quadratic module. Observe that the polynomials involved in (7.11a) and (7.11b) generate an Archimedean quadratic module, since we have

$$2n \cdot \sqrt{\rho_{\max}} - \sum_{i\in[n]} (x_i\overline{x}_i + y_i\overline{y}_i) \in \mathcal{M}_2\Big(\underbrace{\{\sqrt{\rho_{\max}} - x_i\overline{x}_i,\ \sqrt{\rho_{\max}} - y_i\overline{y}_i : i \in [n]\}}_{H:=}\Big). \tag{7.13}$$

Here, the polynomials in H define constraints that are equivalent to the last constraints in (7.5).

Remark 7.3. *Note that the localizing constraints (7.11a) and (7.11b) imply the localizing constraints corresponding to (7.7). This follows from the identity:*

$$\rho_{\max} - x_i\overline{x}_i y_j\overline{y}_j = (\sqrt{\rho_{\max}} - x_i\overline{x}_i)y_j\overline{y}_j + \sqrt{\rho_{\max}}(\sqrt{\rho_{\max}} - y_j\overline{y}_j).$$

Similarly, the polynomial matrix localizing constraint (7.11c) implies the localizing constraints corresponding to (7.8). This follows from the identity:

$$\mathrm{Tr}(\rho) - (\sum_i x_i\overline{x}_i)(\sum_j y_j\overline{y}_j) = \sum_{i,j}(\rho_{ij,ij} - x_i\overline{x}_i y_j\overline{y}_j) = \sum_{i,j} e_{ij}^T(\rho - \mathbf{x}\mathbf{x}^* \otimes \mathbf{y}\mathbf{y}^*)e_{ij}.$$

Here, we use $e_{ij} := e_i \otimes e_j$ to denote the tensor product of the i^{th} unit vector of $\mathbb{R}^n$ with the j^{th} unit vector of $\mathbb{R}^n$.

Thus we have motivated the choice of scaling in (7.11a) and (7.11b) as opposed to directly using (7.7) or (7.8).

Linking the hierarchy $\xi_t^{\text{sep}}(\rho)$ to the parameter $\tau_{\text{sep}}(\rho)$. By Theorem 3.2, we can conclude the following result.

Lemma 7.4. *For any $\rho \in \mathcal{SEP}_n$ and each $t \in \mathbb{N} \cup \{\infty\}$, the program (7.11) attains its optimum, and*

$$\lim_{t\to\infty} \xi_t^{\text{sep}}(\rho) = \xi_\infty^{\text{sep}}(\rho) = \tau_{\text{sep}}(\rho).$$

Moreover, the GMP (7.10) has an optimal solution μ that is finite atomic and is supported on K_ρ.

Proof. We simply check that the three assumptions of Theorem 3.2 hold. Assumption (A) is satisfied because of Lemma 7.1, assumption (B) because of (7.13), and assumption (C) holds by taking $z_{i,j,k,l} = 0$ $(i, j, k, l \in [n])$ and $c = \frac{1}{2}$. □

Alternative SEP factor scalings. The scaling we used in (7.9) resulted in the localizing constraints (7.11a) and (7.11b). However, this is only one of several possible scalings. We now consider three other (possibly mutually exclusive) scalings using the Euclidean norm that follow from (7.8):

$$\|\mathbf{a}_\ell\|_2^2 \le \mathrm{Tr}(\rho),\ \|\mathbf{b}_\ell\|_2 = 1\ (\ell \in [r]), \tag{7.14}$$

$$\|\mathbf{a}_\ell\|_2^2 = \|\mathbf{b}_\ell\|_2^2 \le \sqrt{\mathrm{Tr}(\rho)}\ (\ell \in [r]), \tag{7.15}$$

$$\|\mathbf{a}_\ell\|_2^2 \le \sqrt{\mathrm{Tr}(\rho)},\ \|\mathbf{b}_\ell\|_2 = \sqrt{\mathrm{Tr}(\rho)}\ (\ell \in [r]). \tag{7.16}$$

From each of these scalings, we can derive the following sets of polynomials:

- $\big\{\mathrm{Tr}(\rho) - \|\mathbf{x}\|^2, \pm(1 - \|\mathbf{y}\|^2)\big\}$, corresponding to (7.14),
- $\big\{\pm(\|\mathbf{x}\|^2 - \|\mathbf{y}\|^2), \sqrt{\mathrm{Tr}(\rho)} - \|\mathbf{y}\|^2\big\}$, corresponding to (7.15),
- $\big\{\sqrt{\mathrm{Tr}(\rho)} - \|\mathbf{x}\|^2, \pm 1(\sqrt{\mathrm{Tr}(\rho)} - \|\mathbf{y}\|^2)\big\}$, corresponding to (7.16).

Replacing the constraints (7.11a) and (7.11b) in (7.9) with the localizing constraints based on the above polynomials produces other hierarchies different from ξ_t^{sep}. The resulting hierarchies are incomparable in that there are examples for each case where each one provides better bounds than the others. We give a more detailed explanation of this in Section 7.4.

7.1.1. SEP membership tests. Since $\mathcal{SEP}_n$ is a n^4-dimensional cone, by Carathéodory's theorem we have $\mathrm{rank}_{\text{sep}}(\rho) \le n^4$. Similarly, $\mathrm{rank}_{\text{sep}}(\rho) \le \mathrm{rank}(\rho)^2$ holds. Hence, we have the following necessary condition on any separable state $\rho \in \mathcal{SEP}_n$:

$$\mathrm{rank}_{\text{sep}}(\rho) \le \mathrm{rank}(\rho)^2 \le n^4.$$

Using this necessary condition, we can conclude that $\rho \in \mathcal{H}^n \otimes \mathcal{H}^n \setminus \mathcal{SEP}_n$ if, for some integer $2 \le t \in \mathbb{N}$, one would have $\xi_t^{\text{sep}}(\rho) > \mathrm{rank}(\rho)^2$. Thus, we have a tractable test for non-membership in $\mathcal{SEP}_n$. As we now show, the reverse implication also holds.

Lemma 7.5. *Let $\rho \in \mathcal{H}^n \otimes \mathcal{H}^n$. Then, we have*

$$\rho \in \mathcal{SEP}_n \iff \xi_t^{\text{sep}}(\rho) \le \mathrm{rank}(\rho)^2 \text{ for all } t \ge 2.$$

Proof. ($\Rightarrow$) It is clear that $\xi_t^{\text{sep}}(\rho) \le \mathrm{rank}_{\text{sep}}(\rho) \le \mathrm{rank}(\rho)^2$ when $\rho \in \mathcal{SEP}_n$. ($\Leftarrow$) Conversely, assume $\xi_t^{\text{sep}}(\rho) \le \mathrm{rank}(\rho)^2$ for all integers $t \ge 2$. Then, using Lemma 3.1, one can conclude the existence of a linear functional $L \in \mathbb{C}[\mathbf{x}, \mathbf{y}, \overline{\mathbf{x}}, \overline{\mathbf{y}}]^*$ feasible for $\xi_\infty^{\text{sep}}(\rho)$, so that $\xi_\infty^{\text{sep}}(\rho) \le L(1) < \infty$. Then, by Lemma 7.4, we have $\tau_{\text{sep}}(\rho) < \infty$, which shows ρ is separable. □

Alternatively, one can substitute $\mathrm{rank}(\rho)^2$ with any other valid upper bound on $\mathrm{rank}_{\mathrm{sep}}(\rho)$. Consider, for example, the *birank* of ρ, which is defined as the pair $(\mathrm{rank}(\rho), \mathrm{rank}(\rho^{T_B}))$. Here, $\cdot^{T_B}$ denotes the operation of taking the partial transpose on the second register, i.e.,

$$A \otimes B = \left[\begin{array}{c|c|c} A_{11}B & \cdots & A_{1n}B \\ \hline \vdots & \ddots & \vdots \\ \hline A_{n1}B & \cdots & A_{nn}B \end{array}\right], \ (A \otimes B)^{T_B} = \left[\begin{array}{c|c|c} A_{11}B^T & \cdots & A_{1n}B^T \\ \hline \vdots & \ddots & \vdots \\ \hline A_{n1}B^T & \cdots & A_{nn}B^T \end{array}\right].$$

Since $\mathrm{rank}_{\mathrm{sep}}(\rho) = \mathrm{rank}_{\mathrm{sep}}(\rho^{T_B})$, we have

$$\max\left\{\mathrm{rank}(\rho), \mathrm{rank}(\rho^{T_B})\right\} \leq \mathrm{rank}_{\mathrm{sep}}(\rho) \leq \Big(\min\left\{\mathrm{rank}(\rho), \mathrm{rank}(\rho^{T_B})\right\}\Big)^2. \tag{7.17}$$

Yet another necessary condition for separability is the *positive partial transpose (PPT)* criterion, which states that if $p \in \mathbb{C}^{n_1} \otimes \mathbb{C}^{n_2}$ is PSD, then so is ρ^{T_B}. The PPT criterion was introduced in [**130, 86**]. While it was shown to be sufficient to ensure the separability of bipartite states $\rho \in \mathbb{C}^2 \otimes \mathbb{C}^3$ [**167**], it is, in general, not sufficient for the separability of states acting on larger dimensional spaces (see, e.g., [**87, 167**]). It has been shown that no semidefinite representation exists for $\mathcal{SEP}_n$ when $n \geq 3$ [**64**]. The PPT criterion is an easy necessary condition to consider before resorting to more involved membership tests. In the latter half of [**81**], the PPT criterion is examined from the perspective of the moment method.

We will use these results in Section 7.4 to determine how well our hierarchy can detect known entangled states from the literature.

7.2. Block-diagonal reduction for the parameter $\xi_t^{\mathrm{sep}}(\rho)$

Observe that only monomials of the form $\mathbf{x}^\alpha \overline{\mathbf{x}}^{\alpha'} \mathbf{y}^\beta \overline{\mathbf{y}}^{\beta'}$, with $|\alpha| = |\alpha'|$, $|\beta| = |\beta'|$, occur in the program (7.11) defining $\xi_t^{\mathrm{sep}}(\rho)$. As such, we will try to remove the monomials that do not satisfy this property to create a more economical program than (7.11) with equally strong bounds.

In Lemma 7.6 below, we show that we may restrict the optimization in (7.11) to linear functionals L that satisfy the condition

$$L(\mathbf{x}^\alpha \overline{\mathbf{x}}^{\alpha'} \mathbf{y}^\beta \overline{\mathbf{y}}^{\beta'}) = 0 \ \text{ if } |\alpha| \neq |\alpha'| \text{ or } |\beta| \neq |\beta'|. \tag{7.18}$$

In particular, (7.18) implies that $L(\mathbf{x}^\alpha \overline{\mathbf{x}}^{\alpha'} \mathbf{y}^\beta \overline{\mathbf{y}}^{\beta'}) = 0$ if either $|\alpha + \alpha'|$ or $|\beta + \beta'|$ is odd.

The primary advantage of using only functionals that satisfies (7.18) is that the associated moment matrix and localizing matrices become block-diagonal. By only having to check PSDness on the diagonal blocks (and not the whole) of the matrix, the associated SDP requires fewer computational resources, thereby leading to possibly faster computations and access to higher levels of the hierarchy. To see this, consider first the matrix $L([\mathbf{x}, \mathbf{y}, \overline{\mathbf{x}}, \overline{\mathbf{y}}]_t [\mathbf{x}, \mathbf{y}, \overline{\mathbf{x}}, \overline{\mathbf{y}}]_t^*)$, which is indexed by the set

$$I^t := \{(\alpha, \alpha', \beta, \beta') \in (\mathbb{N}^n)^4 : |\alpha + \beta + \alpha' + \beta'| \leq t\}. \tag{7.19}$$

Here, the tuple $(\alpha, \alpha', \beta, \beta')$ corresponds to the monomial $\mathbf{x}^\alpha \overline{\mathbf{x}}^{\alpha'} \mathbf{y}^\beta \overline{\mathbf{y}}^{\beta'}$. One can partition the index set as follows:

$$I^t = \bigcup_{r,s=-t}^{t} I^t_{r,s},$$

using the sets

$$I^t_{r,s} := \big\{(\alpha, \alpha', \beta, \beta') \in I^t : |\alpha| - |\alpha'| = r,\ |\beta| - |\beta'| = s\big\}\ (r, s \in [-t, t]). \tag{7.20}$$

With respect to this partition of its index set, the matrix $M_t(L)$ is block-diagonal, and thus $M_t(L) \succeq 0$ if and only if its principal submatrices $M_t(L)[I^t_{r,s}]$ indexed by the sets $I^t_{r,s}$ are positive semidefinite, i.e.,

$$M_t(L) \succeq 0 \iff M_t(L)[I^t_{r,s}] \succeq 0\ (r, s \in [-t, t]).$$

The same reasoning applies to each localizing moment matrices in (7.11a) and (7.11b) (indexed by I^{t-1}), and similarly to $L((\rho - \mathbf{x}\mathbf{x}^* \otimes \mathbf{y}\mathbf{y}^*) \otimes [\mathbf{x}, \mathbf{y}, \overline{\mathbf{x}}, \overline{\mathbf{y}}]_{t-2}[\mathbf{x}, \mathbf{y}, \overline{\mathbf{x}}, \overline{\mathbf{y}}]^*_{t-2})$ in (7.11c) (indexed by I^{t-2}).

Lemma 7.6. *Adding the constraint (7.18) to the definition (7.11) of the parameter ξ_t^{sep} does not change the optimal value.*

Proof. It suffices to show that for any L feasible for $\xi_t^{\mathrm{sep}}(\rho)$, we can construct another feasible solution $\widetilde{L}$ with the same objective value as L and satisfying (7.18). Begin by defining

$$\widetilde{L}(\mathbf{x}^\alpha \overline{\mathbf{x}}^{\alpha'} \mathbf{y}^\beta \overline{\mathbf{y}}^{\beta'}) = \begin{cases} L(\mathbf{x}^\alpha \overline{\mathbf{x}}^{\alpha'} \mathbf{y}^\beta \overline{\mathbf{y}}^{\beta'}) & \text{if } |\alpha| = |\alpha'| \text{ and } |\beta| = |\beta'|, \\ 0 & \text{otherwise.} \end{cases} \tag{7.21}$$

Then, by construction, $\widetilde{L}$ satisfies (7.18) and $\widetilde{L}(1) = L(1)$. We now show that $\widetilde{L}$ is feasible for the program (7.11). Clearly, we have $\widetilde{L}(\mathbf{x}\mathbf{x}^* \otimes \mathbf{y}\mathbf{y}^*) = \rho$.

($M_t(\widetilde{L}) \succeq 0$) The matrix $M_t(\widetilde{L})$ is block-diagonal with respect to the partition $I_t = \cup_{r,s=0}^{t} I_{r,s}$ of its index set I^t. The principal submatrix $M_t(L)[I^t_{r,s}]$ of $M_t(L)$ corresponds exactly to the $I^t_{r,s}$-block of $M_t(\widetilde{L})$ because the monomials involved are of the form

$$\mathbf{x}^\gamma \overline{\mathbf{x}}^{\gamma'} \mathbf{y}^\delta \overline{\mathbf{y}}^{\delta'} \text{ with } |\gamma| = |\gamma'| \text{ and } |\delta| = |\delta'|. \tag{7.22}$$

Hence, because $M_t(L) \succeq 0$, it follows that $M_t(L)[I^t_{r,s}] \succeq 0$ $(r, s \in [-t, t])$ and thus $M_t(\widetilde{L}) \succeq 0$.

($M_{t-d_g}(g\widetilde{L}) \succeq 0$) The preceding argument hold mutatis mutandis for showing that $M_{t-d_g}(g\widetilde{L}) \succeq 0$ for $g \in S := \big\{\sqrt{\rho_{\max}} - x_i\overline{x}_i,\ \sqrt{\rho_{\max}} - y_i\overline{y}_i : i \in [n]\big\}$. The diference now is that we adjust the index set $I^{t-1} = \cup_{r,s=-t+1}^{t-1} I^{t-1}_{r,s}$ to account for the degree of g. This works because the polynomials in S have the property that the terms have the same degree in $\mathbf{x}$ as they do in $\overline{\mathbf{x}}$ (resp., $\mathbf{y}$ as they do in $\overline{\mathbf{y}}$), namely either 0 or 1. Hence, if a monomial $\mathbf{x}^\gamma \overline{\mathbf{x}}^{\gamma'} \mathbf{y}^\delta \overline{\mathbf{y}}^{\delta'}$ is of the form (7.22), then the terms of $\mathbf{x}^\gamma \overline{\mathbf{x}}^{\gamma'} \mathbf{y}^\delta \overline{\mathbf{y}}^{\delta'} \cdot g(\mathbf{x}, \overline{\mathbf{x}}, \mathbf{y}, \overline{\mathbf{y}})$ will also be of the form (7.22).

($M_{t-2}(\rho - \mathbf{x}\mathbf{x}^* \otimes \mathbf{y}\mathbf{y}^* \otimes \widetilde{L}) \succeq 0$) The analogous reasoning applies to showing that $M_{t-2}(\rho - \mathbf{x}\mathbf{x}^* \otimes \mathbf{y}\mathbf{y}^* \otimes \widetilde{L}) \succeq 0$. Now, we use the modified index set $[n]^2 \times I_{t-2}$ and its partition

$$[n]^2 \times I_{t-2} = \bigcup_{r,s=-t+2}^{t-2} ([n]^2 \times I_{r,s}^{t-2}).$$

Then, $M_{t-2}(G_\rho \otimes \widetilde{L})$ is block-diagonal with respect to this partition, with each block corresponding to a principal submatrix of $M_{t-2}(G_\rho \otimes L)$. Hence, $M_{t-2}(G_\rho \otimes \widetilde{L}) \succeq 0$.

Thus, $\widetilde{L}$ is feasible for the program (7.11), with $\widetilde{L}(1) = L(1)$. □

Example 7.7. ***A Block-diagonal reduction example.*** *To illustrate the effect of the block-diagonalization, we consider an example with $\rho \in \mathcal{H}^3 \otimes \mathcal{H}^3 \simeq \mathcal{H}^9$ (i.e., $n = 3$) and the bound $\xi_3^{\mathrm{sep}}(\rho)$ at level $t = 3$.*

In Table 1, we indicate the respective sizes of the matrices involved in the program for $\xi_3^{\mathrm{sep}}(\rho)$ with and without block-diagonalization. There, '# entries' stands for $\sum_i m_i^2$, where m_i are the sizes of the matrices involved in the program, and '# variables' indicates the total number of variables in each case. The last line indicates the typical run time for such an instance, we collect the computational details later in Section 7.4. The un-block-diagonalized program cannot be solved; thus, block-diagonalization is crucial to enable computation.

For the next case $n = 4$ (i.e., $\rho \in \mathcal{H}^4 \otimes \mathcal{H}^4$), one can compute the bound at level $t = 2$ but not at level $t = 3$, even after block-diagonalization.

Table 1. Matrix sizes block-diagonalized vs. not.

Matrix	block-diagonalized	not
$M_3(L)$	$25 \times (12 \times 12$ to $96 \times 96)$	455×455
$M_2(gL)$	$78 \times (6 \times 6$ to $38 \times 38)$	$6 \times (91 \times 91)$
$M_1(G_\rho \otimes L)$	$5 \times (36 \times 36$ to $108 \times 108)$	234×234
# entries	110480	286624
# variables	6952	18564
run time	4.6 min	memory error

Remark 7.8. *As observed above, using the block-diagonalized version of the program for ξ_3^{sep} is crucial to be able to compute the bounds for some larger matrix sizes. We note, however, that the optimal solution to this program will not satisfy the flatness condition*

$$\operatorname{rank} M_t(L) = \operatorname{rank} M_{t-1}(L) \ (t = 2, 3).$$

Indeed, one can check that this flatness condition can hold only in the trivial case $\rho = 0$. Intuitively this can be (roughly) explained by noting that, due to its symmetric structure, L tends to lie within the interior of the feasible region. Hence our approach, which produces lower bounds on $\mathrm{rank}_{\mathrm{sep}}(\rho)$, can be viewed as being complementary to the approach in, e.g., [**57, 113, 128**], *which uses flatness to produce separable decompositions of ρ and thus upper bounds on $\mathrm{rank}_{\mathrm{sep}}(\rho)$.*

7.3. Extensions to other matrix factorizations

Using separable factorization as a basis, we explore three other factorizations. The approach we have described thus far (and in Section 3.1) generalizes straightforwardly to these three new settings.

First, however, we would like to state that we have assumed the bipartite states are symmetric in the sense that

$$\rho \in \mathbb{C}^{n_1} \otimes \mathbb{C}^{n_2},$$

with $n_1 = n_2$. This was done to not unnecessarily complicate the notation and exposition in Section 7.1, but the treatment clearly applies to the general case with $n_1 \neq n_2$. We now break from this convention and will shortly see an example where $n_1 \neq n_2$.

Multipartite quantum states. As opposed to bipartite states, we consider *m-partite quantum states*. Here, $\rho \in \mathbb{C}^{n_1} \otimes \mathbb{C}^{n_2} \otimes \cdots \otimes \mathbb{C}^{n_m}$, and separability means that ρ belongs to the cone

$$\text{cone}\{\mathbf{x}^{(1)}(\mathbf{x}^{(1)})^* \otimes \cdots \otimes \mathbf{x}^{(m)}(\mathbf{x}^{(m)})^* : \mathbf{x}_1 \in \mathbb{C}^{n_1}, \dots, \mathbf{x}_n \in \mathbb{C}^{n_m}, \|\mathbf{x}^{(i)}\| = 1 \ (i \in [m])\}.$$

All the previous results of Section 7.1 have analogous results for this setting, with provisions made for the new variables. Practical computations in this setting become much harder because the matrices ρ are much larger than their bipartite counterparts.

Mixed (bipartite) states. In this setting, we consider factorization into *mixed states* as opposed to pure states, i.e., factorization of the form $\rho = \sum_{\ell \in [r]} A_\ell \otimes B_\ell$ with $A_\ell, B_\ell \in \mathcal{H}^n_+$. Then, the *mixed separable rank* of ρ is defined as

$$\text{rank}_{\text{mixsep}}(\rho) := \min\Big\{r \in \mathbb{N} : \rho = \sum_{\ell \in [r]} A_\ell \otimes B_\ell; \quad A_\ell, B_\ell \in \mathcal{H}^n_+\Big\}.$$

This notion has been considered, e.g., in [**46, 47, 57**] and mixed separable decompositions are called S-decompositions in [**128**] (which deals with real states).

To define bounds on the mixed separable rank, one can follow the same approach as in Section 7.1, but with more variables. Indeed, we now need variables $\mathbf{x} = (x_{ij})_{1 \le i \le j \le n}$ and $\mathbf{y} = (y_{ij})_{1 \le i \le j \le n}$ to model the entries of the matrices $A_\ell \in \mathcal{H}^n_+$ and $B_\ell \in \mathcal{H}^n_+$, while we previously only needed variables $(x_i)_{i \in [n]}$ and $(y_i)_{i \in [n]}$ to model the vectors $\mathbf{a}_\ell \in \mathbb{C}^n$ and $\mathbf{b}_\ell \in \mathbb{C}^n$. Additionally, the corresponding Hermitian matrices $X = (x_{ij})^n_{i,j=1}$ and $Y = (y_{ij})^n_{i,j=1}$ are taken to be positive semidefinite. One may again scale the variables so that they satisfy a boundedness condition such as $|x_{ij}|, |y_{ij}| \le \sqrt{\rho_{\max}}$. This enables one to design hierarchies of lower bounds that converge to the mixed separable analog of the parameter $\tau_{\text{sep}}(\rho)$. The details are analogous and thus omitted.

Specialization to bipartite real states. Here, we are given a real symmetric bipartite state $\rho \in \mathcal{S}^n \otimes \mathcal{S}^n$, where $\mathcal{S}^n$ is the set of real symmetric $n \times n$ matrices. The state ρ is called *real separable* if it admits a decomposition like (7.2) with all vectors $\mathbf{a}_\ell, \mathbf{b}_\ell \in \mathbb{R}^n$ real-valued. The smallest r for which such a decomposition exists is called the *real separable rank*, denoted $\text{rank}^{\mathbb{R}}_{\text{sep}}(\rho)$. Note that a real state can be separable but not real separable; this is the case for the state Sep3 discussed in Section 7.4.

Analogously to the complex case, one can define a parameter $\tau^{\mathbb{R}}_{sep}(\rho)$ and a hierarchy $\xi_t^{\text{sep},\mathbb{R}}(\rho)$ ($t \in \{2, 3, ..., \infty\}$) of bounds. The result of Lemma 7.4 has an analog for these real parameters. In particular, we again have

$$\xi_2^{\text{sep},\mathbb{R}}(\rho) \leq \xi_3^{\text{sep},\mathbb{R}}(\rho) \leq \cdots \leq \xi_\infty^{\text{sep},\mathbb{R}}(\rho)$$
$$\lim_{t\to\infty} \xi_t^{\text{sep},\mathbb{R}}(\rho) = \xi_\infty^{\text{sep},\mathbb{R}}(\rho) = \tau^{\mathbb{R}}_{\text{sep}}(\rho).$$

The most significant difference is that we now replace the complex conjugate with the real transpose operation and work with linear functionals L acting on the real polynomial space $\mathbb{R}[\mathbf{x}, \mathbf{y}]_{2t}$. So the parameter $\xi_t^{\text{sep},\mathbb{R}}$ reads

$$\begin{aligned}\xi_t^{\text{sep},\mathbb{R}}(\rho) := \inf \Big\{ L(1) \mid\ & L : \mathbb{R}[\mathbf{x}, \mathbf{y}]^*_{2t}, \\ & L(\mathbf{x}\mathbf{x}^T \otimes \mathbf{y}\mathbf{y}^T) = \rho, \\ & L([\mathbf{x}, \mathbf{y}]_t[\mathbf{x}, \mathbf{y}]_t^T) \succeq 0, & (7.23a) \\ & L((\sqrt{\rho_{\max}} - x_i^2)[\mathbf{x}, \mathbf{y}]_{t-1}[\mathbf{x}, \mathbf{y}]^T_{t-1}) \succeq 0 \ (i \in [n]), & (7.23b) \\ & L((\sqrt{\rho_{\max}} - y_i^2)[\mathbf{x}, \mathbf{y}]_{t-1}[\mathbf{x}, \mathbf{y}]^T_{t-1}) \succeq 0 \ (i \in [n]), & (7.23c) \\ & L((\rho - \mathbf{x}\mathbf{x}^T \otimes \mathbf{y}\mathbf{y}^T) \otimes [\mathbf{x}, \mathbf{y}]_{t-2}[\mathbf{x}, \mathbf{y}]^*_{t-2}) \succeq 0 \Big\}. & (7.23d)\end{aligned}$$

By removing the complex conjugates, we end up with much smaller matrices in the SDP. Moreover, we can also apply a variant of block-diagonalization to reduce the involved matrices' size further.

Real block-diagonalization. Since the terms of the polynomials involved in the constraints of the above program have even degree in $\mathbf{x}$ (resp., $\mathbf{y}$), we may assume that the variable L satisfies the condition

$$L(\mathbf{x}^\alpha \mathbf{y}^\beta) = 0 \ \text{ if } |\alpha| \text{ or } |\beta| \text{ is odd.} \tag{7.24}$$

This is the real analog of the complex case's condition (7.18). By adding the condition (7.24) to (7.23) we can replace the PSD constraint matrices in (7.23) with block-diagonal matrices. The key to this insight (similar to what was done in Section 7.2) is that one can take the index set of $M_t(L) = L([\mathbf{x}, \mathbf{y}]_t[\mathbf{x}, \mathbf{y}]_t)^T$ in (7.23a)

$$I^t := \big\{(\alpha, \beta) \in (\mathbb{N}^n)^2 : |\alpha + \beta| \leq t\big\}$$

and partition it by the sets

$$I^t = \bigcup_{a,b\in\{0,1\}} I^t_{a,b} \ ; \ I^t_{a,b} := \big\{(\alpha, \beta) \in I^t : |\alpha| \equiv a, \ |\beta| \equiv b \text{ modulo } 2\big\}.$$

Using this partition, the matrix $M_t(L)$ becomes block-diagonal with only the principal submatrices $M_t(L)[I^t_{a,b}]$ ($a, b \in \{0, 1\}$) being nonzero. Thus, $M_t(L) \succeq 0$ if and only if $M_t(L)[I^t_{a,b}] \succeq 0$ ($a, b \in \{0, 1\}$).

A mutatis mutandis argument will work for showing block-diagonalization for the matrices in (7.23b) and (7.23c) using the index set $I^{t-1} = \cup_{a,b\in\{0,1\}} I^{t-1}_{a,b}$. Similarly, we can block-diagonalize the matrix in (7.23d) using the following partition of its index set: $[n^2] \times I^{t-2} = \cup_{a,b\in\{0,1\}}([n^2] \times I^{t-2}_{a,b})$.

7.4. Numerical results and examples

We now illustrate the behavior of the bounds $\xi_t^{\mathrm{sep}}(\rho)$ and $\xi_t^{\mathrm{sep},\mathbb{R}}(\rho)$ for different choices of localizing constraints, at levels $t = 2, 3, 4$, respectively; see Tables 2, 3, and 4.

Computations were made in Windows using Julia [**20**], JuMP [**59**], and MOSEK [**5**] with hardware specifications: i7-8750 CPU with 32 Gb Memory. [1]

Examples of states ρ from the literature. For our examples, we will use the separable states Sep1, Sep2, Sep3, and the entangled state Ent1 that we describe now. For numerical stability, we do the computations with a scaling of these states so that they have trace equal to 1. We present the examples in matrix form with lines drawn to indicate the block structure $\rho = \big((\rho_{ij,i'j'})_{j,j'\in[d_2]}\big)_{i,i'\in[d_1]}$. Zero-valued entries are left blank for easier viewing.

$$\mathrm{Sep1} := \left[\begin{array}{cc|cc} 1 & & & \\ & & & \\ \hline & & & \\ & & & 1 \end{array}\right] ; \ \mathrm{Sep2} := \left[\begin{array}{cc|cc} 2 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 2 \end{array}\right]$$

$$\mathrm{Sep3} := \left[\begin{array}{ccc|ccc} 4 & & & & & \\ & 4 & 2 & & & 2 \\ & 2 & 2 & 1 & -1 & \\ \hline & & 1 & 2 & 1 & -1 \\ & & -1 & 1 & 5 & 1 \\ & 2 & & -1 & 1 & 2 \end{array}\right] ; \ \mathrm{Ent1} = \left[\begin{array}{ccc|ccc|ccc} 1 & & & & 1 & & & & 1 \\ & 2 & & 1 & & & & & \\ & & \frac{1}{2} & & & & 1 & & \\ \hline & 1 & & \frac{1}{2} & & & & & \\ 1 & & & & 1 & & & & 1 \\ & & & & & 2 & & 1 & \\ \hline & & 1 & & & & 2 & & \\ & & & & & 1 & & \frac{1}{2} & \\ 1 & & & & 1 & & & & 1 \end{array}\right].$$

The separable states Sep1, Sep2, and Sep3 were previously studied for example in [**31**]. The entangled state Ent1 was constructed by Choi in [**35**] as the first example in dimension $(n_1, n_2) = (3, 3)$ of an entangled state ρ that satisfies the PPT condition (Section 7.1.1).

In addition, we revisit four examples taken from [**57**]. They include three separable states, named here Sep4, Sep5 and Sep6, corresponding to Examples 3.8, 3.10 and 3.11 in [**57**], and one entangled state, named here Ent2, corresponding to Example 3.9 in [**57**].

[1] The code is available at: `https://github.com/JAndriesJ/sep-rank`

Different SEP factor scalings. In Section 7.1 we provided three different choices of localizing constraints in (7.9), (7.15) and (7.14), that we denote here as S1, S2 and S3, respectively.

The examples show that the different choices lead to incomparable bounds. Let us use the notation S1 $<$ S2 as short hand for "there exists a ρ such that ξ_t^{sep} (using scaling S1) $<$ ξ_t^{sep} (using scaling S2)". Then, at level $t = 2$, the state Sep1 demonstrates both S3 $<$ S1 and S2 $<$ S1, and, at level $t = 3$, Sep2 demonstrates both S2 $<$ S3 and S1 $<$ S3 and Sep3 demonstrates both S1 $<$ S2 and S3 $<$ S2. A case where the various constraints differ in ability to detect entanglement is provided by the state Ent1 at level $t = 2$. On the other hand, for the state Ent2, all three scalings detect entanglement at level $t = 2$. Thus we have detection at the same level as for the approach followed in [**57**].

In addition, we show in Figure 1 a scatter plot of the bound $\xi_3^{\mathrm{sep}}(\rho)$ vs. its computation time in seconds for 100 random complex matrices ρ grouped and colored by the respective scalings S1, S2 and S3. These matrices are defined by

$$\rho = \sum_{j=1}^{5} \mathbf{a}^{(j)}(\mathbf{a}^{(j)})^* \otimes \mathbf{b}^{(j)}(\mathbf{b}^{(j)})^*,$$

where $\mathbf{a}^{(j)}, \mathbf{b}^{(j)} \in \mathbb{C}^3$ are random vectors whose entries are of the form $\mathbf{x} + \sqrt{-1}\mathbf{y}$, with $\mathbf{x}, \mathbf{y} \in \mathcal{N}(0,1)$, i.e., the entries of $\mathbf{x}$ (resp., $\mathbf{y}$) are sampled from the Gaussian distribution with mean 0 and variance 1. We also normalize the trace here for numerical stability. This construction guarantees separability and provides the upper bound $\mathrm{rank}_{\mathrm{sep}}(\rho) \leq 5$. Such states also satisfy the reverse inequality $\mathrm{rank}_{\mathrm{sep}}(\rho) \geq 5$ almost surely since $\mathrm{rank}(\rho) = 5$ almost surely. We use this class of examples merely to test the quality of the bounds.

From the figure, we can draw the following observations: First, the bounds are concentrated around the means 2.7, 3.4, and 3.3 for the scalings S1, S2, and S3, respectively. Second, in this class of examples, the S1 rescaling yields inferior bounds compared to S2 and S3. Third, out of the hundred examples and for the three different scalings considered, no bound exceeded the value 4.

Separable but not real separable states. As mentioned in Section 7.3, there exist real states $\rho \in \mathcal{S}^n \otimes \mathcal{S}^n$ that are separable but do not admit a decomposition using real vectors $\mathbf{a}_\ell, \mathbf{b}_\ell \in \mathbb{R}^n$.

Our bound $\xi_2^{\mathrm{sep},\mathbb{R}}$ provides a proof of the latter for the state Sep3: its real separable rank is infinity since our lower bound is infeasible (i.e., there exists a dual certificate that proves $\mathrm{rank}_{\mathrm{sep}}^{\mathbb{R}}(\mathrm{Sep3}) = \infty$).

Finally, we note that one sometimes needs to go beyond level $t = 2$ (and thus beyond the PPT criterion) to reveal entanglement: with the localizing constraints S3, the bound for Ent1 is feasible at $t = 2$, but infeasible at $t = 3$. Going to a higher level naturally increases the size of the SDP. For the examples Sep3, Sep4, Sep6, and Ent1, this prevented us from computing level $t = 4$.

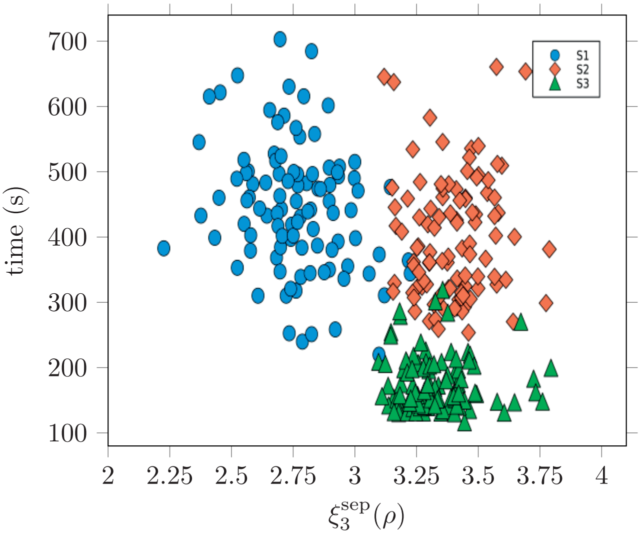


FIGURE 1. Scatter plot of $\xi_3^{\mathrm{sep}}(\rho)$ vs computation time (sec.) for 100 random matrices, grouped and colored by rescalings S1, S2 and S3.

TABLE 2. Examples and numerical bounds level $t = 2$

ρ	(n_1, n_2)	bi-r	$\xi_2^{\mathrm{sep}}(\rho)$			$\xi_2^{\mathrm{sep},\mathbb{R}}(\rho)$			$\mathrm{r_{sep}}$	time
			S1	S2	S3	S1	S2	S3		
Sep1[**31**]	(2,2)	(2,2)	**2.0**	1.0	1.0	**2.0**	1.0	1.0	2	< 1
Sep2[**31**]	(2,2)	(3,3)	1.421	1.0	1.0	1.421	1.0	1.0	3	< 1
Sep3[**31**]	(2,3)	(4,6)	1.333	1.0	1.0	*	*	*	6	< 1
Sep4[**57**]	(3,3)	(2,2)	1.0	1.0	1.0	1.0	**1.953**	1.0	2	< 1
Sep5[**57**]	(2,2)	(4,4)	1.069	1.0	1.0	N/A	N/A	N/A	≤ 7	< 1
Sep6[**57**]	(3,3)	(7,7)	1.053	1.0	1.0	N/A	N/A	N/A	≤ 9	< 1
Ent1[**35**]	(3,3)	(4,4)	2.069	*	1.525	2.069	*	1.525	∞	< 1
Ent2[**57**]	(2,2)	(2,4)	*	*	*	N/A	N/A	N/A	∞	< 1

Run time given in seconds

bi-r : birank(ρ)

$\mathrm{r_{sep}}$: $\mathrm{rank_{sep}}(\rho)$

* : Infeasibility certificate returned

- : Solver could not reach a conclusion (not a memory error)

N/A : Not Applicable

We indicate using boldface when a bound (after rounding up) equals the separable rank

TABLE 3. Examples and numerical bounds level $t = 3$

ρ	(n_1, n_2)	bi-r	$\xi_3^{\mathrm{sep}}(\rho)$			$\xi_3^{\mathrm{sep},\mathbb{R}}(\rho)$			$\mathrm{r_{sep}}$	time
			S1	S2	S3	S1	S2	S3		
Sep1	(2,2)	(2,2)	**2.0**	**2.0**	**2.0**	**2.0**	**2.0**	**2.0**	2	< 1
Sep2	(2,2)	(3,3)	1.909	2.0	**2.178**	1.909	2.0	**2.178**	3	2
Sep3	(2,3)	(4,6)	2.423	3.0	2.790	*	*	*	6	25
Sep4	(3,3)	(2,2)	**1.652**	**2.0**	-	**1.65**	**2.0**	1.0	2	261
Sep5	(2,2)	(4,4)	1.988	2.048	2.079	N/A	N/A	N/A	≤ 7	4
Sep6	(3,3)	(7,7)	2.715	3.326	-	N/A	N/A	N/A	≤ 9	290
Ent1	(3,3)	(4,4)	-	-	*	-	*	*	∞	67

TABLE 4. Examples and numerical bounds level $t = 4$

ρ	(n_1, n_2)	bi-r	$\xi_4^{\mathrm{sep}}(\rho)$			$\xi_4^{\mathrm{sep},\mathbb{R}}(\rho)$			$\mathrm{r_{sep}}$	time
			S1	S2	S3	S1	S2	S3		
Sep1	(2,2)	(2,2)	**2.0**	**2.0**	**2.0**	**2.0**	**2.0**	**2.0**	2	105
Sep2	(2,2)	(3,3)	**3.0**	**3.0**	**3.0**	**3.0**	**3.0**	**3.0**	3	332
Sep5	(2,2)	(4,4)	4.0	4.0	4.0	N/A	N/A	N/A	≤ 7	161

Discussion

A link between completely positive and nonnegative rank. For a matrix $M \in \mathbb{R}_+^{n\times m}$ its nonnegative rank can be seen as

$$\text{rank}_+(M) = \min\left\{\text{rank}_{\text{cp}}\begin{pmatrix} X & M \\ M^T & Y \end{pmatrix} : X \in \mathcal{S}^n,\ Y \in \mathcal{S}^m\right\}.$$

Hence, nonnegative rank can be recast as a (completely positive) matrix completion problem. However, this line of attack does not appear to contribute much.

Non-commutative matrix ranks. We have explored applications of the moment method to nonnegative and completely positive matrix factorization ranks. In Section 7.3, we discussed the mixed-separable rank and, in so doing, hinted at a generalization to noncommutative ranks.

The non-commutative analogs of CP rank and NN rank, namely the positive semidefinite (PSD) rank and the completely positive semidefinite (CPSD) rank, introduced in Section 4.5, can be attacked using similar techniques to what we have discussed thus far, as was done in [**80**].

In the noncommutative setting of the PSD rank, if the matrix M has a PSD factorization

$$M_{i,j} = \langle A_i, B_j\rangle, \text{ for } i \in [n] \text{ and } j \in [m],$$

for some $A_1, ..., A_n, B_1, ..., B_m \in \mathcal{S}_+^r$, then the zero entries of M also imply ideal-type constraints of the form $A_iB_j = 0$. Thus the techniques of ideal sparsity may extend to this general setting. We leave this extension to future work.

Tensor ranks. Hierarchies of moment-based relaxations have also been employed to obtain sequences of bounds for the rank of tensors [**153**], as well as for the symmetric nuclear norm of tensors [**127**]. We hope that our exposition on the separable rank and its generalizations in Section 7.3 gave the reader an inkling of what is possible with the moment approach.

Disadvantages of atom extraction in the ideal-sparse setting. On the surface, it may seem less likely that each of the p-many moment matrices satisfies the flatness condition (3.18) simultaneously than simply requiring that flatness holds on the single dense moment matrix. This was not an issue in most of our computations (recall Table 3)). However, we have no proper motivation for why this would hold in general GMPs.

Because atom extraction is carried out independently on each linear functional L_k, it is possible for one to end up with multiples of the same atom across different

L_k's. Indeed, suppose that $\mathbf{x}'$ is an atom with support contained in both V_{k_1} and V_{k_2}, then it is possible that the atom extraction process of Section 3.2.2 would produce $\mathbf{x}'$ for both L_{k_1} and L_{k_2}. This may seem relatively rare, but it has been observed in our computations of Table 3.

Advantages of atom extraction in the ideal-sparse setting. The atoms one attains are sparse by construction. This is clear when one recalls that an atom $\mathbf{x}'$ of L_k must have support contained in V_k, i.e. $\mathbf{x}'_i = 0$ for all $i \notin V_k$. The atoms recovered via the dense GMP formulation tend to have small non-zero entries, culminating in inaccurate factorization. Even if one tries to clean up the factors with rounding, the approximations are still poorer than their sparse counterpart.

Because each moment matrix $L_k([\mathbf{x}(V_k)]_{s_k}[\mathbf{x}(V_k)]_{s_k}^T)$ can be handled independently of each other, the process is well suited for parallel computing. Though, atom extraction is seldom the computational bottleneck in solving GMPs.

More general ideal sparsity and applications. We have considered an ideal sparsity structure, where the ideal in (2.8) is generated by monomials. Beside their use for bounding matrix factorization ranks, constraints of the form $x_i x_j = 0$ naturally arise in a number of other applications. First, we note that up to a change of variables, one can consider more general constraints of the form $(a^\top x + b)(c^\top x + d) = 0$. This type of constraint is commonly referred to as a *complementarity constraint*, where either the term $(a^\top x + b)$ or the term $(c^\top x + d)$ is required to be zero. We mention two areas where such complementary constraints naturally arise: analysis of neural networks and optimality conditions in optimization.

Complementarity constraints arise naturally when modeling neural networks with the rectified linear activation functions (ReLU). The semialgebraic representation of the graph of the ReLU function involves a constraint of the form $y(y - x) = 0$, which is exactly a complementarity constraint. The fact that the graph of the ReLU function admits a semialgebraic representation has been exploited computationally using the moment-sum-of-squares framework for analyzing the Lipschitz constant of the neural network as well as stability and performance properties of dynamical systems controlled by the ReLU neural networks, see, e.g., [**33, 34, 98**]. Ideal sparsity is, therefore, a natural candidate to render these methods more computationally efficient and would deserve further study.

Complementarity systems also arise in optimization within the Karush-Kuhn-Tucker (KKT) conditions. The complementarity slackness of the KKT condition reads $\lambda_i f_i(x) = 0$, where λ_i is the Lagrange multiplier associated to the i^{th} constraint $f_i(x) \leq 0$. If f_i is affine, this is in the form of ideal constraints. The KKT conditions form a basic semialgebraic set when the optimization problem has polynomial data was exploited in [**99**] to analyze dynamical systems controlled by optimization algorithms, albeit without exploiting the ideal sparsity. More generally, the ideal sparsity could be used to analyze the *linear complementarity problems* (LCP) that have applications in, e.g., economics, engineering, or game theory; see [**37**] for an extensive treatment of the subject.

Part 3

Convex scalarizations in portfolio selection

In Chapter 8, we give some background on the portfolio optimization problem in finance. In particular, we introduce the Markowitz model and several extensions: higher-order moments, shorting and leverage, and sparse portfolios. This leads to the mean-variance-skewness-kurtosis (MVSK) problem with possible sparse variants. We also introduce some general theory concerning multi-objective optimization problems. There, we discuss the link between multi-objective problems and their scalarizations.

In Chapter 9, we mathematically formalize the MVSK problem as a multi-objective optimization problem. We attack the MVSK problem via linear scalarization. We characterize sufficient conditions on the scalarizing hyper-parameter defining the scalarization to result in a convex optimization problem. We show that a large class of scalarizations results in convex (single-objective) optimization problems, which are amenable to first-order optimization methods. Analogous results are shown for a sparse variant of the MVSK problem. Hence, we partially recover the Pareto set of MVSK by solving different scalarized MVSK problems.

In Chapter 10, we collect our numerical experiments and methodology supporting the theory built in Chapter 9. To solve the scalarized MVSK problems, we use the well-known optimization algorithm *fast iterative shrinkage-thresholding algorithm* (FISTA) (Section 10.1). We visualize and compare the resulting approximate Pareto sets for various domains (simplex and cube) with and without sparsity. We observe that there are some points that provide a better trade-off among the four objectives. We call such points *solutions of superior trade-off*. They are described and visualized in Section 10.2.

This part of the thesis is based on the work in [**150**].

CHAPTER 8

The portfolio selection problem

8.1. Background

In finance, the portfolio selection problem is the task of selecting a subset of assets (called a portfolio) from a pool of available assets in such a way as to maximize the appreciation of the selection's value while minimizing the risk of losing the initial capital investment, see [**19**]. In 1952, Markowitz [**118**] created the first mathematical formulation of this problem. Since then, models have evolved into various directions, each trying to capture some aspect of the practical problem, e.g., transaction cost [**112, 121**], non-Gaussian data [**149, 141, 152**], short selling and leveraging [**119**]. Regardless of the model used, the resulting problem should not be so difficult that available computers cannot solve it (at least approximately) in a reasonable time. To deal with computational difficulties, new techniques have been developed, in particular, in [**103, 117, 121**].

The Markowitz model. Markowitz modeled a portfolio's profitability by the *mean returns* and its risk by using the *variance* as a proxy. The model can be seen as a bi-objective optimization problem

$$\begin{aligned} &\max\ w^T M \\ &\min\ w^T V w \\ &\text{s.t. } w \in \Delta^n, \end{aligned} \tag{8.1}$$

where Δ^n is the standard simplex, $V \in \mathcal{S}_+^n$ is a covariance matrix, and $M \in \mathbb{R}^n$ is the vector of means. Here, n denotes the number of assets that are available for selection and, for each $i \in [n]$, w_i denotes the weight of the i^{th} asset in the portfolio w. The values of M and V are known or computed from some available data. Problem (8.1) can be converted into a single-objective optimization problem of the form:

$$\begin{aligned} &\min\ (1-\lambda) w^T V w - \lambda w^T M \\ &\text{s.t. } w \in \Delta^n, \end{aligned} \tag{8.2}$$

for some hyper-parameter $\lambda \in [0,1]$ modelling the investors risk tolerance. Hence, the two conflicting objectives, maximizing the mean returns $w^T M$ and minimizing the variance $w^T V w$, are pitted linearly against each other.

Extending to higher-order moments. Problem (8.1) is often called the *mean-variance model*, as it only uses the means (i.e., $M \in \mathbb{R}^n$) and the variance (i.e., $V \in \mathcal{S}_+^n$) in its description. In statistics, the mean and variance are the data's first and second moments. Vastly different distributions can have identical means and variances. Hence, by only using these moments, the model implicitly assumes that higher-order moments like skewness and kurtosis are not important. This assumption is financially hazardous if the market data is not Gaussian distributed, which has been shown to be the case in practice; see [**141, 149**]. The danger comes from underestimating the frequency of extreme events, like rare but significant losses. We mention two ways of addressing the problem.

The first way is to combine or reject the covariance-based risk model in favor of other methodologies like mutual information or entropy-based risk, thereby removing the Gaussian assumption, see, e.g., [**79, 152**]. We also class asymmetric risk models like Value at Risk (VaR) in this category, see, e.g., [**38**].

The second approach extends the model to include higher-order statistical moments, allowing for more varied data distributions. Publications that propose extending the model to include higher-order moments like *skewness* and *kurtosis* include [**101, 174, 120, 95, 147, 117, 92**].

Skewness, the third data moment, represents the asymmetric characteristics of a distribution. One can think of a distribution leaning in a particular direction, the skewness quantifying the direction and intensity of the leaning, see Figure 1.

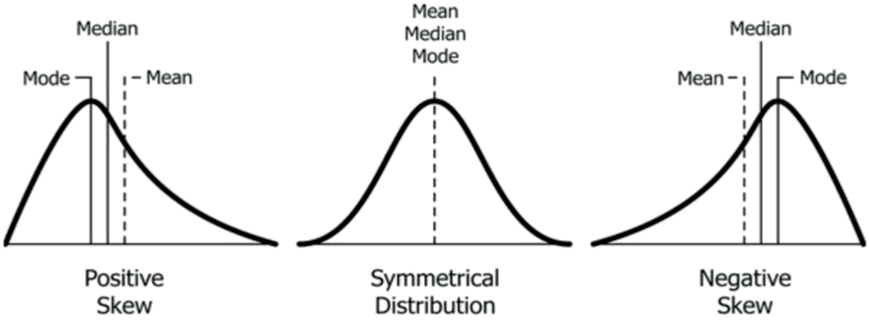


FIGURE 1. Diagram of the probability density function of a Gaussian with skewness, taken from [**56**].

Kurtosis, the fourth data moment, is similar to variance in that larger values correspond to a sharper peak and fatter tails, i.e., more extreme returns on either side of the mean, compared with the normal distribution, see Figure 2.

Most investors would prefer a large positive skewness and a small kurtosis if given a choice. Adding these new terms improves the model's expressiveness at the cost of adding more complexity. Skewness, in particular, is likely non-convex (more on this in Section 9.1).

The extended model (now including skewness and kurtosis) is called the *mean-variance-skewness-kurtosis* (MVSK) problem. It is a *multi-objective optimization problem* (MOOP) with the first four moment functions as objectives, see (9.2) and

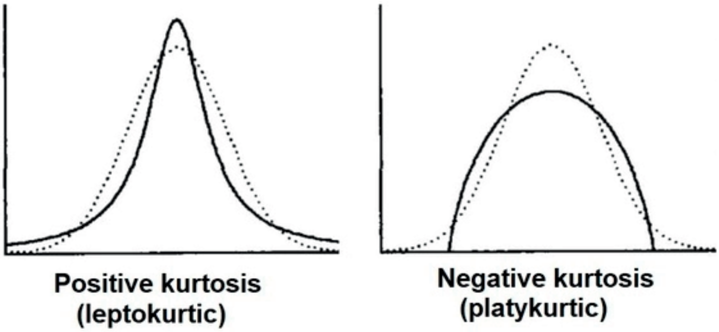


Figure 2. Diagram of distributions with positive and negative kurtosis compared to the normal distribution (dashed line), taken from [**48**].

(9.3) in Section 9.1 for the formal definitions. Finding solutions for the MVSK will be our primary task.

Leverage and shorting. We consider the option to hold *shorted* and *leveraged* positions; this means that the portfolio can consist of borrowed assets.

In brief, *short selling* (or "holding a shorted position") is an investment strategy where one borrows an asset, speculating that the asset will soon appreciate in value. One then sells the asset before it depreciates and repurchases it for a profit after it has depreciated. The repurchased asset and a premium are then returned to the original owner. Mathematically, negative portfolio weights can model this.

Leverage is an investment strategy that uses credit to bolster assets, magnifying potential profits and losses. Mathematically this means that the portfolio weights can sum up to more than one, i.e., we no longer optimize over the simplex but rather over a bounded cube.

For our intents and purposes, shorting and leverage are expressed in the feasible regions of the optimization problems we consider (see the beginning of Section 9.1). We contribute little to the topic in this regard other than showing the compatibility of our approach to both these settings. For a general overview of financial terms, we refer the reader to any standard text like [**119**].

Sparse portfolios via cardinality constraints. A portfolio $w \in \mathbb{R}^n$ is *sparse* if it supports fewer assets than the selection pool allows, i.e., if $|\{i \in [n] : w_i \neq 0\}| < n$. Given two equally well-performing portfolios, one often prefers the sparser portfolio to dense portfolios (where all weights are non-zero) because having fewer assets to manage leads to fewer transaction costs [**19**].

Transaction costs (fees paid to brokers to purchase and sell assets) often undermine the profitability of portfolios. Investors manage these costs in one of two ways.

First, explicitly modeling the cost as a type of penalty in the objective, i.e., as a quantity to be minimized; see, for example, [**112**].

Second, by imposing cardinality constraints, i.e., requiring that any solution has support of some bounded size. This adds to the difficulty of optimizing the portfolio, as cardinality constraints are often combinatorial. Authors like [**121**] have attacked such problems using penalized alternating direction methods. These methods break the problem at the constraints into two coupled sub-problems, each capturing a different half of the original problem. For example, the domain constraints (e.g., solutions must belong to the simplex) and the support constraint (e.g., the solution support may not exceed some fixed $k \in \mathbb{N}$) could be separated into respective sub-problems. Iteratively solving and alternating between these sub-problems, one obtains a sequence of solutions that, under convexity conditions, converge to a globally optimal solution.

Our work is also in the spirit of cardinality restriction. We consider sparse variants of MVSK in Section 9.1. We show that in our setting, one often attains sparse solutions by projecting onto the simplex as part of using projected gradient descent to compute an optimum (see Section 10.1). Suppose our projection approach fails (i.e., the solution is not sparse enough). In that case, we impose the support restriction by "splitting" the domain and using a heuristic (based on the solution of the original problem) to search over the resulting parts (see Section 10.1.1).

A disclaimer on variance-based models for portfolio selection. The saying "all models are wrong, but some are useful" is worth repeating in this section. We would like to explicitly state that this part of the thesis is not intended to advocate the use of the MVSK model in portfolio selection, as this would fall in the domain of economics. Our core message is that if one is interested in the MVSK model, then the problem has convexity properties that are, to the best of our knowledge, untapped.

8.2. Multi-objective optimization problems

In addition to inheriting the difficulties of single-objective optimization problems, *multi-objective optimization problems* (MOOPs) have new challenges to address, see, e.g., [**61**] for background. Consider the general MOOP:

$$\begin{aligned} \min f(x) &:= \big(f_1(x), f_2(x), ..., f_p(x)\big) \\ \text{s.t.}\ \ & x \in X, \end{aligned} \tag{8.3}$$

where $f_1, f_2, ..., f_p$ are some scalar-valued functions defined on $\mathbb{R}^n$, and $X \subseteq \mathbb{R}^n$.

How one defines optimality is the first change from single to multiple objectives. Real numbers are well ordered by $\leq$, and as such, it is clear when one solution gives a better objective value than another. In contradistinction, the values of MOOPs are real-valued vectors, and thus they are only partially ordered by $\leq$, applied entrywise between two vectors (of equal size). Optimal solutions to MOOPs are hence only optimal in the sense of not being strictly worse than any other solution vector.

Formally we define a partial order on vectors $v, w \in \mathbb{R}^p$:

$$\begin{aligned} v \geq w &\iff v_i \geq w_i \quad (i \in [p]), \\ v \gneqq w &\iff v \neq w,\ v_i \geq w_i \quad (i \in [p]), \\ v > w &\iff \ v_i > w_i \quad (i \in [p]). \end{aligned} \tag{8.4}$$

A point $x \in X$ is said to be *Pareto optimal* for (8.3) if there exists no $y \in X$ such that

$$f(x) = (f_1(x), f_2(x), ..., f_p(x)) \gneqq (f_1(y), f_2(y), ..., f_p(y)) = f(y). \tag{8.5}$$

Similarly, a point $x \in X$ is said to be *locally Pareto optimal* for (8.3) if it is Pareto optimal in some open neighborhood of x. The *Pareto front* of (8.3) is defined as the set of all Pareto optimal solutions of (8.3). The following is a well-known fact.

LEMMA 8.1. *Consider the MOOP (8.3), and assume that the objectives $f_1, f_2, ..., f_p$ are all convex functions and that the domain X is a convex set. Then, any local Pareto optimal point x of (8.3) is also (globally) Pareto optimal.*

PROOF. Suppose by way of contradiction that x is not globally Pareto optimal and let $y \in X$ be a point such that $f(y) \lneqq f(x)$. Take any $t \in (0, 1)$ and observe that via convexity we have

$$f(ty + (1 - t)x) \leq tf(y) + (1 - t)f(x) \lneqq tf(x) + (1 - t)f(x) = f(x).$$

Since this holds for arbitrarily small positive values of t, it holds that x^* is not locally Pareto optimal, contradicting our initial assumption. □

Scalarized multi-objective optimization problems. Among the several approaches to optimizing a MOOP, we will look for optimizers via *scalarizations* of the MOOP. Scalarization is a well-known approach that converts a MOOP into a single objective optimization problem called the scalarized problem. Several authors have done this for MVSK by encoding some objectives as constraints, see, e.g., [**117**]. One downside of this approach is that one must make an a priori estimate of these objectives. Alternatively, one can scalarize by combining the multiple objectives into a single scalar-valued objective function. We follow this approach. For the MVSK problem, the literature predominantly considers two scalarizations. The first is the Minkowski scalarization, as seen in [**103, 120, 6**]. Here one first computes the optimal value for each of the objectives independent of the others

$$f_i^* := \min_{x \in X} f_i(x) \quad (i \in [p]).$$

Using these independent optima one constructs, for some positive user-defined hyper-parameter $\lambda \in \mathbb{R}^p$, the Minkowski distance scalarization is as follows:

$$\min_{x \in X} \sum_{i \in [p]} \left| f_i(x) - f_i^* \right|^{\lambda_i}. \tag{8.6}$$

We elaborate more on the Minkowski distance scalarization in the discussion at the end of Part 3.

The second scalarization is simply a linear combination of the objectives with the linear weights being some choice of hyper-parameter $\lambda \in \mathbb{R}^p$, see [**95, 94**]. The resulting scalarized optimization problem is hence

$$\min_{x \in X} F_\lambda(x), \tag{8.7}$$

where

$$F_\lambda(x) := \sum_{i \in [p]} \lambda_i f_i(x). \tag{8.8}$$

Note that this scalarization has a linear dependence on hyper-parameters and is also conceptually simple to interpret. We will be using this linear scalarization throughout Part 3. Optimizers of the scalarized problem are not guaranteed to be Pareto optimal for the MOOP, but for neat scalarizations, this is the case. A scalarization is said to be *neat* if any optimal solution x of the scalarized problem is also a Pareto optimal solution of the original MOOP.

LEMMA 8.2 (Proposition 3.9 in [**61**]). *If $\lambda > 0$, then the scalarization (8.7) is neat, i.e., global optimizers of (8.7) are (global) Pareto optimizers of (8.3).*

PROOF. Let $x \in X$ be an optimal solution of (8.7) and suppose by way of contradiction that x is not Pareto optimal for (8.3), i.e., there exists a $y \in X$ such that $f(x) \gneqq f(y)$. Then

$$\sum_{i \in [p]} \lambda_i f_i(x) > \sum_{i \in [p]} \lambda_i f_i(y)$$

because $\lambda_i > 0$ for all $i \in [p]$. Hence, this contradicts the fact that x optimizes (8.7). □

Consider now the case when the feasible set is defined as

$$X := \{x \in \mathbb{R}^n : g_j(x) \geq 0, \ j \in [q]\}, \tag{8.9}$$

for some functions $g_1, g_2, ..., g_q : \mathbb{R}^n \to \mathbb{R}$. Let $J(x) = \{j \in [q] : g_j(x) = 0\}$ denote the index set of active constraints at x. The following result holds for the MOOP (8.3).

THEOREM 8.3 (Theorem 3.25 in [**61**]). *Let X be the set defined in (8.9). Let $f_1, f_2, ..., f_p$, $g_1, g_2, ..., g_q$ be scalar-valued functions that are continuously differentiable at $x^* \in X$. Assume that x^* is a Pareto optimal point of (8.3) and that there is no vector $v \in \mathbb{R}^n$ such that*

$$\langle \nabla f_i(x^*), v \rangle \leq 0 \textit{ for all } i \in [p], \tag{8.10a}$$

$$\langle \nabla f_k(x^*), v \rangle < 0 \textit{ for some } k \in [p], \tag{8.10b}$$

$$\langle \nabla g_j(x^*), v \rangle \leq 0 \textit{ for all } j \in J(x^*). \tag{8.10c}$$

Then, there exist vectors $\lambda \in \Delta^p$ and $\eta \in \mathbb{R}^q$ such that $\lambda > 0$, $\eta \geq 0$, and

$$\sum_{i \in [p]} \lambda_i \nabla f_i(x^*) + \sum_{j \in [q]} \eta_j \nabla g_j(x^*) = 0,$$

$$\sum_{j \in [q]} \eta_j g_j(x^*) = 0.$$

Therefore, x^* is a KKT point of the following scalarization of the problem (8.3):

$$\begin{aligned} \min \quad & F_\lambda(x) := \sum_{i \in [p]} \lambda_i f_i(x) \\ \textit{s.t.} \quad & x \in X. \end{aligned} \tag{8.11}$$

A point $x^* \in X$ that is Pareto optimal and does not satisfy the system (8.10) for any $v \in \mathbb{R}^n$ is also known in the literature as being *properly efficient in the Kuhn-Tucker sense* (see Definition 2.49 in [**61**]).

Proposition 8.4. *Assume the conditions of Theorem 8.3 hold. If, in addition, X is a convex set and F_λ a convex function, then x^* is a global optimizer of (8.11).*

Proof. The claim follows from the fact that any KKT point of a convex problem must be a global optimizer. □

Let us again consider the scalarized problem (8.7) where $F_\lambda(x) = \sum_{i \in [p]} \lambda_i f_i(x)$ for some $0 < \lambda \in \mathbb{R}^p$. Depending on the functions $f_1, ..., f_p$, the hyper-parameter λ, and the domain X, problem (8.7) can still be extremely difficult to solve. However, in the special case when the objective $F_\lambda(x)$ and the domain X are convex (strictly convex), there are efficient methods to find the (unique) minimizer [**15**]. Having found an optimizer to the scalarized problem, Lemma 8.2 relates said optimizer back to a Pareto point of the MOOP.

The core theme of Part 3 is to partially recover the Pareto set of the MVSK problem by solving different linear scalarizations of the MVSK problem. In order to achieve this, we identify classes of hyper-parameters $\lambda \in \Delta^4$ that ensure the resulting scalarization F_λ is convex over the optimization domain (either the standard simplex or the cube).

CHAPTER 9

The MVSK problem

This chapter gives the mathematical formulation of the MVSK optimization problem (9.5) starting from a random variable representing asset price (see Section 9.1). We look at two possible domains of optimization, each motivated by the inclusion or omission of shorting and leveraging of assets. Using asset price, we define the four objectives of the MVSK multi-objective optimization problem. Combining the four objective functions and the chosen feasible region we get the MVSK optimization problem (9.5). After defining and interpreting the MVSK problem, we look at a possible sparse variant of the MVSK problem and give two motivations for it.

In Section 9.2, we consider a linear scalarization (9.7) of the multi-objective (9.5) and analyze the conditions under which the resulting (scalar objective) optimization problem is convex. Convex optimization problems (having a convex objective function and convex feasible region) have the useful property that any local optimizer is also a global optimizer. This is a fact we will make use of in order to recover part of the Pareto front of problem (9.5). Moreover, convex functions are well-studied; they can be efficiently optimized if the gradient is known and one can project onto the feasible region efficiently. See, for example, the standard textbook [**24**].

9.1. Formulating the MVSK

To distinguish the general results of Chapter 8 from the particular setting of the MVSK problem, we now change the notation from a general vector $x \in \mathbb{R}^n$ to a vector of weights, $w \in \Delta^n$ or $w \in [-1,1]^n$.

The domain of optimization. As a variable, we consider a *portfolio*, which consists of a weighted selection of $n \in \mathbb{N}$ assets, represented by $w \in \mathbb{R}^n$. At first, we consider two choices of the domain for portfolios.

We consider the *standard simplex*, where investors cannot short assets nor take leveraged positions (recall the definitions in Chapter 8), i.e.,

$$w \in \Delta^n := \Big\{ w \in [0,1]^n : \sum_{i \in [n]} w_i = 1 \Big\}.$$

Secondly, we consider the *cube*, where we allow short selling and leverage. We assume that there is a bound $B \in \mathbb{R}_+$ on how leveraged a position can be. Mathematically we write

$$w \in [-B,B]^n := \Big\{ w \in \mathbb{R}^n : -B \leq w_i \leq B \ (i \in [n]) \Big\},$$

where we set $B = 1$ for simplicity.

Later we will look at the sparse variants of these domains.

The objective functions. For asset $i \in [n]$ let $\widetilde{R}_i$ denote the *relative return* of asset i, i.e., $\widetilde{R}_i$ is the fractional change in price relative to the initial cost of purchasing the asset i. For our intents and purposes, we consider $\widetilde{R}$ to be a random variable taking values in $\mathbb{R}^n$. Denote the vector of expected returns by

$$M := \mathbb{E}[\widetilde{R}] = (\mathbb{E}[\widetilde{R}_i])_{i\in[n]} \in \mathbb{R}^n, \tag{9.1}$$

and define the *centralized relative returns* as

$$R := \left(\widetilde{R}_i - M_i\right)_{i\in[n]}.$$

The expected return of the portfolio w is given by

$$f_1(w) := M^T w. \tag{9.2}$$

For $k = 2, 3, 4$ we can define the functions

$$f_k(w) := \mathbb{E}\left[\langle R, w\rangle^k\right] \ , \ \langle R, w\rangle^k := (R^T w)^k = \Big(\sum_{i\in[n]} R_i w_i\Big)^k, \tag{9.3}$$

which relate to the second, third, and fourth moments of R as follows:

$$f_2(w) = w^T V w, \ f_3(w) = (w \otimes w)^T S w, \ f_4(w) = (w \otimes w)^T K (w \otimes w), \tag{9.4}$$

where $V := \mathbb{E}[RR^T] \in \mathbb{R}^{n\times n}$ is the *covariance matrix*, $S := \mathbb{E}[(R \otimes R)R^T] \in \mathbb{R}^{n^2\times n}$ is the *skewness matrix*, and $K := \mathbb{E}[(R \otimes R)(R \otimes R)^T] \in \mathbb{R}^{n^2\times n^2}$ is the *kurtosis matrix*, all w.r.t. the data R. With slight abuse of terminology, we refer to $f_2(w)$ as the variance of portfolio w, and similarly, $f_3(w)$ and $f_4(w)$ are called its skewness and kurtosis.

The functions f_1, f_2, and f_4 are convex. We note that f_1, f_2, and f_4 are convex on $\mathbb{R}^n$. Indeed, f_1 is linear and therefore convex. The functions f_2 and f_4 are convex and nonnegative for all $w \in \mathbb{R}^n$ because V and K are PSD. To see why V is PSD observe that V is the expectation of a random variable RR^T, taking PSD matrices as values. Hence, the Hessian of f_2, $H(f_2) = V$, is PSD, and thus f_2 is convex. Similarly, the Hessian $H(f_4)(w)$ of the kurtosis function f_4 at w can be written as

$$H(f_4)(w) = 12 \cdot \mathbb{E}[w^T(RR^T)w \cdot RR^T] \succeq 0,$$

where the PSDness follows from the fact that $w^T(RR^T)w \geq 0$ and $RR^T \succeq 0$ for all $w \in \mathbb{R}^n$.

MVSK optimization problem. Using the objective functions defined in (9.2) and (9.3), and using the simplex as the feasible region, we define the MVSK problem:

$$\begin{aligned} \max \ & f_1(w) \\ \min \ & f_2(w) \\ \max \ & f_3(w) \\ \min \ & f_4(w) \\ s.t. \ & w \in \Delta^n. \end{aligned} \tag{9.5}$$

This program can be interpreted as follows: one wishes to maximize returns while minimizing extreme events like rare but significant losses. In expectation, the "odd" functions f_1 and f_3 correspond to increased returns when positive and losses when negative. The "even" functions f_2 and f_4 describe the spread of returns, with larger values corresponding to more significant fluctuations at the extremes. Note that variance and kurtosis are symmetric, which means they treat extreme profits and losses with equal prejudice.

As discussed in Chapter 8, multi-objective optimization problems have a set of Pareto optimal points. Since each Pareto optimal point is not strictly worse than any other Pareto optimal point, it falls to the investor to choose among these solutions. However, some Pareto solutions provide a better spread among the multiple objectives of (9.5); more on this in Chapter 10.

A sparse variant of MVSK. A sparse portfolio w is one with many of its entries w_i set to zero. Our general sparse version of the problem (9.5) reads as follows:

$$\begin{aligned} \max \quad & f_1(w) \\ \min \quad & f_2(w) \\ \max \quad & f_3(w) \\ \min \quad & f_4(w) \\ s.t. \quad & w \in \Delta^n \\ & \prod_{i \in C} w_i = 0 \text{ for } C \in \mathcal{C}, \end{aligned} \tag{9.6}$$

where $\mathcal{C} \subseteq \mathcal{P}([n])$. We give two motivations for the above form of sparsity.

Reducing transaction costs and management fees. One of the core ideas in portfolio selection, *diversification*, is the principle that buying causally-unrelated assets will protect the investor from rare but significant losses. The idea is that one expects the random depreciation of a single asset to be unrelated (or inversely related) to the value of other assets. Of course, this only holds outside systemic events like economic crises, see [**152**]. To get the benefits of diversification, many assets must often be held in one's portfolio. This creates a new problem, as larger portfolios lead to increase management fees and transaction costs when rebalancing (re-optimizing the portfolio to account for new data). The additional costs will then counteract the profitability of the portfolio.

We impose an upper bound on the portfolio size to prevent this, i.e., we require that $|\operatorname{supp}(w)| \leq k - 1$, where $\operatorname{supp}(w) := \{i \in [n] : w_i \neq 0\}$ and k is some integer such that $1 \leq k \leq n$. In problem (9.6) we model this by setting

$$\mathcal{C} = \{C \subseteq [n] : |C| = k\}.$$

Accounting for causally linked assets. The second way a portfolio can become sparse is by disallowing certain asset combinations. When one knows that two assets are causally linked, the portfolio gains negligible diversification by holding both. To factor in this notion of causally linked assets into the above model (9.6), we set

$$\mathcal{C} = \{(i,j) : i \neq j, \ | \, \mathrm{D}_{KL}(R_i, R_j)| \leq \gamma\},$$

for some $\gamma > 0$, where D_{KL} is the Kullback–Leibler divergence, see [**139**]. Other notions of mutual information or expert opinion could also be used instead of the Kullback–Leibler divergence when constructing $\mathcal{C}$.

By adding sparsity, we restrict the optimization domain and obtain a possibly weaker optimal solution. Indeed, if w is a Pareto optimal solution of the sparse problem (9.6), it need not be Pareto optimal for the dense problem (9.5).

9.2. Scalarizing the MVSK

In this section, we consider the linear scalarization (9.7) (resp, (9.12)) of problem (9.5) (resp, (9.6)). The main result is Lemma 9.2 which characterizes sufficient conditions under which the scalarized problem is convex. This result adds value, as many authors, e.g., [**95, 117, 174, 120, 147, 92**], studying the MVSK problem assumed the scalarization (or even just kurtosis alone) to be nonconvex and as such, forgo applying the powerful techniques of convex optimization.

Our plan now is as follows: Via Lemma 8.2, we can find Pareto optimal solutions of (9.5) by solving (9.7) for $\lambda > 0$ such that F_λ is convex. By doing this for various appropriate λ, we hope to recover part of the Pareto front. Later, we also apply the same process for λ that are neither strictly positive nor resulting in convex F_λ, this still yields a feasible solution of (9.7), but we have no guarantees of it being Pareto optimal for the multi-objective optimization problem (9.5).

Linear scalarization of MVSK. For any choice of $\lambda := (\lambda_1, \lambda_2, \lambda_3, \lambda_4) \geq 0$, consider the following scalarization of the multi-objective problem (9.5):

$$\begin{aligned} F_\lambda^* := \min \ \ & F_\lambda(w) := -\lambda_1 f_1(w) + \lambda_2 f_2(w) - \lambda_3 f_3(w) + \lambda_4 f_4(w) \\ s.t. \ \ & w \in \Delta^n. \end{aligned} \tag{9.7}$$

Since, for any scalar $c > 0$, we have

$$\mathrm{argmin}_{w\in\Delta^n} F_\lambda(w) = \mathrm{argmin}_{w\in\Delta^n} F_{c\lambda}(w),$$

we can hence, without loss of generality, scale λ such that it lies in the simplex Δ^4 because we are only looking for the optimizers of F_λ and not an optimal value.

Pareto optimizers of (9.5) and optimizers of (9.7). Via Lemma 8.2, a local optimizer of (9.7) for $\lambda > 0$ such that F_λ is convex is also a Pareto optimizer of (9.5). However, it is not necessarily true that all Pareto optimizers of (9.5) are also optimizers for some scalarization of the form (9.7). It is true, though, that each Pareto optimizer w^* of (9.5) satisfying (8.10) corresponds to a Karush–Kuhn–Tucker (KKT) point of some scalarization with $\lambda > 0$, as was seen in Theorem 8.3. Applying Theorem 8.3 to our setting with the simplex domain, we have the following result.

Corollary 9.1 (KKT). *Let w^* be a Pareto optimal point of (9.5) with the property that there exists no $v \in \mathbb{R}^n$ satisfying the conditions (8.10a), (8.10b), $v_j \leq 0$*

for any $j \in [n] \setminus \operatorname{supp}(w^*)$*, and* $e^T v = 0$*. Then, there exists a positive* $\lambda \in \Delta^4$*,* $\lambda > 0$*, and* $\eta \in \mathbb{R}^{n+1}$ *such that:*

- $\eta_j = 0$ *if* $j \in \operatorname{supp}(w^*)$*,*
- $\eta_j \geq 0$ *if* $j \in [n] \setminus \operatorname{supp}(w^*)$*, and*
- $\sum_{i=1}^4 \lambda_i (-1)^i \nabla f_i(w^*) + \sum_{j=1}^n \eta_j e_j + \eta_0 e = 0.$

That is to say, w^* *is also a KKT point of (9.7).*

We now shift to finding λ for which F_λ is convex over the simplex, the cube, or the whole space $\mathbb{R}^n$.

9.2.1. Convex linear scalarization of MVSK. In general, optimizing a quadratic polynomial over the simplex is already hard. Indeed, recall the Motzkin-Straus [**122**] formulation of the stability number $\alpha(G)$ of an undirected graph G as a quadratic optimization problem over the simplex, i.e.,

$$\frac{1}{\alpha(G)} = \min_{x \in \Delta^n} x^T (I + A(G)) x,$$

where $A(G)$ is the adjacency matrix of G. Problem (9.7) has a quartic objective and is expected to contain the difficulty of the quadratic case.

However, as shown in [**15**], one can optimize programs of the form

$$\min_{x \in \mathbb{R}^n} \; f(x) + g(x)$$

using the proximal gradient method, under some assumptions on the functions f and g. Assuming some convexity and closedness conditions on the functions f and g, and the existence of optimizers (see Assumption 10.1 of [**15**] for details), [**15**, Theorem 10.21] claims convergence to the global optimum at a rate of $\mathcal{O}(\frac{1}{k})$, where k is the number of iterations. In Chapter 10, we will use a proximal gradient method called FISTA which converges to the global optimum at a rate of $\mathcal{O}(\frac{1}{k^2})$. The proximal gradient method consists of iterating between a gradient step and a proximal map. In the case when the gradient and the proximal map are efficiently computable (in the sense that it requires polynomially many (in terms of input data) operations to compute), the method as a whole becomes efficient; see [**15**, Chapter 10] for details.

We now give several characterizations of $\lambda \in \Delta^4$ for which $F_\lambda(w)$ is convex. We begin by considering the gradient of F_λ at a point w as

$$\begin{aligned}\nabla F_\lambda(w) &= \nabla\Big(-\lambda_1 M^T w + \lambda_2 \mathbb{E}[\langle R, w\rangle^2] - \lambda_3 \mathbb{E}[\langle R, w\rangle^3] + \lambda_4 \mathbb{E}[\langle R, w\rangle^4]\Big) \\ &= -\lambda_1 M + 2\lambda_2 \mathbb{E}[R\langle R, w\rangle] - 3\lambda_3 \mathbb{E}[R\langle R, w\rangle^2] + 4\lambda_4 \mathbb{E}[R\langle R, w\rangle^3].\end{aligned}$$

The Hessian $H(F_\lambda)(w)$ of F_λ at w is given by

$$\begin{aligned}\nabla^2 F_\lambda(w) &= 2\lambda_2 \mathbb{E}[RR^T] - 6\lambda_3 \mathbb{E}[RR^T \langle R, w\rangle] + 12\lambda_4 \mathbb{E}[RR^T \langle R, w\rangle^2] \\ &= \mathbb{E}\Big[\big(2\lambda_2 - 6\lambda_3 \langle R, w\rangle + 12\lambda_4 \langle R, w\rangle^2\big) RR^T\Big] = \mathbb{E}[2\Phi_\lambda(R, w) RR^T],\end{aligned} \tag{9.8}$$

where we define the function

$$\Phi_\lambda(R, w) := 6\lambda_4 \langle R, w\rangle^2 - 3\lambda_3 \langle R, w\rangle + \lambda_2. \tag{9.9}$$

Define the univariate quadratic polynomial

$$\Psi_\lambda(y) := 6\lambda_4 y^2 - 3\lambda_3 y + \lambda_2,$$

so that $\Psi_\lambda(\langle R, w\rangle) = \Phi_\lambda(R, w)$ under the change of variables $y := \langle R, w\rangle$.

Lemma 9.2. *Let* $\underline{R}_\Delta := \min_{w\in\Delta^n}\langle R, w\rangle$ *and* $\overline{R}_\Delta := \max_{w\in\Delta^n}\langle R, w\rangle$. *Then we have*

$$\Phi_\lambda(R, w) \geq 0 \textit{ for all } w \in \Delta^n$$

if and only if one of the following conditions hold:

(i) $\lambda_4 = 0$ *and* $3\overline{R}_\Delta\lambda_3 \leq \lambda_2$,

(ii) $\lambda_4 > 0$ *and* $\lambda_3 \leq \sqrt{\frac{8}{3}\lambda_2\lambda_4}$,

(iii) $\lambda_4 > 0$, $\lambda_3 > \sqrt{\frac{8}{3}\lambda_2\lambda_4}$, $3\overline{R}_\Delta\lambda_3 \leq \lambda_2 + 6\overline{R}_\Delta^2\lambda_4$, *and* $4\overline{R}_\Delta\lambda_4 \leq \lambda_3$,

(iv) $\lambda_4 > 0$, $\lambda_3 > \sqrt{\frac{8}{3}\lambda_2\lambda_4}$, $3\underline{R}_\Delta\lambda_3 \leq \lambda_2 + 6\underline{R}_\Delta^2\lambda_4$, *and* $4\underline{R}_\Delta\lambda_4 \geq \lambda_3$.

Proof. If $\lambda_4 = 0$, then $\Phi_\lambda(R, w) \geq 0$ if and only if $3\langle R, w\rangle\lambda_3 \leq \lambda_2$. Requiring that this hold for all $w \in \Delta^n$ is equivalent to requiring $3\overline{R}_\Delta\lambda_3 \leq \lambda_2$. Hence, we have shown case (i).

Suppose $\lambda_4 > 0$ and consider the discriminant $\Delta_\lambda := 9\lambda_3^2 - 24\lambda_2\lambda_4$ of $\Psi_\lambda(y)$. If $\Delta_\lambda < 0$, then $\Psi_\lambda(y)$ has no real roots, and thus $\Psi_\lambda(y) > 0$ for all $y \in \mathbb{R}$. The condition $\Delta_\lambda < 0$ is equivalent to requiring $\lambda_3 < \sqrt{\frac{8}{3}\lambda_2\lambda_4}$. In the case that $\Delta_\lambda = 0$, then $\Psi_\lambda(y)$ has double root at $y = \frac{3\lambda_3}{12\lambda_4}$ and $\Psi_\lambda(y) \geq 0$ for all $y \in \mathbb{R}$. Thus, we get case (ii).

Assume $\Delta_\lambda > 0$. Then, $\Psi_\lambda(y)$ has two roots

$$y_l := \frac{3\lambda_3 - \sqrt{\Delta_\lambda}}{12\lambda_4},\ y_u := \frac{3\lambda_3 + \sqrt{\Delta_\lambda}}{12\lambda_4}.$$

Hence, there are only two cases when $\Phi_\lambda(R, w) \geq 0$ for all $w \in \Delta^n$. The first is when all values of $y = \langle R, w\rangle$ are below y_l, i.e.,

$$\overline{R}_\Delta \leq \frac{3\lambda_3 - \sqrt{\Delta_\lambda}}{12\lambda_4} = y_l \iff \sqrt{\Delta_\lambda} \leq 3\lambda_3 - 12\overline{R}_\Delta\lambda_4$$
$$\iff \Delta_\lambda = 9\lambda_3^2 - 24\lambda_2\lambda_4 \leq (3\lambda_3 - 12\overline{R}_\Delta\lambda_4)^2 \text{ and } 0 \leq 3\lambda_3 - 12\overline{R}_\Delta\lambda_4$$
$$\iff 4\lambda_4\overline{R}_\Delta \leq \lambda_3 \text{ and } 3\overline{R}_\Delta\lambda_3 \leq \lambda_2 + 6\lambda_4\overline{R}_\Delta^2.$$

Hence, we have shown case (iii).

The second case is when all values of $y = \langle R, w\rangle$ are above y_u, i.e.,

$$\underline{R}_\Delta \geq \frac{3\lambda_3 + \sqrt{\Delta_\lambda}}{12\lambda_4} = y_u \iff 12\underline{R}_\Delta\lambda_4 - 3\lambda_3 \geq \sqrt{\Delta_\lambda}$$
$$\iff 9\lambda_3^2 - 24\lambda_2\lambda_4 \leq (3\lambda_3 - 12\overline{R}_\Delta\lambda_4)^2 \text{ and } 0 \leq 12\underline{R}_\Delta\lambda_4 - 3\lambda_3$$
$$\iff 4\underline{R}_\Delta\lambda_4 \geq \lambda_3 \text{ and } \lambda_2 + 6\lambda_4\underline{R}_\Delta^2 \geq 3\underline{R}_\Delta\lambda_3.$$

With this, case (iv) is proved and the proof is concluded. □

Remark 9.3. *Note that condition (iv) in Lemma 9.2 implies* $\underline{R}_\Delta > 0$. *In numerical experiments with real-world data, we often have* $\underline{R}_\Delta < 0$, *and thus condition (iv) seldom holds.*

Corollary 9.4. *If* $\lambda \in \Delta^4$ *satisfies any of the conditions (i)-(iv) of Lemma 9.2 then* F_λ *is convex on* Δ^n. *Moreover, if* $\lambda \in \Delta^4$ *satisfies the condition (ii) of Lemma 9.2, then* F_λ *is convex on* $\mathbb{R}^n$.

PROOF. These results follow directly from the fact that the Hessian of F_λ takes only PSD matrix values when $\Phi_\lambda(R,w) \geq 0$, i.e.,

$$\Phi_\lambda(R,w) \geq 0 \implies H(F_\lambda)(w) \succeq 0 \ (w \in \mathbb{R}^n).$$

□

Generalizing the convexity results. The scalarization $F_\lambda(w)$ is convex on the standard simplex Δ^n if and only if the hyper-parameter $\lambda = (\lambda_1, \lambda_2, \lambda_3, \lambda_4) \in \Delta^4$ satisfies

$$\lambda_3 \leq \max_{\gamma \geq 0} \Big\{ \gamma : \mathbb{E}\big[(2\lambda_2 - 6\gamma\langle R,w\rangle + 12\lambda_4\langle R,w\rangle^2)RR^T\big] \succeq 0 \ \ (w \in \Delta^n) \Big\}. \tag{9.10}$$

When $\lambda_3 = 0$, F_λ is convex, so λ_3 is the limiting factor to PSDness of the Hessian of F_λ. Hence, we seek the largest λ_3 for which $H(F_\lambda)(w) \succeq 0$ for all $w \in \Delta^n$. The parameter λ_1 plays no role in the convexity of F_λ. The Hessian is linear in λ but quadratic in w. The expression in problem (9.10) is not simply a *linear matrix inequality*, and to the best of our knowledge, it cannot be solved efficiently [**17**].

Thus far, we have considered convexity over the simplex domain. Analogous results hold for the cube. To generalize Lemma 9.2 to the cube, simply modify the bounds $\overline{R}_\Delta$ and $\underline{R}_\Delta$ by defining

$$\overline{R}_\Box := \max_{w \in [-1,1]^n} \langle R,w\rangle, \ \underline{R}_\Box := \min_{w \in [-1,1]^n} \langle R,w\rangle.$$

The results of Lemma 9.2 and Corollary 9.4 can be extended to strict convexity by making a mild assumption on the random variable R.

COROLLARY 9.5. *Consider the Hessian given in (9.8) for some $\lambda \in \Delta^4$. Assume that $\mathbb{E}[RR^T] \succ 0$ and that, for all $w \in \Delta^n$, $\Phi_\lambda(R,w) > 0$ a.e.*[1]*. Then, F_λ is strictly convex on Δ^n.*

PROOF. Since $\Phi_\lambda(R,w) > 0$ a.e. for all $w \in \Delta^n$, we have that $H(F_\lambda) \succeq 0$ on Δ^n. Assume by way of contradiction that $H(F_\lambda)$ is not positive definite, then there exists a nonzero $v \in \mathbb{R}^n \setminus \{0\}$ such that $v^T H(F_\lambda)v = v^T\mathbb{E}[\Phi_\lambda(R,w)RR^T]v = 0$. By linearity of the expectation this implies that $\mathbb{E}[\Phi_\lambda(R,w)v^TRR^Tv] = 0$. Since each argument is a.e. nonnegative we have that $\Phi_\lambda(R,w)v^TRR^Tv = 0$ a.e., and thus $v^TRR^Tv = 0$ a.e. by virtue of $\Phi_\lambda(R,w) > 0$ a.e.. Taking the expectation we get $0 = \mathbb{E}[v^TRR^Tv] = v^T\mathbb{E}[RR^T]v$ contradicting our assumption that $\mathbb{E}[RR^T] \succ 0$. □

Regions of hyper-parameters λ for which F_λ is convex. We define the following nested sets of hyper-parameters λ

$$\Lambda_+ \subseteq \Lambda_\Delta \subseteq \widehat{\Delta} \ \text{ and } \ \Lambda_+ \subseteq \Lambda_\Box \subseteq \widehat{\Delta},$$

[1] The abbreviation *a.e.* stands for *almost everywhere* and is used to indicate that the accompanying statement may fail, but only on a set of measure zero.

where

$$\begin{aligned}
&\widehat{\Delta} := \{(\lambda_2,\lambda_3,\lambda_4) \geq 0 : \lambda_2+\lambda_3+\lambda_4 \leq 1\} \subseteq \mathbb{R}^3,\\
&\Lambda_+ := \{(\lambda_2,\lambda_3,\lambda_4) \in \widehat{\Delta} : \lambda_2\lambda_4 \geq (3/8)\lambda_3^2\},\\
&\Lambda_\Delta := \{(\lambda_2,\lambda_3,\lambda_4) \in \widehat{\Delta} : \lambda \text{ satisfies any condition of Lemma 9.2 for } \overline{R}_\Delta \text{ and } \underline{R}_\Delta\},\\
&\Lambda_\square := \{(\lambda_2,\lambda_3,\lambda_4) \in \widehat{\Delta} : \lambda \text{ satisfies any condition of Lemma 9.2 for } \overline{R}_\square \text{ and } \underline{R}_\square\}.
\end{aligned} \tag{9.11}$$

Via Lemma 9.2, it now follows that if $\lambda \in \Lambda_+$, then F_λ is convex over $\mathbb{R}^n$. Similarly, if $\lambda \in \Lambda_\Delta$ (resp., $\Lambda_\square$), then F_λ is convex over the simplex Δ^n (resp., the cube $[-1,1]^n$). The benefit of eliminating a variable (λ_1 in this case) is that the hyper-parameter sets $\Lambda_+,\ \Lambda_\Delta,\ \Lambda_\square$, and $\widehat{\Delta}$ can now be plotted, see Figure 1. Keep in mind that the set Λ_Δ is a conservative estimate for the set of all $\lambda \in \widehat{\Delta}$ for which F_λ is convex over the simplex, i.e.,

$$\Lambda_\Delta \subseteq \Big\{(\lambda_2,\lambda_3,\lambda_4) \in \widehat{\Delta} : F_\lambda \text{ is convex on } \Delta^n\Big\}.$$

Hence, the region Λ_Δ shown in Figure 1 should be thought of as pessimistic, and similarly for $\Lambda_\square$. Furthermore, even if F_λ is non-convex, one can still optimize (9.7) and hope that the local optimum attained is sufficiently good.

The function of these sets is as follows. By optimizing F_λ for different $\lambda \in \Delta^4$, we recover local optimizers w_λ. If $\lambda \in \Lambda_\Delta$, then Corollary 9.4 guarantees that the optimizer w_λ is globally optimal for problem (9.7). If additionally, we know that $\lambda > 0$, then by Lemma 8.2, we know that w_λ is a Pareto optimal point of the problem (9.5). Later in Chapter 10, we will visualize the quality of solutions w_λ by plotting objective values $f_i(w_\lambda)$ against $\lambda \in \widehat{\Delta}$, for $i \in [4]$. Hence, the sets Λ_Δ, $\Lambda_\square$ are useful in showing where we certainly have Pareto optimality.

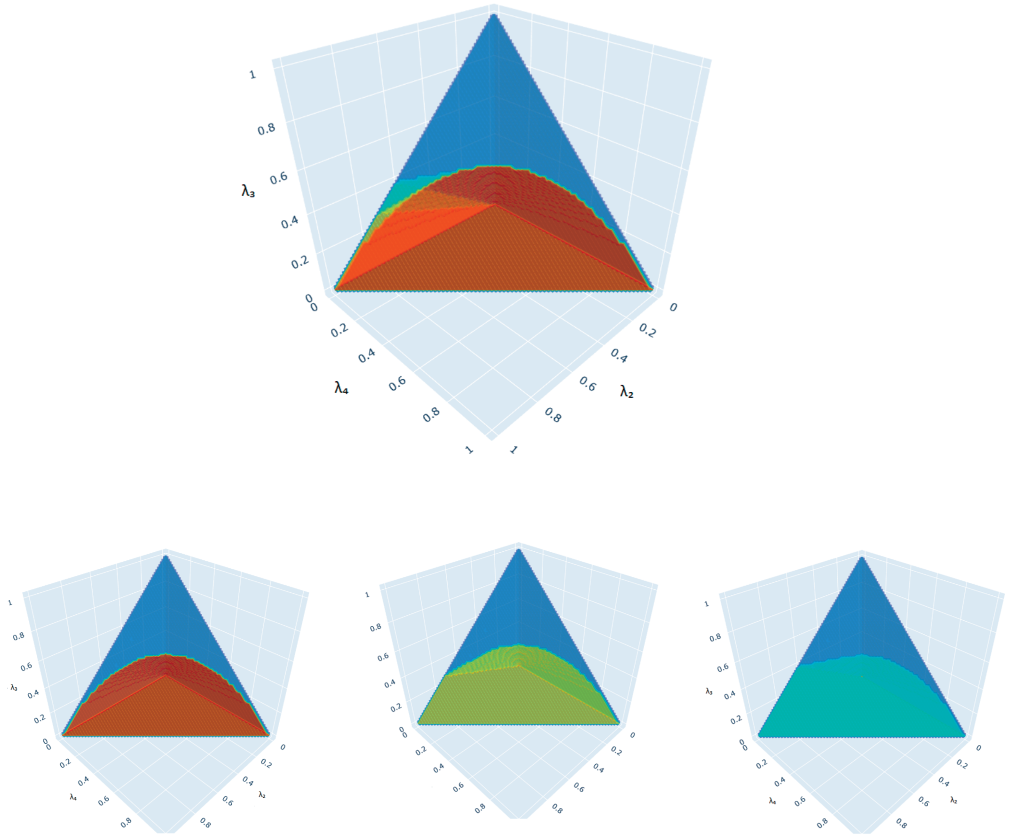
9.2.2. Scalarized sparse MVSK. Analogously to the above discussion, one can associate a linear scalarization to the sparse multi-objective optimization problem in (9.6) in the following way

$$\begin{aligned}
F^*_{\lambda,\mathcal{C}} := \min \quad & F_\lambda(w) := -\lambda_1 f_1(w) + \lambda_2 f_2(w) - \lambda_3 f_3(w) + \lambda_4 f_4(w)\\
s.t. \quad & w \in \Delta^n\\
& \prod_{i\in C} w_i = 0 \text{ for } C \in \mathcal{C}.
\end{aligned} \tag{9.12}$$

We now show that optimization problems of the form (9.12) can be decomposed into a collection of several independent sub-problems of the form (9.7). The motivation for doing this is that the sub-problems could possibly be solved independently using parallelization or other forms of distributed computing.

For any set $U \subseteq [n]$ and vector $x \in \mathbb{R}^U$, denote by $x(0,U) \in \mathbb{R}^n$ the lifting of x into $\mathbb{R}^n$, defined entrywise by

$$x(0,U)_i := \begin{cases} x_i & i \in U \\ 0 & i \notin U \end{cases} \quad (i \in [n]).$$



((A)) Λ_+ ((B)) $\Lambda_\square$ ((C)) Λ_Δ

FIGURE 1. This plot shows the transparent three-dimensional hyper-parameter set $\widehat{\Delta}$ in blue as viewed from the facet: $\{(\lambda_2, \lambda_3, \lambda_4) \geq 0 : \lambda_2 + \lambda_3 + \lambda_4 = 1\} \subset \widehat{\Delta}$. The different regions are distinguished by color. In particular, Λ_+ is shown in red, $\Lambda_\square$ is shown in green, and Λ_Δ is shown in light-blue. The domains Λ_Δ and $\Lambda_\square$ shown here were computed using $\overline{R}_\Delta = 0.52$, $\overline{R}_\square = 0.87$, and $\underline{R}_\Delta$, $\underline{R}_\square < 0$. For this instance $\Lambda_\square \subseteq \Lambda_\Delta$. The approximate relative volumes for the sub-domains are as follows: $\frac{\text{vol}(\Lambda_+)}{\text{vol}(\widehat{\Delta})} \approx 0.59$, $\frac{\text{vol}(\Lambda_\square)}{\text{vol}(\widehat{\Delta})} \approx 0.61$, and $\frac{\text{vol}(\Lambda_\Delta)}{\text{vol}(\widehat{\Delta})} \approx 0.63$.

Similarly, for $x \in \mathbb{R}^n$, let $x_{|_U} = (x_i)_{i \in U} \in \mathbb{R}^U$ denote the restriction of x to $\mathbb{R}^U$. For a function $g : \mathbb{R}^n \to \mathbb{R}$ define the restricted function $g_{|_U} : \mathbb{R}^U \to \mathbb{R};\ x \mapsto g(x(0, U))$.

PROPOSITION 9.6. *Let $U_1, ..., U_p \subseteq [n]$ be all the maximal subsets of $[n]$ not containing any set $C \in \mathcal{C}$. Then, we have*

$$F^*_{\lambda,\mathcal{C}} = \widetilde{F}^*_{\lambda,\mathcal{C}},$$

where

$$\widetilde{F}^*_{\lambda,\mathcal{C}} := \min_{\ell\in[p]} \min_{\widetilde{w}\in\Delta^{U_\ell}} F_{\lambda|_{U_\ell}}(\widetilde{w}). \tag{9.13}$$

Proof. ($F^*_{\lambda,\mathcal{C}} \geq \widetilde{F}^*_{\lambda,\mathcal{C}}$) Any optimal solution w of (9.12) must have its support contained in some U_ℓ. Hence, there is a $\widetilde{w} \in \Delta^{U_\ell}$ such that $\widetilde{w}(0, U_\ell) = w$ and

$$\widetilde{F}^*_{\lambda,\mathcal{C}} \leq F_{\lambda|_{U_\ell}}(\widetilde{w}) = F_\lambda(w) = F^*_{\lambda,\mathcal{C}}.$$

($F^*_{\lambda,\mathcal{C}} \leq \widetilde{F}^*_{\lambda,\mathcal{C}}$) Let $\widetilde{w} \in \Delta^{U_\ell}$ for some $\ell \in [p]$ be an optimizer of (9.13), then

$$\widetilde{F}^*_{\lambda,\mathcal{C}} = F_{\lambda|_{U_\ell}}(\widetilde{w}) = F_\lambda(\widetilde{w}(0, U_\ell)) \geq F^*_{\lambda,\mathcal{C}}.$$

□

Proposition 9.6 bares many similarities with Proposition 2.2. In Section 2.2.1, ideal sparsity was introduced in the context of the generalized moment problem (GMP) when one restricts the support of the involved measure. A GMP with a single measure having restricted support can be shown to be equivalent (see Proposition 2.2) to another GMP involving several measures, each having smaller support than the measure in the original GMP. In Proposition 9.6, we show that a polynomial optimization problem with restricted support is equivalent to optimizing over a set of smaller polynomial optimization problems without support constraints.

In both settings, the critical insight is that the restricted support constraint decomposes into a collection of smaller objects without support constraints.

Convexity of the scalarized sparse MVSK. Similar to the dense case in Section 9.2.1, the objective function in (9.12) is convex if λ satisfies any of the conditions (i)-(iv) of Lemma 9.2. The result of Lemma 9.2 transfers to the sparse case because $\Phi_\lambda(R, w) \geq 0$ on Δ^n implies that $\Phi_\lambda(R, w) \geq 0$ on

$$\Delta^n_{\mathcal{C}} := \Big\{ w \in \Delta^n : \prod_{i\in C} w_i = 0 \text{ for } C \in \mathcal{C} \Big\} \subseteq \Delta^n. \tag{9.14}$$

Hence, Lemma 9.2 and its consequences continue to hold in the sparse setting.

Note that the domain $\Delta^n_{\mathcal{C}}$ is not convex, so the problem (9.12) is not convex. However, if $U_1, ..., U_p \subseteq [n]$ denote all the maximal subsets of $[n]$ not containing any set $C \in \mathcal{C}$, then for any $\ell \in [p]$ the sub-problem

$$\min_{\widetilde{w}\in\Delta^{U_\ell}} F_{\lambda|_{U_\ell}}(\widetilde{w}),$$

does have a convex domain, namely the simplex Δ^{U_ℓ}. Moreover, on this sub-problem, Lemma 9.2 can be adapted by using the following bounds

$$\underline{R}_{\Delta^{U_\ell}} := \min_{\widetilde{w}\in\Delta^{U_\ell}} \langle R, \widetilde{w}(0, U_\ell)\rangle, \ \overline{R}_{\Delta^{U_\ell}} := \max_{\widetilde{w}\in\Delta^{U_\ell}} \langle R, w(0, U_\ell)\rangle.$$

Observe that $\underline{R}_\Delta \leq \underline{R}_{\Delta^{U_\ell}} \leq \overline{R}_{\Delta^{U_\ell}} \leq \overline{R}_\Delta$ for all $\ell \in [p]$. Furthermore, any λ that satisfies at least one of the conditions (i)-(iv) of Lemma 9.2 using the bounds $\underline{R}_\Delta$ and $\overline{R}_\Delta$ will necessarily again satisfy one of the conditions using instead now the bounds $\overline{R}_{\Delta^{U_\ell}}$ and $\underline{R}_{\Delta^{U_\ell}}$, for any $\ell \in [p]$. Intuitively one can think of using these new bounds $\underline{R}_{\Delta^{U_\ell}}$ and $\overline{R}_{\Delta^{U_\ell}}$ as relaxing the condition $\Phi_\lambda(R, w) \geq 0$ for all $w \in \Delta^n$ to the weaker condition $\Phi_\lambda(R, w) \geq 0$ for all $w \in \Delta^n$ with $\mathrm{supp}(w) \subseteq U_\ell$. This mirrors the fact that there are potentially more hyper-parameters $\lambda \in \Delta^4$ for which $F_{\lambda|_{U_\ell}}$ is convex over Δ^{U_ℓ} for each $\ell \in [p]$ than there are $\lambda \in \Delta^4$ for which F_λ is convex over Δ^n.

The sparse problem (9.12) could have many sub-problems to solve, but each sub-problem is smaller than the original problem and can be solved independently of the other sub-problems. If we set $\mathcal{C}$ to be the collection of all sets of size $k+1$, then there are $\binom{n}{k}$ sub-problems to solve, each involving k variables.

CHAPTER 10

Numerical Experiments

In this section, we apply the theory from Chapter 9 to empirical data from the Standard and Poor's 500 (S&P500) stock market index [**60**].

We discuss the optimization algorithm *FISTA*, by Beck and Teboulle [**16**], that we use to solve the scalarized problem (9.7), and we motivate its use by listing some of FISTA's desirable properties (see Section 10.1). We explain our methodology for acquiring a grid approximation for the Pareto set of problem (9.5) (see Section 10.2). Having obtained a set of optimizers of the scalarized problem (9.7) for different choices of hyper-parameters, we compare and visualize the objective values of the multi-objective problem (9.5) at said optimizers (see Section 10.3). We observe that some optimizers give a better overall balance among the four objectives. This procedure is performed for the simplex and cube settings as well as their sparse analogs.

10.1. Optimization algorithm for the scalarized problem

We will be using the *fast iterative shrinkage-thresholding algorithm* (FISTA), also known as *fast proximal gradient method*, which is a well-studied first-order iterative optimization algorithm first devised and analyzed by Beck and Teboulle [**16**]. Broadly speaking, FISTA repeats the following two steps a prespecified number of times, starting from an initial point: A gradient-descent step gives a new point; the new point is projected back into the feasible region. Notice that one can efficiently project onto the simplex [**164**] (resp., the cube).

FISTA provides sparse solutions on the simplex domain. Like many gradient descent algorithms, FISTA uses a projection operator to maintain the simplex (resp., cube) constraints. The operator that projects to the simplex is defined by

$$\mathrm{Proj}_{\Delta^n} : \mathbb{R}^n \to \Delta^n;\ x \mapsto \mathrm{argmin}_{y\in\Delta^n} \|x-y\|.$$

If the nearest unconstrained optimizer lies outside the simplex, then most gradient steps will leave the domain. Projecting back to the simplex results in a sparse vector, i.e., without full support. The sparsity seems to be due to the fact that projections are often on a face of the simplex. Hence, most optimizers obtained from FISTA will be sparse. We provide a histogram of the supports of optimizers from the set $W_{\Delta}^{[40]}$ (defined in Section 10.1.2) for our particular problem in Figure 1. This sparsity does not occur in the case of the cube domain, i.e., the supports of $W_{\square}^{[40]}$ (defined in Section 10.1.2) are all full. One possible reason for this is that the unconstrained optimizer lies within the cube, and, as such, the projection operator does nothing. Note that the cube is full-dimensional in contradistinction to the simplex, which lies

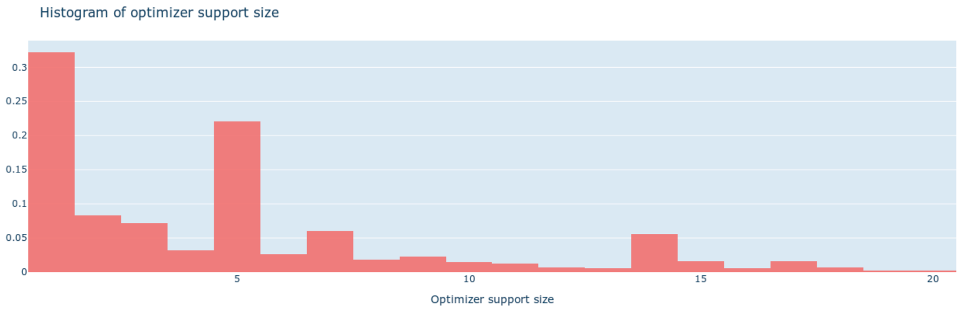


FIGURE 1. Normalized histogram of the support sizes $|\operatorname{supp}(w_\lambda)|$ of optimizers $w_\lambda \in W_\Delta^{[40]}$.

in the hyperplane $\{w \in \mathbb{R}^n : \sum_{i\in[n]} w_i = 1\}$.

FISTA benefits from a warm start. FISTA is an iterative algorithm that starts from an initial guess x^0 and then incrementally improves a proposed optimizer until a certain number of iterations have been completed. In the convex problem, the algorithm will converge to the global optimum regardless of where one starts, but a closer start x^0 to an optimizer x^* does imply faster convergence (recall (10.6)). Furthermore, if the problem is not convex, and one starts sufficiently close to the global optimizer, then one can be sure that FISTA will converge to the true optimum. An initial guess x^0 that is close to the global optimizer x^* is called a *warm start*. We now propose to use the optimizer from an already solved problem (9.7), with a fixed λ, as a warm start for solving (9.7) with a different hyper-parameter $\widehat{\lambda}$. In other words, fix $\lambda \in \Delta^n$. If

$$w_\lambda \in \operatorname{argmin}_{w\in\Delta^n} F_\lambda(w),$$

then take $x^0 = w_\lambda$ as a warm start for FISTA when solving

$$\min_{w\in\Delta^n} F_{\widehat{\lambda}}(w). \tag{10.1}$$

Our intuition here is as follows: if λ is close to $\widehat{\lambda}$, then we expect w_λ should be close to an optimizer $w_{\widehat{\lambda}}$ of (10.1). We provide no proof of the validity of this intuition. Note that using this warm start heuristic could have the unintended effect of preventing us from reaching the global optimizer when F_λ is nonconvex over the simplex. Our warm start heuristic can also be applied to computing sparse optimizers via FISTA. We will elaborate more on this next.

10.1.1. Optimization algorithm for the sparse scalarized problem. We saw above that the optimizers of the problem (9.7) are sometimes sparse for the simplex setting, but not always. So, we propose a simple scheme for finding optimizers with support not exceeding some fixed integer $k \in \mathbb{N}$. We do this starting from a set of possibly dense solutions W. Let $w_\lambda \in W$ be the optimizer of the (dense) problem (9.7). If $|\operatorname{supp}(w_\lambda)| \leq k$, then we are done. So, suppose that $|\operatorname{supp}(w_\lambda)| > k$. Keeping with the notation of Section 9.2.2 we let $\mathcal{C} = \{C \subseteq [n] : |C| > k\}$ and define the set

$$\mathcal{U}^k := \{U \subseteq [n] : |U| = k\}$$

of all maximal subsets of $[n]$ that do not contain any set $C \in \mathcal{C}$. The sparse problem (9.12) can be rewritten as follows:

$$\min_{\substack{w\in\Delta^n \\ |\operatorname{supp}(w)|\leq k}} F_\lambda(w) = \min_{\substack{U\in\mathcal{U}^k \\ \widetilde{w}\in\Delta^U}} F_{\lambda|_U}(\widetilde{w}). \tag{10.2}$$

For a fixed $U \in \mathcal{U}^k$ we can solve the sub-problem

$$\min_{\widetilde{w}\in\Delta^U} F_{\lambda|_U}(\widetilde{w}), \tag{10.3}$$

using FISTA with $x^0 = \operatorname{Proj}_{\Delta^U}(w_\lambda)$ as a warm start, where w_λ is assumed to be an optimizer from the dense problem (9.7) with the same hyper-parameter λ.

Removing sub-problems based on proximity to the dense optimizer. In order to not consider all $\binom{n}{k}$-many sets U of $\mathcal{U}^k$, we propose the following two heuristics to remove sets U for which the resulting sub-problem (10.3) could have a poor optimum value. The two heuristics we introduce can be used independently of each other. However, we will use them together in the sequence we introduce them.

The first heuristic consists of discarding all sets $U \in \mathcal{U}^k$ that do not satisfy $U \subseteq \operatorname{supp}(w_\lambda)$. Doing so yields only $\binom{|\operatorname{supp}(w_\lambda)|}{k} \leq \binom{n}{k}$ sets to optimize over in (10.2).

The second heuristic is to look at the elements $w_{\lambda,U}$ of the set

$$W_{\lambda,k} := \Big\{w_{\lambda,U} := \big(\operatorname{Proj}_{\Delta^U}(w_\lambda)\big)(0,U) : U \in \mathcal{U}^k,\ U \subseteq \operatorname{supp}(w_\lambda)\Big\} \subseteq \mathbb{R}^n,$$

obtained by projecting w_λ onto $\Delta^U \subseteq \mathbb{R}^U$ and then lifting the projection to a vector in $\mathbb{R}^n$ by padding entries not supported by U with zeros, for all appropriate sets U. To use the second heuristic independently of the first one, simply drop the constraint $U \subseteq \operatorname{supp}(w_\lambda)$ in the definition of $W_{\lambda,k}$. We can then choose to solve the problem (10.3) only over sets U for which $w_{\lambda,U}$ is close to w_λ in the Euclidean norm. For our implementation we take the sets U corresponding to the n closest $w_{\lambda,U}$ to w_λ. Though we provide no guarantee that choosing a $U \subseteq [n]$ such that $w_{\lambda,U}$ is closest to w_λ would result in an optimum value of problem (10.3) being any better than another choice of U, we still find that this a helpful heuristic for removing poor choices of U.

10.1.2. The set of obtained optimizers. Whether F_λ is convex or not, we can apply FISTA to obtain at least a feasible point w_λ for the problem (9.7). Construct the following sets of points obtained by applying FISTA to various scalarizations:

$$\begin{aligned} W_\Delta &:= \{w_\lambda \in \operatorname{argmin}_{\text{FISTA } w\in\Delta^n} F_\lambda(w) : \lambda \in \Delta^4\}, \\ W_\Box &:= \{w_\lambda \in \operatorname{argmin}_{\text{FISTA } w\in[-1,1]^n} F_\lambda(w) : \lambda \in \Delta^4\}. \end{aligned} \tag{10.4}$$

Here, $\operatorname{argmin}_{\text{FISTA}}$ denotes the points obtained via the algorithm FISTA, not to be confused with the true (unknown) global minimizers. In Section 10.3.1, we will visualize the values of the objectives $f_1(w)$, $f_2(w)$, $f_3(w)$, and $f_4(w)$ for $w \in W_\Delta$ (resp., $w \in W_\Box$) using colors. Similar to (10.4), we construct the sets of sparse points recovered via FISTA

$$\begin{aligned} W_{\Delta,k} &:= \{w_\lambda \in \operatorname{argmin}_{\text{FISTA } w\in\Delta^n,\ |\operatorname{supp}(w)|\leq k} F_\lambda(w) : \lambda \in \Delta^4\}, \\ W_{\Box,k} &:= \{w_\lambda \in \operatorname{argmin}_{\text{FISTA } w\in[-1,1]^n,\ |\operatorname{supp}(w)|\leq k} F_\lambda(w) : \lambda \in \Delta^4\}, \end{aligned}$$

obtained by following the procedure described in Section 10.1.1. As mentioned before, projecting onto the simplex often produces a sparse vector. Hence, it makes sense to use W_Δ as a starting point for computing $W_{\Delta,k}$, as many of the vectors of W_Δ may already be sparse enough. Regardless of whether the elements of W_Δ (resp., $W_\square$) are sparse, we can use the ideas of Section 10.1.1 to prune computations and generate warm starts for the problems associated with $W_{\Delta,k}$ (resp., $W_{\square,k}$).

Defining objective functions from empirical data. For the sake of generality, we have introduced the MVSK problem in Chapter 9 using a vector-valued random variable R (resp., $\widetilde{R}$) taking values in $\mathbb{R}^n$ to describe the data-dependency. Practically, the data will arise from a table of results, taking the form of an $n \times m$ matrix $\widetilde{T} \in \mathbb{R}^{n\times m}$, where m is the number of outcomes observed over time. We introduce a new notation for the empirical data $\widetilde{T}$ and the subsequently derived quantities. The entry $\widetilde{T}_{i,j}$ (resp., $T_{i,j}$) is interpreted as the *empirical (resp., centralized) relative returns* of asset i at time j. In this context, the expectation is taken over the outcomes. The mean becomes the *empirical mean*, i.e.,

$$M = \Big(\frac{1}{m}\sum_{p\in[m]} \widetilde{T}_{i,p}\Big)_{i\in[n]} \in \mathbb{R}^n.$$

Hence, the *empirical centralized relative returns* is defined for each $i \in [n]$ and $p \in [m]$ by $T_{i,p} := \widetilde{T}_{i,p} - M_i$. Similar to the mean, the formulation of the other empirical moments is as follows:

$$\begin{aligned} V &= \Big(\frac{1}{m-1}\sum_{p,q\in[m]} T_{i,p}T_{j,q}\Big)_{i,j\in[n]}, \\ S &= \Big(\frac{1}{m}\sum_{p,q,r\in[m]} T_{i,p}T_{j,q}T_{k,r}\Big)_{(i,j)\in([n]\times[n]),\ k\in[n]}, \\ K &= \Big(\frac{1}{m}\sum_{p,q,r,s\in[m]} T_{i,p}T_{j,q}T_{k,r}T_{\ell,s}\Big)_{(i,j),(k,\ell)\in([n]\times[n])}. \end{aligned} \tag{10.5}$$

Observe that we use the unbiased estimator of the variance in (10.5); for a general reference on statistical estimators, we refer to [**158**]. The objective functions f_1, f_2, f_3, and f_4 defined in (9.2) and (9.3) can henceforth be redefined in terms of the above M, V, S, or K.

Using $T \in \mathbb{R}^{n\times m}$, the bounds $\overline{R}_\Delta$ and $\underline{R}_\Delta$ in Lemma 9.2 now become

$$\overline{R}_\Delta = \max_{i\in[n],p\in[m]} T_{i,p},\ \underline{R}_\Delta = \min_{i\in[n],p\in[m]} T_{i,p}.$$

For the cube the bounds are $\overline{R}_\square = \max_{p\in[m]}\sum_{i\in[n]} |T_{i,p}|$ and $\underline{R}_\square = -\overline{R}_\square$. The sparse analogs $\underline{R}_{\Delta^U}$, $\overline{R}_{\Delta^U}$, $\underline{R}_{\square^U}$, and $\overline{R}_{\square^U}$ are defined, mutatis mutandis, in the same manner. The bounds we gave in Figure 1 are also used for all computations we show. We only compute and use the dense bounds ($\overline{R}_\Delta$, $\underline{R}_\Delta$, $\underline{R}_\square$, and $\overline{R}_\square$), even for the sparse settings.

In the next section we sub-sample the sets W_Δ, $W_\square$, $W_{\Delta,5}$, and $W_{\square,5}$, described in the preceding section. Our empirical data $\widetilde{T}$ will be a selection of stocks from the well-known *Standard and Poor's 500* (S&P500) stock market index, see [**60**]. We will consider $n = 20$ stocks, each measured in increments of a day over a timespan of $m = 500$ days starting in January 1990. We have chosen this dataset because it is well-known and publicly available. However, everything we describe in this chapter could also be applied to any other time series data of asset prices. For the reader's convenience, we list some papers [**120, 6**] that investigate the MVSK model on markets different from the S&P500. Using $\widetilde{T}$ we can generate M, V, S, and K as described above. From here, we can define the problem (9.7) and its sparse analog (9.12). Solving these problems, using the procedure described in Section 10.1, we obtain elements from the sets W_Δ, $W_\Box$, $W_{\Delta,5}$, and $W_{\Box,5}$.

Convergence bounds for the simplex setting. Consider the scalar optimization problem (9.7) and assume that F_λ is convex. Under some mild smoothness assumptions (see Assumption 10.31 of [**15**] for details), which hold in our setting, the following performance guarantee (see [**15**, Theorem 10.34]) holds for the k^{th} iteration of FISTA when applied to (9.7):

$$F_\lambda(x^k) - F_\lambda(x^*) \leq \frac{2L_F\|x^0 - x^*\|^2}{(k+1)^2}. \tag{10.6}$$

Here, $L_F > 0$ is the Lipschitz constant of F_λ, x^0 is the initial point, x^* is an optimizer, and x^k is the point obtained from FISTA at the k^{th} iteration.

Using a result from [**102**], one can bound the Lipschitz constant L_f of an n-variate degree d polynomial $f(x)$ over a convex body $\mathcal{K}$ as follows:

$$L_f \leq \frac{2d^2}{\text{width}(\mathcal{K})} \sup_{x\in\mathcal{K}} |f(x)|, \tag{10.7}$$

where width($\mathcal{K}$) is the minimum distance between two distinct parallel supporting hyperplanes of $\mathcal{K}$. In the scalarized MVSK setting with simplex domain, we have

$$\text{width}\Big(\Big\{x \in \mathbb{R}^{n-1} : x_i \geq 0 \ (i \in [n-1]), \sum_{i\in[n-1]} x_i \leq 1\Big\}\Big) = \frac{1}{\sqrt{n-1}}$$

and $\|x^0 - x^*\| \leq \sqrt{2}$. We can upper-bound the objective function by

$$\sup_{w\in\Delta^n} |F_\lambda(w)| \leq \max\Big\{\max_i |M_i|, \ \max_{i,j} |V_{ij}|, \ \max_{i,j,k} |S_{ijk}|, \ \max_{i,j,k,l} |K_{ijkl}|\Big\}$$

because $\lambda \in \Delta^4$, and

$$\max_{w\in\Delta^n} | \sum_{i,j,k,l\in[n]} w_i w_j w_k w_l K_{ijkl}| \leq (\sum_{i\in[n]} w_i)^4 \max_{i,j,k,l} |K_{ijkl}| = \max_{i,j,k,l} |K_{ijkl}|$$

and similarly for the other terms.

For the above mentioned S&P500 stock market data we get that $F_\lambda(w)$ is upper bounded by $\sup_{w\in\Delta^n} |F_\lambda(w)| \leq 0.003$ for all $\lambda \in \Delta^4$. Hence, using the bound in (10.7), we get $L_F \leq 2 \cdot 4^2\sqrt{19} \cdot 0.003 = 0.49$. For all our applications of FISTA, we used $k = 2000$ iterations. Thus, using (10.6), we get that

$$F_\lambda(x^{2000}) - F_\lambda(x^*) \leq \frac{2L_F\|x^0 - x^*\|^2}{(k+1)^2} \leq \frac{2 \cdot 0.49 \cdot 2}{2001^2} = 4.84 \times 10^{-7}.$$

In order to take the scale of the values of F_λ into account, we note that for all of the points of $W_\Delta^{[40]}$ we compute using FISTA, the values of F_λ do not go below $-3.4 \times 10^{-3} \leq \min_{w \in W_\Delta^{[40]}, \lambda \in \Delta^4_{[40]}} F_\lambda(w)$, where $W_\Delta^{[40]}$ and $\Delta^4_{[40]}$ are described in the next section. We found that $k = 2000$ iterations adequately balanced accuracy and computation time.

Convergence bounds for the cube setting. We can use the same process described above to get performance guarantees for the cube setting. The difference now being that we have to use $\mathrm{width}([-1,1]^n) = 2$, $\|x^0 - x^*\|^2 \leq 4n$, and the objective function is now upper-bounded by

$$\sup_{w \in [-1,1]^n} |F_\lambda(w)| \leq \max\Big\{ \sum_i |M_i|, \sum_{i,j} |V_{ij}|, \sum_{i,j,k} |S_{ijk}|, \sum_{i,j,k,l} |K_{ijkl}| \Big\}.$$

In the cube setting, for the above mentioned S&P500 stock market data, we get a bound $F_\lambda(x^{2000}) - F_\lambda(x^*) \leq 2.09 \times 10^{-5}$.

10.2. A grid approximation of the Pareto set

Recall that the ultimate goal is to obtain Pareto optimizers of the MVSK problem (9.5). Via Lemma 8.2, solving the scalarization (9.7) for $\lambda > 0$ gives a Pareto optimizer of (9.5). However, we can still recover a point by solving the scalarization (9.7) for $\lambda \geq 0$; we simply have no optimality guarantees with respect to (9.5). Because Δ^4 contains uncountably many elements, we resort to sub-sampling Δ^4 with a uniform mesh. Fix $s \in \mathbb{N}$, and consider the following sets:

$$\begin{aligned}
\Delta^4_{[s]} &:= \{\lambda : \lambda \in \{0, \frac{1}{s}, \frac{2}{s}, ..., 1\}^4 \cap \Delta^4\},\\
\widehat{\Delta}_{[s]} &:= \{(\lambda_2, \lambda_3, \lambda_4) : (1 - (\lambda_2 + \lambda_3 + \lambda_4), \lambda_2, \lambda_3, \lambda_4) \in \Delta^4_{[s]}\} \subseteq \mathbb{R}^3,
\end{aligned}$$

that are clearly in bijection. For our computations we take $s = 40$, resulting in $|\widehat{\Delta}_{[40]}| = 11480$ choices of hyper-parameter λ to consider. For each $\lambda \in \Delta^4_{[40]}$, we solve the associated scalarization (9.7) using FISTA to obtain a set of local optimizers w_λ, denoted by

$$W_\Delta^{[40]} \subseteq \{w_\lambda \in \mathrm{argmin}_{\mathrm{FISTA}\ w \in \Delta^n} F_\lambda(w) : \lambda \in \Delta^4_{[40]}\} \subseteq W_\Delta.$$

Observe that the set $W_\Delta^{[40]}$ is not necessarily contained in the Pareto front, but the following subset is:

$$\{w_\lambda \in W_\Delta^{[40]} : \lambda \in \Lambda_\Delta,\ \lambda > 0\}.$$

Here, we use the claims from Corollary 9.4 and Lemma 8.2 that if $\lambda \in \Lambda_\Delta$ and $\lambda > 0$, then w_λ is a Pareto optimizer of problem (9.5). The reason we consider the bigger set $W_\Delta^{[40]}$ is that we get a more complete picture, see the figures of Section 10.3. Although some points of $W_\Delta^{[40]}$ are not guaranteed to be a Pareto optimizer of (9.5), they are nonetheless quite comparable to the points that are Pareto optimal for (9.5). We illustrate this claim with visualization.

Comparing objective values. In order to compare points $w \in W_\Delta^{[40]}$, we rank them in terms of their values for the objective functions f_1, f_2, f_3, and f_4 in (9.5). For each $w \in W_\Delta^{[40]}$ we compute the values $f_1(w)$, $f_2(w)$, $f_3(w)$, and $f_4(w)$. For the sake of clarity, since there is a scale difference between the different functions, we linearly rescale the values to be in the interval $[0,1]$. Formally, for each $i \in [4]$ define

$$F_{i,\Delta}^{[40]} := \{f_i^{[40]}(w) : w \in W_\Delta^{[40]}\},$$

to be the set of linearly scaled values $f_i(w)$ for $w \in W_\Delta^{[40]}$, where

$$f_i^{[40]}(w) := \begin{cases} \frac{f_i(w) - f_i^{\min,[40]}}{f_i^{\max,[40]} - f_i^{\min,[40]}} & i = 1 \text{ or } 3 \\ 1 - \frac{f_i(w) - f_i^{\min,[40]}}{f_i^{\max,[40]} - f_i^{\min,[40]}} & i = 2 \text{ or } 4 \end{cases}, \tag{10.8}$$

with

$$f_i^{\max,[40]} := \max_{w \in W_\Delta^{[40]}} f_i(w), \;\; f_i^{\min,[40]} := \min_{w \in W_\Delta^{[40]}} f_i(w).$$

Hence, for any $i \in [4]$, the set $F_{i,\Delta}^{[40]}$ is contained in the unit interval $[0,1]$, with “less desirable” values close to zero and “more desirable” values close to one. Note that the scaling $f_i^{[40]}$ considers the fact that we want to maximize f_1 and f_3, and to minimize f_2 and f_4. Hence, the set $F_{i,\Delta}^{[40]}$ gives us a way to compare the performance of each portfolio $w \in W_\Delta^{[40]}$ with respect to the objective function f_i, for all $i \in [4]$. For each $i \in [4]$, we plot $F_{i,\Delta}^{[40]}$ (in color) against $\widehat{\Delta}_{[40]}$ (in $\mathbb{R}^3$), see Figure 2.

In order to aggregate the quality of an optimizer $w \in W_\Delta^{[40]}$ over all of the objectives f_1, f_2, f_3, and f_4, we propose looking at the value

$$f^{[40]}(w) := \sum_{i \in [4]} f_i^{[40]}(w) \in [0,4].$$

The intuition behind this value is that if $w \in W_\Delta^{[40]}$ has a value $f^{[40]}(w)$ close to four, then it does well among many of the objectives and is hence a superior choice to another solution $v \in W_\Delta^{[40]}$ for which $f_i^{[40]}(v) \geq f_i^{[40]}(w)$ for some $i \in [4]$ but $f^{[40]}(v) < f^{[40]}(w)$. We refer to the following set

$$W_\Delta^{[40],\eta} := \{w \in W_\Delta^{[40]} : f^{[40]}(w) \geq (1-\eta) \cdot \Big(\max_{w \in W_\Delta^{[40]}} f^{[40]}(w) \Big)\},$$

where $\eta \in (0,1)$, as the set of portfolios with *η-superior trade-off*, and we define the set of associated *scores*

$$F_\Delta^{[40],\eta} := \{f^{[40]}(w) : w \in W_\Delta^{[40],\eta}\}.$$

We plot $F_\Delta^{[40],0.01}$ in color against $\widehat{\Delta}_{[40]}$ in Figure 3. Our plots should not be compared to figures as those in [**117**] where three of four objectives are plotted against each other with two independent and the third dependent. We give a separate plot for each objective, and we scale for comprehensibility.

TABLE 1. Selected results for $\lambda \in \widehat{\Delta}_{\Delta}^{[40],0.01} \subseteq \mathbb{R}^4$.

λ	$f_1^{[40]}(w_\lambda)$	$f_2^{[40]}(w_\lambda)$	$f_3^{[40]}(w_\lambda)$	$f_4^{[40]}(w_\lambda)$	$\lvert \operatorname{supp}(w_\lambda) \rvert$
[0.154, 0.256, 0.077, 0.513]	0.623	0.81	0.058	0.978	5
[0.026, 0.077, 0.256, 0.641]	0.601	0.825	0.05	0.98	10
[0.231, 0.41 , 0.308, 0.051]	0.581	0.854	0.034	0.989	5
[0.462, 0.513, 0.026, 0.0]	0.691	0.724	0.118	0.942	5
[0.051, 0.051, 0.205, 0.692]	0.677	0.741	0.104	0.95	5
[0.179, 0.359, 0.308, 0.154]	0.562	0.872	0.026	0.992	5
[0.462, 0.41 , 0.128, 0.0]	0.774	0.586	0.241	0.85	3
[0.282, 0.333, 0.385, 0.0]	0.752	0.625	0.203	0.881	5
[0.154, 0.231, 0.487, 0.128]	0.676	0.742	0.104	0.95	5
[0.256, 0.256, 0.077, 0.41]	0.715	0.686	0.148	0.922	5

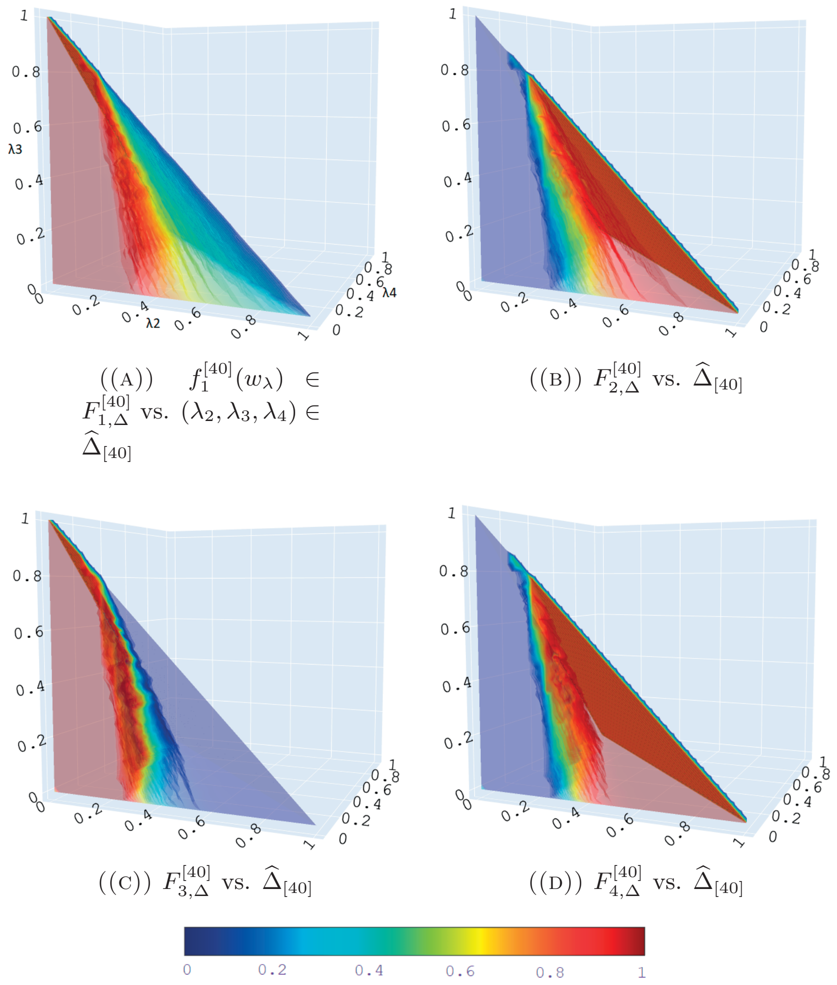
Handling the cube and sparse cases. Above, we have described the process for the simplex ($w \in \Delta^n$), but the same treatment works for the cube domain ($w \in [-1,1]^n$) and sparse domains ($w \in \Delta^n$, $|\operatorname{supp}(w)| \leq k$) and ($w \in [-1,1]^n$, $|\operatorname{supp}(w)| \leq k$). Notation-wise, the sets $W_{\square}^{[40]}$, $F_{i,\square}^{[40]}$ ($i \in [4]$), $W_{\square}^{[40],\eta}$ and $F_{\square}^{[40],\eta}$ are all defined analogously to the simplex case, now using the domain $w \in [-1,1]^n$ instead of $w \in \Delta^n$. Similarly, the sparse simplex sets are denoted by $W_{\Delta,k}^{[40]}$, $F_{i,\Delta,k}^{[40]}$ ($i \in [4]$), $W_{\Delta,k}^{[40],\eta}$ and $F_{\Delta,k}^{[40],\eta}$, where $k \in \mathbb{N}$ is an upper bound on the support size of the elements as described in Section 10.1.1. The sparse cube sets denoted $W_{\square,k}^{[40]}$, $F_{i,\square,k}^{[40]}$ ($i \in [4]$), $W_{\square,k}^{[40],\eta}$ and $F_{\square,k}^{[40],\eta}$, are defined, mutatis mutandis, in the same manner.

10.3. Numerical results

This final subsection is the culmination of the preceding subsections. For the S&P500 data considered at the end of Section 10.1.2 we compute $W_{\Delta}^{[40]}$, $F_{i,\Delta}^{[40]}(i \in [4])$, $W_{\Delta}^{[40],0.01}$, and $F_{\Delta}^{[40],0.01}$. For each $i \in [4]$ we plot $F_{i,\Delta}^{[40]}$ (in color) against the hyper-parameter set $\Delta_{[40]} \subseteq \mathbb{R}^3$. Doing so, we observe how each portfolio $w_\lambda \in W_{\Delta}^{[40]}$ makes a trade-off between the objectives f_1, f_2, f_3, and f_4. Which of the objectives are favored by w_λ is influenced by the choice of λ. For example, for $\lambda_1 = 1-(\lambda_2+\lambda_3+\lambda_4) \geq 0.4$ the portfolios w_λ tend to have values $f_1^{[40]}(w_\lambda)$ close to one, see Figure 2(a). Observations like these are useful to investors who can now visually navigate the $F_{i,\Delta}^{[40]}$ ($i \in [4]$) in Figure 2 to find a portfolio $w_\lambda \in W_{\Delta}^{[40]}$ that matches their risk preferences.

To see which $\lambda \in \Delta_{[40]}$ correspond to portfolios w_λ with a good balance of all four objectives we plot $F_{\Delta}^{[40],0.01}$ (in color) against $\Delta_{[40]}$ (resp., $\Delta_{[40]} \cap \Lambda_{\Delta}$) in Figure 3. Note that the hyper-parameters $\lambda \in \Delta_{[40]}$ for which $w_\lambda \in W_{\Delta}^{[40]} \setminus W_{\Delta}^{[40],0.01}$ are not displayed so as not to clutter the plot.

Above, we explained the process for the (dense) simplex setting, but the same treatment applies to the cube and sparse settings, resulting in analogous figures and similar observations.



((A)) $f_1^{[40]}(w_\lambda) \in F_{1,\Delta}^{[40]}$ vs. $(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta}_{[40]}$

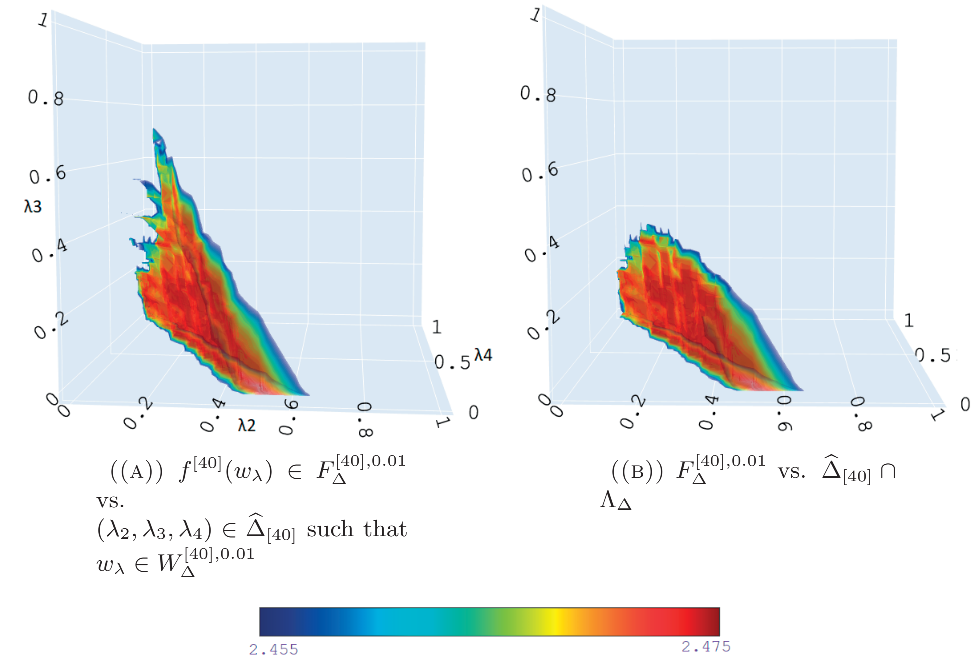
((B)) $F_{2,\Delta}^{[40]}$ vs. $\widehat{\Delta}_{[40]}$

((C)) $F_{3,\Delta}^{[40]}$ vs. $\widehat{\Delta}_{[40]}$

((D)) $F_{4,\Delta}^{[40]}$ vs. $\widehat{\Delta}_{[40]}$

FIGURE 2. This figure shows the transparent three-dimensional plots of $F_{i,\Delta}^{[40]}$ ($i \in [4]$) (in color) versus $(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta}_{[40]}$, viewed from the facet: $\{(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta} : \lambda_4 = 0\}$. For every $i \in [4]$, every point $(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta}_{[40]}$ is assigned a color $f_i^{[40]}(w_\lambda) \in [0, 1]$, where $w_\lambda \in W_\Delta^{[40]}$. Hence, red regions correspond to better values while blue regions correspond to worse values.

10.3.1. Numerical results in the simplex setting: $w \in \Delta^n$. In Figure 2, regions where the objectives f_1 and f_3 perform well (are red) overlap heavily, see Figure 2(a) and Figure 2(c). Furthermore, these regions overlap with the regions where the objectives f_2 and f_4 do poorly (are blue), namely the rear slice of the



((A)) $f^{[40]}(w_\lambda) \in F_{\Delta}^{[40],0.01}$ vs. $(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta}_{[40]}$ such that $w_\lambda \in W_{\Delta}^{[40],0.01}$

((B)) $F_{\Delta}^{[40],0.01}$ vs. $\widehat{\Delta}_{[40]} \cap \Lambda_\Delta$

FIGURE 3. This figure shows the transparent three-dimensional plot of $f^{[40]}(w_\lambda) \in F_{\Delta}^{[40],0.01}$ in color versus $(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta}_{[40]}$ such that $w_\lambda \in W_{\Delta}^{[40],0.01}$, viewed from the facet: $\{(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta} : \lambda_4 = 0\}$. In particular, every point $(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta}_{[40]}$ is assigned a color $f^{[40]}(w_\lambda) \in [0, 4]$. The values $F_{\Delta}^{[40],0.01}$ range from $0.99 \cdot 2.475$ to $\max_{w_\lambda \in W_{\Delta}^{[40],0.01}} f^{[40]}(w_\lambda) \approx 2.475$, which is indicated by the color bar. Again, red regions correspond to better values while blue regions correspond to worse values.

simplex where either λ_2 or λ_4 is small, see Figure 2(b) and Figure 2(d). The central wedge, $(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta}_{[40]}$ such that $w_\lambda \in W_{\Delta}^{[40],0.01}$, where the objectives seem to balance out is shown in Figure 3(a) along with the same wedge restricted to Λ_Δ, shown in Figure 3(b).

Recall from definition (9.11) that Λ_Δ is a set of hyper-parameters λ for which F_λ is convex over the simplex Δ^n. Further, recall that FISTA converges to a global minimizer when applied to a convex problem. With this in mind, one would expect the quality of optimizers produced by FISTA to decline as λ leaves Λ_Δ and F_λ (possibly) ceases to be convex. However, this is not apparent from our plots. Observe how there does not seem to be a change in color in the plots of Figures 2 and 3 as the hyper-parameters λ move out of the region Λ_Δ. This hints at the possibility that the local optima obtained by FISTA for hyper-parameters in $\widehat{\Delta} \setminus \Lambda_\Delta$ are not much worse than the global optima.

Lastly, observe how the set

$$\widehat{\Delta}_\Delta^{[40],0.01} := \{\lambda \in \widehat{\Delta}_{[40]} : w_\lambda \in W_\Delta^{[40],0.01}\}$$

of hyper-parameters corresponding to solutions of superior trade-off overlap with the respective sets $\{\lambda \in \widehat{\Delta} : \lambda > 0\}$ and Λ_Δ, see Figure 1(c) and Figure 3(b). The approximate volumes of these two sets and their intersection relative to $\widehat{\Delta}_\Delta^{[40],0.01}$ are as follows:

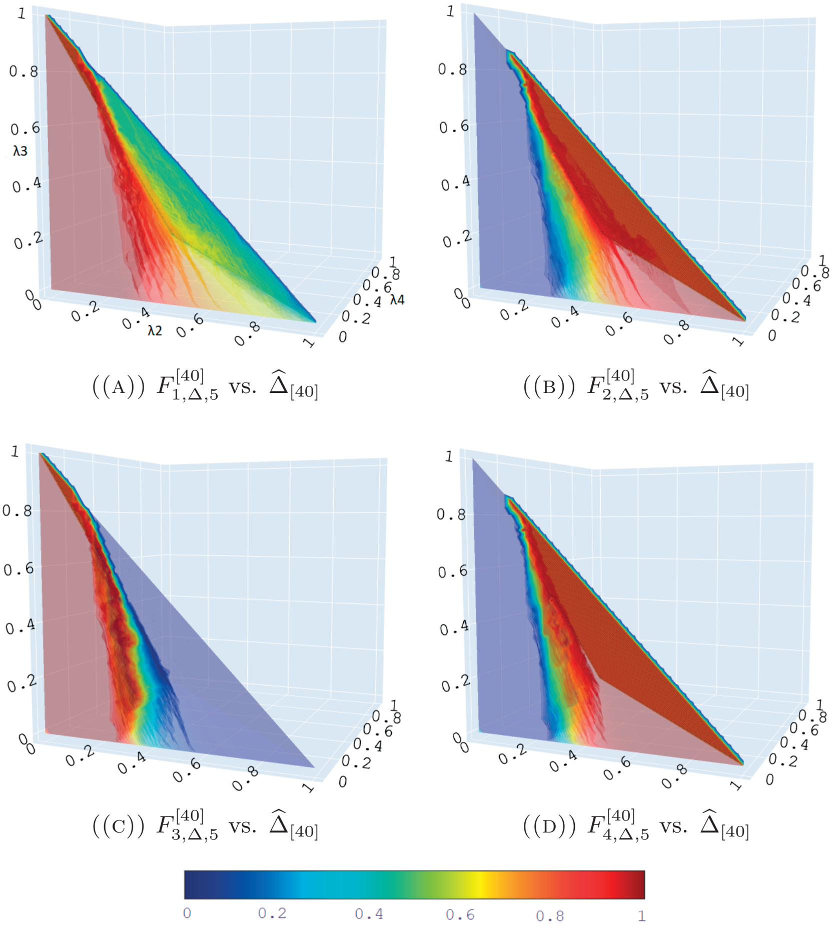
$$\frac{\mathrm{vol}(\{\lambda \in \widehat{\Delta}_\Delta^{[40],0.01} : \lambda > 0\} \cap \Lambda_\Delta)}{\mathrm{vol}(\widehat{\Delta}_\Delta^{[40],0.01})} \approx 0.77,$$
$$\frac{\mathrm{vol}(\{\lambda \in \widehat{\Delta}_\Delta^{[40],0.01} : \lambda > 0\})}{\mathrm{vol}(\widehat{\Delta}_\Delta^{[40],0.01})} \approx 0.83,$$
$$\frac{\mathrm{vol}(\widehat{\Delta}_\Delta^{[40],0.01} \cap \Lambda_\Delta)}{\mathrm{vol}(\widehat{\Delta}_\Delta^{[40],0.01})} \approx 0.90.$$

Hence, by virtue of Lemma 8.2 and Corollary 9.4, we have that approximately 77% of the optimizers with a superior trade-off in $W_\Delta^{[40],0.01}$ are guaranteed to be Pareto optimizers of the MVSK problem (9.5).

For concreteness we show in Table 1 the numerical values $f_i^{[40]}(w_\lambda)$ $(i \in [4])$ for ten randomly selected hyper-parameters $\lambda \in \widehat{\Delta}_\Delta^{[40],0.01}$. We make the following observations. First, the skewness, i.e., $f_3^{[40]}(w_\lambda)$, seems to be the weakest performing objective relative to the others. Second, variance and kurtosis, i.e., $f_2^{[40]}(w_\lambda)$ and $f_4^{[40]}(w_\lambda)$, seem to be positively correlated. Third, the associated portfolios w_λ are all sparse, with at least half of their entries zero. Eight of the ten portfolios in Table 1 have support size 5; this corroborates the data in the histogram shown in Figure 1.

In the literature, computational results are often represented in tabular form as we did in Tabel 1, see for example [**103, 95**]. Presenting results in this way for a large selection of hyper-parameters soon becomes cumbersome, especially in our case where we have $|\widehat{\Delta}_{[40]}| = 11480$. Moreover, the overall patterns are often obscured by the detail of each specific entry. In contradistinction, by representing the results as we did in Figure 2 and Figure 3, we see larger trends across the various choices of hyper-parameters λ. Hence, via the grid sampling approach of Section 10.2 and the visualizations of this section, we believe we get a better overall understanding of the relationship between λ, w_λ, and the objective values $f_i(w_\lambda)$ $(i \in [4])$ than by simply looking at a few specific values of λ. We now proceed with the other settings (sparse simplex, cube, and sparse cube).

10.3.2. The sparse simplex setting: $w \in \Delta^n$, $|\mathrm{supp}(w)| \leq 5$**.** The similarity between Figure 4 and its dense analog Figure 2 is because at least half of the portfolios $w_\lambda \in W_{\Delta,5}^{[40]}$ are from $W_\Delta^{[40]}$. Recall the histogram in Figure 1, in which more than half of the points $w_\lambda \in W_\Delta^{[40]}$ are shown to have support size five or less. Following the



((A)) $F^{[40]}_{1,\Delta,5}$ vs. $\widehat{\Delta}_{[40]}$

((B)) $F^{[40]}_{2,\Delta,5}$ vs. $\widehat{\Delta}_{[40]}$

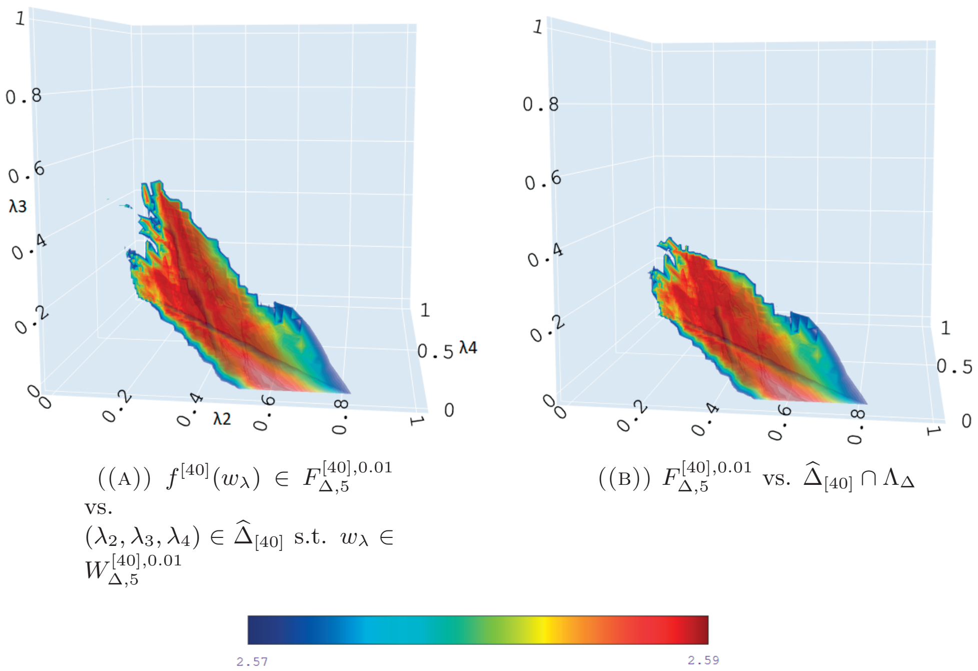
((C)) $F^{[40]}_{3,\Delta,5}$ vs. $\widehat{\Delta}_{[40]}$

((D)) $F^{[40]}_{4,\Delta,5}$ vs. $\widehat{\Delta}_{[40]}$

FIGURE 4. This figure shows the transparent three-dimensional plots of $F^{[40]}_{i,\Delta,5}$ $(i \in [4])$ (in color) versus $(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta}_{[40]}$, viewed from the facet: $\{(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta} : \lambda_4 = 0\}$.

procedure of Section 10.1.1, these portfolios are taken as they are when constructing $W^{[40]}_{\Delta,5}$.

Between Figure 5 and Figure 3, there is again much similarity. The reader may wonder why the range of values in the sparse setting Figure 5 (from 2.57 to 2.59) exceeds that of the dense setting Figure 3 (from 2.455 to 2.475). There is no contradiction here because the scaling (10.8) is different in each setting (simplex, cube, sparse, and dense). Hence, the values $F^{[40],0.01}_{\Delta,5}$ and $F^{[40],0.01}_{\Delta}$ are incomparable, similarly for the forthcoming cube setting.



((A)) $f^{[40]}(w_\lambda) \in F^{[40],0.01}_{\Delta,5}$ vs. $(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta}_{[40]}$ s.t. $w_\lambda \in W^{[40],0.01}_{\Delta,5}$

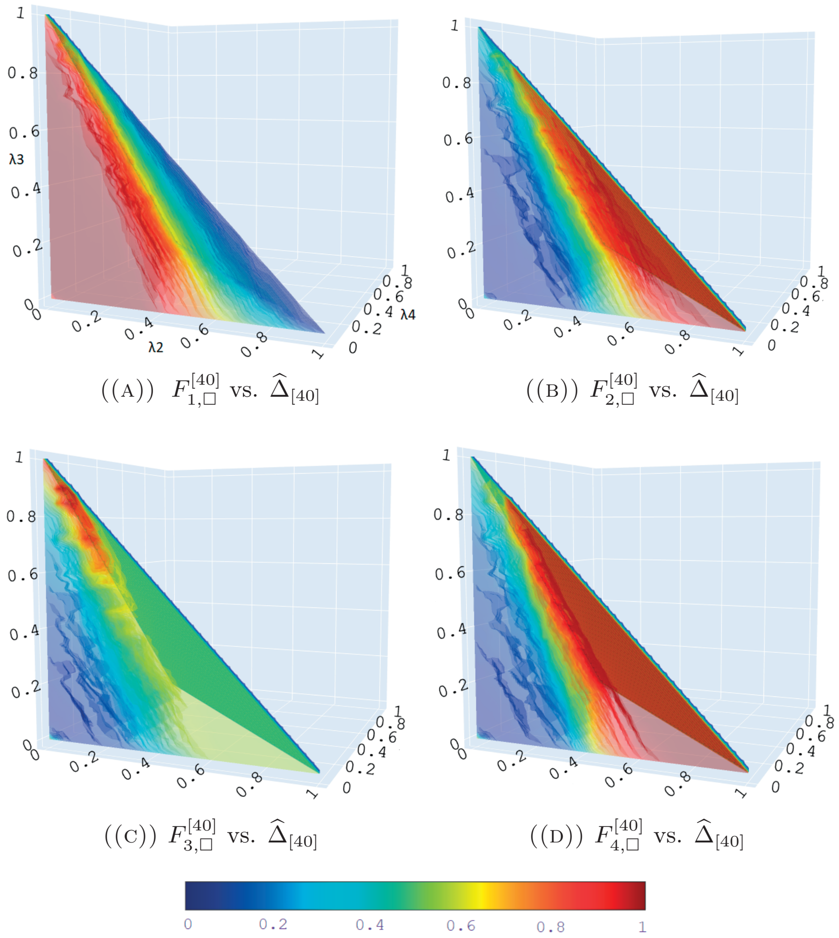
((B)) $F^{[40],0.01}_{\Delta,5}$ vs. $\widehat{\Delta}_{[40]} \cap \Lambda_\Delta$

FIGURE 5. This figure shows the transparent three-dimensional plot of $f^{[40]}(w_\lambda) \in F^{[40],0.01}_{\Delta,5}$ in color versus $(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta}_{[40]}$ such that $w_\lambda \in W^{[40],0.01}_{\Delta,5}$.

10.3.3. Numerical results in the cube setting: $w \in [-1, 1]^n$. The cube setting differs significantly from the simplex setting. Portfolios w_λ are now in $[-1, 1]^n$ and have full support (at least for all examples we have computed). Except for skewness, Figure 6(c), the figures of Figure 6 follow roughly the same pattern as in Figures 2 and 4. In the cube setting, portfolios $w_\lambda \in W^{[40]}_\square$ now require a large λ_3 to attain good values for the skewness objective, see Figure 6(c).

We observe that the portfolios with superior trade-offs are more scarce in the cube setting than in the simplex counterpart. Hence, in Figure 7, we now take $\eta = 0.025$ because the set $\widehat{\Delta}^{[40],0.025}_\square$ (of hyper-parameters corresponding to solutions of superior trade-off) gives a fuller and more informative plot than $\widehat{\Delta}^{[40],0.01}_\square$. Secondly, we observe the same "wedge" of superior portfolios we saw in Figures 3 and 5. Lastly, the portfolios $w_\lambda \in W^{[40],0.025}_\square$ that do the best in Figure 7 have $\lambda_3 \geq 0.5$, with the concentration lying outside of $\Lambda_\square$.

10.3.4. The sparse cube setting: $w \in [-1, 1]^n$, $|\operatorname{supp}(w)| \leq 5$. The results for the sparse cube setting differ vastly from the dense cube setting. The difference in results is primarily due to the portfolios $w_\lambda \in W^{[40]}_\square$ having full support and thus differing greatly from the portfolios $w_\lambda \in W^{[40]}_{\square,5}$. In particular, we see concentrations



((A)) $F_{1,\square}^{[40]}$ vs. $\widehat{\Delta}_{[40]}$

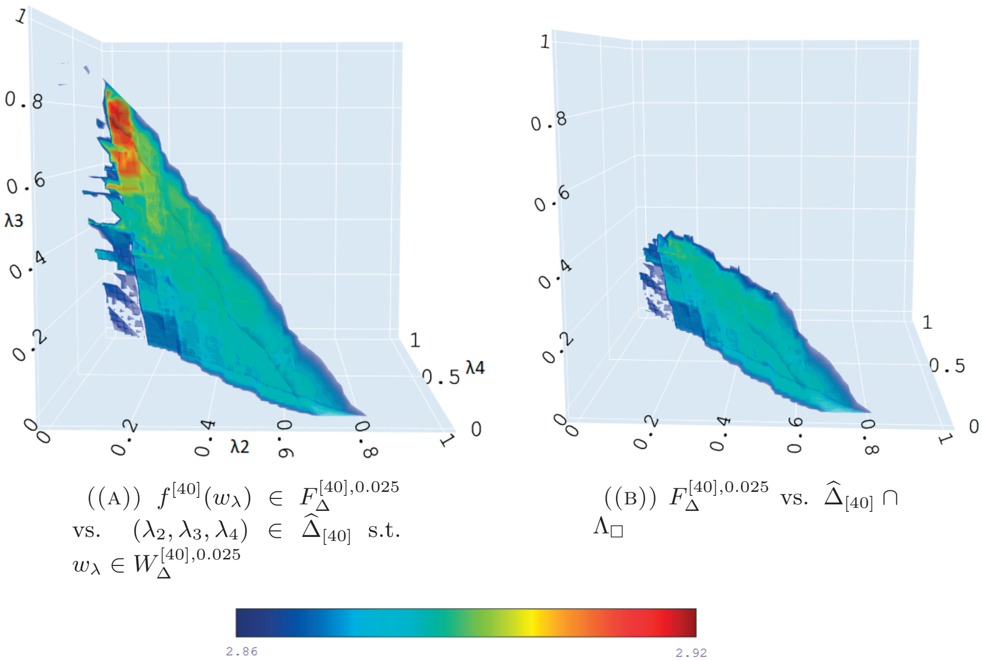
((B)) $F_{2,\square}^{[40]}$ vs. $\widehat{\Delta}_{[40]}$

((C)) $F_{3,\square}^{[40]}$ vs. $\widehat{\Delta}_{[40]}$

((D)) $F_{4,\square}^{[40]}$ vs. $\widehat{\Delta}_{[40]}$

Figure 6. This figure shows the transparent three-dimensional plots of $F_{i,\square}^{[40]}$ $(i \in [4])$ (in color) versus $(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta}_{[40]}$, viewed from the facet: $\{(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta} : \lambda_4 = 0\}$.

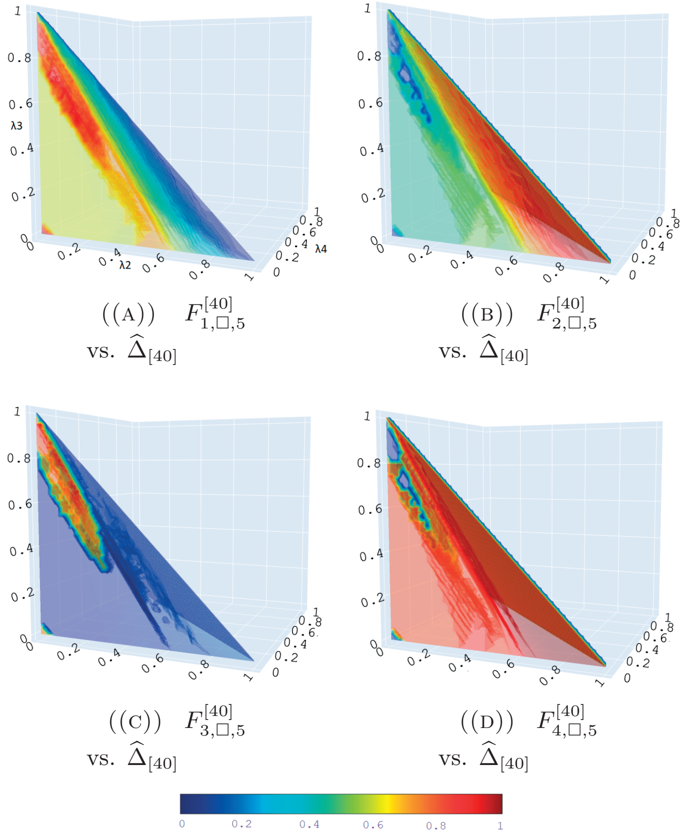
forming in the same places as in Figure 6(c), namely the upper tip of $\widehat{\Delta}_{[40]}$ where λ_3 is large. We also see a tinny concentration near $\lambda_1 = 1$. Despite the changes, we still have that the odd objectives (mean and skewness) perform better in regions where the even objectives (variance and kurtosis) do poorly and vice versa, see Figure 8. Surprisingly, $\widehat{\Delta}_{\square,5}^{[40],0.025}$ in Figure 9 is again the same "wedge"-like shape we have seen in Figures 3, 5 and 7. There are now hardly any red regions, indicating that very few points reach the higher value range.



((A)) $f^{[40]}(w_\lambda) \in F_\Delta^{[40],0.025}$ vs. $(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta}_{[40]}$ s.t. $w_\lambda \in W_\Delta^{[40],0.025}$

((B)) $F_\Delta^{[40],0.025}$ vs. $\widehat{\Delta}_{[40]} \cap \Lambda_\square$

FIGURE 7. This figure shows the transparent three-dimensional plot of $f^{[40]}(w_\lambda) \in F_\Delta^{[40],0.025}$ in color versus $(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta}_{[40]}$ such that $w_\lambda \in W_\Delta^{[40],0.025}$.



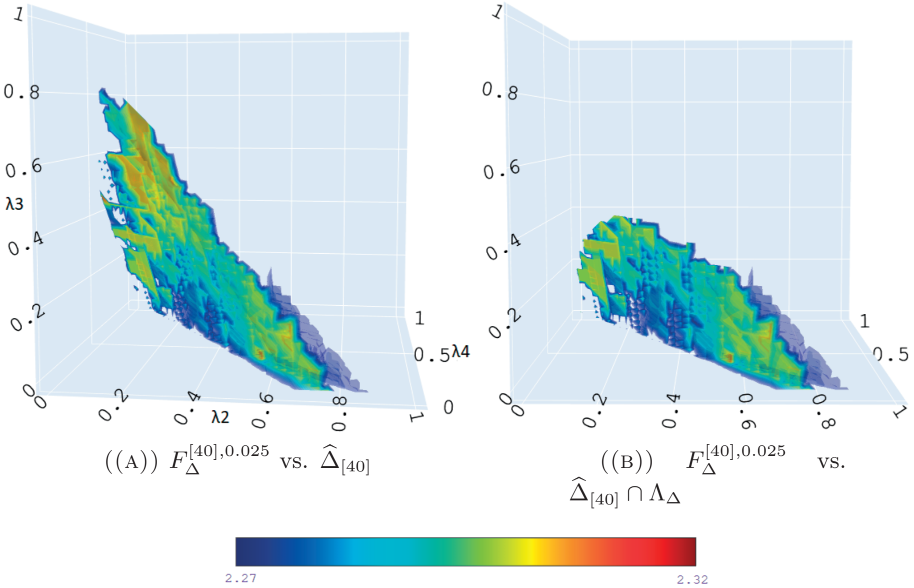
((A)) $F^{[40]}_{1,\square,5}$ vs. $\widehat{\Delta}_{[40]}$

((B)) $F^{[40]}_{2,\square,5}$ vs. $\widehat{\Delta}_{[40]}$

((C)) $F^{[40]}_{3,\square,5}$ vs. $\widehat{\Delta}_{[40]}$

((D)) $F^{[40]}_{4,\square,5}$ vs. $\widehat{\Delta}_{[40]}$

FIGURE 8. This figure shows the transparent three-dimensional plots of $F^{[40]}_{i,\square,5}$ $(i \in [4])$ (in color) versus $(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta}_{[40]}$, viewed from the facet: $\{(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta} : \lambda_4 = 0\}$.



((A)) $F_{\Delta}^{[40],0.025}$ vs. $\widehat{\Delta}_{[40]}$

((B)) $F_{\Delta}^{[40],0.025}$ vs. $\widehat{\Delta}_{[40]} \cap \Lambda_{\Delta}$

FIGURE 9. This figure shows the transparent three-dimensional plot of $f^{[40]}(w_\lambda) \in F_{\square,5}^{[40],0.025}$ in color versus $(\lambda_2, \lambda_3, \lambda_4) \in \widehat{\Delta}_{[40]}$ such that $w_\lambda \in W_{\square,5}^{[40],0.025}$.

Discussion

Disclaimer on the use of the MVSK model in portfolio selction. It should be noted that the MVSK model is heavily reliant on the estimates of the moments and is not robust to errors in the data. It is for this reason that the MVSK model and other (generalized) Markowitz models are often treated with skepticism in practice. Furthermore, simple portfolio selection strategies (like distributing one's capital equally among all assets) surprisingly outperform mean-variance models, see [**51**]. The work of Part 3 should not be seen as an endorsement of generalized mean-variance models, but rather as an observation that there is latent convexity in the MVSK problem that has been overlooked and underutilized.

Even higher-order moments than kurtosis. The formulation of the objective functions in (9.3) is well-defined for any integer $k > 4$. Hence, one can define objectives f_k with $k > 4$ in addition to those already present in (9.5) and thereby extend the model. With an extended model, one can again scalarize linearly, now using more hyper-parameters than before. Again one can characterize the Hessian of this new scalarization and possibly recover results similar to Lemma 9.2 and Corollary 9.4. There is not much motivation in the literature for this further extension. Some authors even advocate against relying on correlation-based risk measures [**152**]. We have not carried out any extensions to higher-order moments beyond kurtosis.

Minkowski distance scalarization. Much of the popularity of the Minkowski distance scalarization (8.6) comes from the fact that it is an alternative to the linear scalarization (8.8). One might reasonably ask what is gained from this more involved formulation. First, this scalarization is also neat for $\lambda > 0$ and could reveal Pareto points that the linear scalarization approach cannot. Second, the Minkowski distance function could be susceptible to signomial optimization (we elaborate more in the next paragraph). However, the benefits of the Minkowski distance scalarization must be weighed against the fact that it is much harder to interpret than linear scalarization. Moreover, one has to compute the independent optima f_k^* for $k \in [4]$, which can already be challenging in the case of $k = 3$.

The Minkowski distance formulation lends itself to a signomial optimization interpretation [**30**]. Indeed, the scalarization (8.6) applied to the MVSK problem with simplex domain has the following form:

$$\min_{w \in \Delta^n} \sum_{k \in [4]} \left| f_k(w) - f_k^* \right|^{\lambda_k},$$

where $f_k^* := \min_{w \in \Delta} (-1)^k f_k(w)$ with f_k given in (9.4) and (9.2) for each $k \in [4]$. Under the change of variable $\exp(u) := (\exp(u_i))_{i \in [n]} := (w_i)_{i \in [n]}$, the above problem can be written as a signomial optimization program

$$\begin{aligned} \min & \sum_{k \in [4]} \exp(\lambda_k v_k) \\ \text{s.t.} & \sum_{i \in [n]} \exp(u_i) = 1 \\ & \exp(v_k) = (-1)^k f_k(\exp(u)) + (-1)^{k+1} f_k^* \ (k \in [4]) \\ & u, v \in \mathbb{R}^n \end{aligned}$$

Problems of this type have been studied before and have mature methods to solve them, see [**123**]. Approaching the MVSK problem from the signomial programming direction opens a new and unexplored line of inquiry into the MVSK problem. As before, one can try to characterize the convexity of such a scalarization in the hopes of getting similar results to Lemma 9.2. Though we do not include content on this topic in this thesis nor in [**150**], we did investigate this line of inquiry but failed to reach conclusive results. As such, more research beyond this thesis is required.

The author would like to express his gratitude towards Prof. Dr. Monique Laurent (CWI Amsterdam) for diligently proofreading several versions of the paper that led to this part of the thesis and for providing critical feedback where it was most needed. Furthermore, the author thanks Shuanghua Bai and Jan Fiala (NAG Oxford) for bringing this exciting problem to his attention.

Part 4

Hypergraph-based polynomials in queueing theory

Part 4 considers a class of polynomial optimization problems that arise naturally in queueing theory when considering a particular job-occupancy model with redundancy scheduling policies. The polynomial objectives f_d in question (see (11.1)) have variables indexed by the edges of a uniform hypergraph and coefficients depending on certain patterns of unions of edges. Cardinaels, Borst, and van Leeuwaarden [**29**] conjectured that the f_d polynomials attain their global minimum over the standard simplex at the barycenter. In order to address this conjecture, we consider a related but easier-to-analyze class of polynomials p_d (see (11.2)). By exploiting the symmetry properties of these related polynomials p_d, we prove that they attain their global minimum over the standard simplex at the barycenter. Relating this result back to the original class of polynomial optimization problems posed by Cardinaels et al., we can partially prove their conjecture.

In Chapter 11, we introduce the main polynomial classes of interest (Section 11.1), we give a brief motivation from queueing theory for our interest in these polynomials (Section 11.2), and finally, we give some classical results on matrix algebras (Section 11.3) that we will use for the main proof in Chapter 12. In particular, we will look at some preliminaries on the Terwilliger algebra of the binary Hamming cube.

In Chapter 12, we give the main result of this part of the thesis; namely, we show that the polynomials p_d are convex over the standard simplex and that this implies that they attain their global minimum at the barycenter of the simplex. Our proof is based on showing the respective Hessians of p_d are positive semidefinite on the simplex. First, we deal with the simplest case in Section 12.1 to get an intuition for the proof method. Then, we deal with the general case in Section 12.2.

In the last Chapter 13, we examine the polynomials f_d. In Section 13.1, we relate the polynomials p_d to the polynomials f_d and show some partial results for f_d in the same spirit as was done in Chapter 12. For a fixed integer $d \geq 2$, we show that the polynomials f_d also attain their global minimum at the barycenter of the simplex provided the f_d's are convex. In particular, we decompose the Hessians $H(f_d)$ of f_d and observe a deep connection with the Hessians of p_d. This is instrumental in proving that $H(f_d) \succeq 0$ for a special case. We then end with a numerical motivation for why the polynomials f_d are in general convex over the simplex (Section 13.2).

Part 4 is based on our joint work with Daniel Brosch and Monique Laurent in [**27**].

CHAPTER 11

Hypergraph-based polynomials and preliminaries

This chapter has three sections. First, in Section 11.1, we introduce two classes of polynomials f_d and p_d that we will study in detail in Part 4. Second, in Section 11.2, we explain the problem from queueing theory that motivates our interest in these two polynomial classes. Third, in Section 11.3, we state some classical results pertaining to matrix algebras that we will need for proving the core result of Chapter 12.

Notation. Recall that I_n (resp., J_n) denotes the $n \times n$ identity matrix (resp., all-ones matrix). Given two integers $n, m \geq 1$, $J_{m,n}$ denotes the $m \times n$ all-ones matrix. If the sizes are clear from the context, we omit the subscripts.

Given two matrices $A, B \in \mathbb{R}^{n\times n}$ we let $A \circ B \in \mathbb{R}^{n\times n}$ denote their *Hadamard product*, with entries $(A \circ B)_{ij} = A_{ij}B_{ij}$ for $i, j \in [n]$. It is known that $A \succeq 0$ and $B \succeq 0$ imply $A \circ B \succeq 0$, which follows from the fact that the matrix $A \circ B$ is a principal submatrix of the Kronecker product $A \otimes B$.

Throughout, we let $u_1, \dots, u_m$ denote the standard basis of $\mathbb{R}^m$, where all entries of u_i are 0 except its i^{th} entry, which is equal to 1. We let $\mathrm{Sym}(n)$ denote the set of permutations of the set $[n]$. When we take the factorial $\alpha!$ of some integer-valued vector $\alpha \in \mathbb{N}^n$, we mean the product $\alpha! := \alpha_1! \cdots \alpha_m!$.

11.1. Introduction

The polynomial class of interest. Given integers $n, L \geq 2$ we set $V = [n]$ and $E = \{e \subseteq V : |e| = L\}$, so that (V, E) can be seen as the complete L-uniform hypergraph on n elements. We set $m := |E| = \binom{n}{L}$, where we omit the explicit dependence on n, L to simplify notation. Denote the standard simplex in $\mathbb{R}^m$ as follows:

$$\Delta_m = \Big\{x = (x_e)_{e\in E} \in \mathbb{R}^m : x \geq 0, \sum_{e\in E} x_e = 1\Big\},$$

and denote the *barycenter* of Δ_m by $x^* = \frac{1}{m}(1, \dots, 1)$.

Given an integer $d \geq 2$ we consider the m-variate homogeneous degree d polynomial

$$f_d(x) = \sum_{(e_1,\dots,e_d)\in E^d} \prod_{i=1}^{d} \frac{x_{e_i}}{|e_1 \cup \ldots \cup e_i|}. \tag{11.1}$$

We wish to optimize f_d over the simplex, i.e.,

$$f_d^* := \min_{x\in\Delta_m} f_d(x).$$

In [**29**], it is conjectured that the polynomial f_d attains its minimum over the simplex at the barycenter.

Conjecture 11.1. *For any integers $n, L, d \geq 2$, $f_d^* = f_d(x^*)$.*

We partially prove this conjecture by showing that a class of related polynomials all attain their minimum at the barycenter. We now describe the easier-to-analyze class of polynomials related to f_d.

An easier-to-analyze related class of polynomials. For any integer $d \geq 2$, consider the polynomial

$$p_d(x) := \sum_{(e_1,\ldots,e_d)\in E^d} \frac{1}{|e_1 \cup \ldots \cup e_d|} x_{e_1} \cdots x_{e_d}. \tag{11.2}$$

Note that, for degree $d = 2$, we have $f_2 = \frac{1}{L} p_2$. In Chapter 13, we will see that the Hessian of the polynomial p_d enters in some way as a component of the Hessian of the polynomial f_d. This forms a natural motivation for the study of the polynomials p_d. We now claim that the polynomial p_d attains its global minimum over the simplex at the barycenter.

Theorem 11.2. *For any integers $n, L, d \geq 2$,*

$$p_d(x^*) = \min_{x\in\Delta_m} p_d(x).$$

Proof. See Chapter 12. □

Thus, it follows that $f_d^* = f_d(x^*)$ when $d = 2$ and $L \geq 2$. As a further partial result, we give the next theorem, which we prove in Chapter 13.

Theorem 11.3. *For $n \geq 2$, $d = 3$ and $L = 2$,*

$$f_d(x^*) = \min_{x\in\Delta_m} f_d(x).$$

The key ingredient in proving Theorems 11.2 and 11.3 is showing that p_d and f_d are convex by proving that their respective Hessians are PSD over the simplex Δ_m. We exploit symmetry properties inherent in p_d and f_d and use Terwilliger algebras to show that the Hessians are PSD.

Example 11.4. ***Instances of p_d with $L = 2$ and $d = 1, 2, 3$.*** *As an illustration of what these polynomials look like, let us consider the polynomial p_d for edge size $L = 2$. Given a sequence $\underline{e} = (e_1, \ldots, e_d) \in E^d$, set $c_{\underline{e}} := 1/|e_1 \cup \ldots \cup e_d|$ as a short-hand for the coefficients in the definition (11.2) of the polynomial p_d.*

We need to enumerate the possible configurations of d-tuples of edges, i.e., the distinct multigraphs with d edges. Note that their number is given by the OEIS sequence A050535 [**1**]*, which takes the values $1, 3, 8, 23, 66, 212, 686$ for $d = 1, 2, 3, 4, 5, 6, 7$.*

For $d = 1$, we have $p_1(x) = \frac{1}{2}\sum_{e\in E} x_e$.

For $d = 2$ we have

$$p_2(x) = \frac{1}{2}\sum_{e\in E} x_e^2 + \frac{1}{3}\sum_{\substack{(e_1,e_2)\in E^2:\\ |e_1\cup e_2|=3}} x_{e_1}x_{e_2} + \frac{1}{4}\sum_{\substack{(e_1,e_2)\in E^2:\\ |e_1\cup e_2|=4}} x_{e_1}x_{e_2}.$$

We show in Figure 1 the three possible patterns for pairs of edges $\underline{e} = (e_1, e_2)$ and the corresponding coefficients $c_{\underline{e}}$.

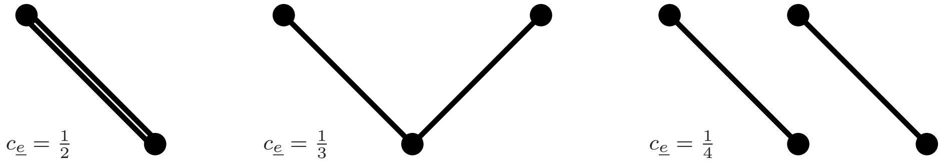

$c_{\underline{e}} = \frac{1}{2}$
$c_{\underline{e}} = \frac{1}{3}$
$c_{\underline{e}} = \frac{1}{4}$

FIGURE 1. Edge pair patterns for $d = 2, L = 2$

In the same way, for $d \geq 3$, $p_d(x) = \sum_{k=2}^{2d} \frac{1}{k} q_{d,k}(x)$, *where the summand* $q_{d,k}(x)$ *is a summation over all* d*-tuples of edges with a given pattern, depending on the cardinality of their union*

$$q_{d,k}(x) = \sum_{\substack{(e_1,\ldots,e_d)\in E^d:\\ |e_1\cup\ldots\cup e_d|=k}} x_{e_1}\cdots x_{e_d}.$$

For the case $d = 3$, *we need to consider the values* $k = 2, 3, 4, 5, 6$. *In Figure 2 we show all eight possible patterns of triplets of edges* $\underline{e} = (e_1, e_2, e_3)$ *and the corresponding coefficients* $c_{\underline{e}}$ *that contribute to the summands* $q_{3,k}$.

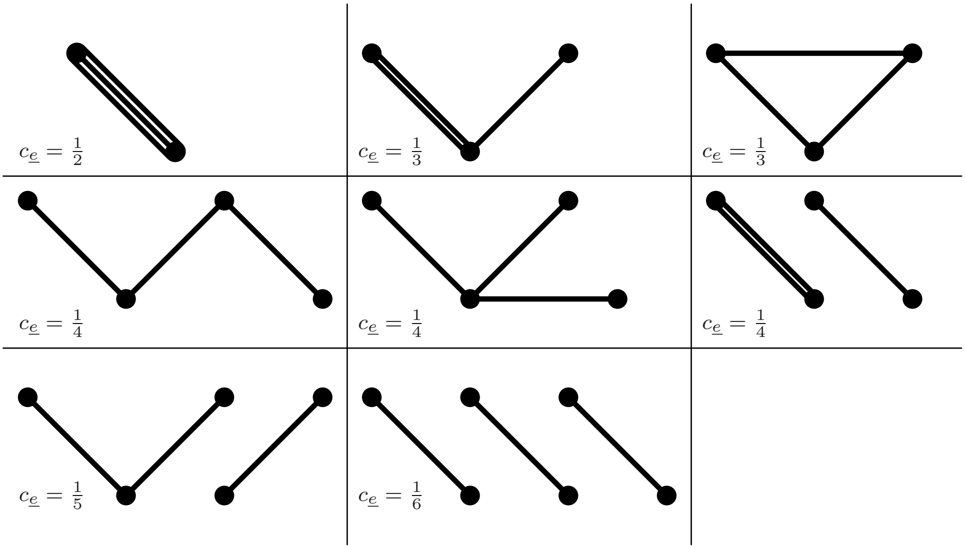

$c_{\underline{e}} = \frac{1}{2}$
$c_{\underline{e}} = \frac{1}{3}$
$c_{\underline{e}} = \frac{1}{3}$
$c_{\underline{e}} = \frac{1}{4}$
$c_{\underline{e}} = \frac{1}{4}$
$c_{\underline{e}} = \frac{1}{4}$
$c_{\underline{e}} = \frac{1}{5}$
$c_{\underline{e}} = \frac{1}{6}$

FIGURE 2. Edge pair patterns for $d = 3, L = 2$

11.2. Motivation from queueing theory

The polynomials f_d are motivated by a problem in queueing theory. The conjecture that these polynomials attain their minimum at the uniform probability distribution was presented to us by the authors of [**29**], who use an affirmative answer to this question to establish a result about the asymptotic behavior of the job occupancy in a parallel-server system with redundancy scheduling in the light-traffic regime.

In what follows, we will give only a high-level sketch of this connection, and we refer to the paper [**29**] for a detailed exposition and an extended review of the relevant literature.

A crucial mechanism that has been considered to improve the performance of parallel-server systems in queueing theory is redundancy scheduling. The key feature of this policy is that several replicas are created for each arriving job, which are then assigned to distinct servers (and then, as soon as the first of these replicas completes (or enters) service on a server, the remaining ones are stopped). The underlying idea is that sending replicas of the same job to several servers will increase the chance of having shorter queueing times. This, however, must be weighed against the risk of wastage of capacity. An important question is thus to assess the impact of redundancy scheduling policies. While most papers in the literature on redundant scheduling assume that the set of servers to which the replicas are sent is selected uniformly at random, the paper [**29**] considers the case when the set of servers is selected according to a given probability distribution. It investigates the impact of this probability distribution on the system's performance. It is shown that while the impact remains relatively limited in the heavy-traffic regime, the system occupancy is much more sensitive to the selected probability distribution in the light-traffic regime.

We will now only introduce a few elements of the model considered in [**29**] so that we can make the link to the polynomials studied in this part of the thesis. We keep our presentation high level and refer to [**29**] for details. The setting is as follows. There are n parallel servers with average speed μ. Jobs arrive as a Poisson process of rate $n\lambda$ for some $\lambda > 0$. When a job arrives, L replicas are created that are sent — with probability x_e — to a subset $e \subseteq [n]$ of L servers. Here, $L \geq 2$ is an integer and $x = (x_e)_{e\in E}$ is a probability distribution on the set $E = \{e \subseteq [n] : |e| = L\}$ of possible collections of L servers. As noted in [**29**], this can be seen as selecting an edge $e \in E$ with probability x_e in the uniform hypergraph $(V = [n], E)$ (with edge size L).

An important performance parameter is the system occupancy at time t, which is represented by a vector $(e_1, ..., e_M) \in E^M$, where $M = M(t)$ is the total number of jobs in the system and $e_i \in E$ is the collection of servers to which the replicas of the i^{th} longest job in the system have been assigned. We need three modeling assumptions. First, one needs to assume suitable stability conditions. Second, all servers should have the same speed μ. Third, the service requirements of the jobs are assumed to be independent and exponentially distributed with unit mean. Under these assumptions, the stationary distribution of the occupancy of the above edge selection is given by

$$\pi(e_1, \dots, e_M) = C \prod_{i=1}^{M} \frac{n\lambda x_{e_i}}{\mu|e_1 \cup \ldots \cup e_i|}$$

for some constant $C > 0$ ([**69**], see relation (3) in [**29**]). Following [**29**], let $Q_\lambda(x)$ be a random variable with the stationary distribution of the system occupancy when the edge selection is given by the probability vector $x = (x_e)_{e\in E}$. It then follows that, for any integer $d \geq 1$, the probability that d jobs are present in the system is given by

$$\mathbb{P}\{Q_\lambda(x) = d\} = \sum_{(e_1,\dots,e_d)\in E^d} \pi(e_1, \dots, e_d).$$

Hence, $\mathbb{P}\{Q_\lambda(x)=0\}=C$ and

$$\mathbb{P}\{Q_\lambda(x)=d\}=\mathbb{P}\{Q_\lambda(x)=0\}\Big(\frac{n\lambda}{\mu}\Big)^d\sum_{(e_1,\dots,e_d)\in E^d}\prod_{i=1}^d\frac{x_{e_i}}{|e_1\cup\ldots\cup e_i|}.$$

(See relation (11) in [**29**]). Therefore, $\mathbb{P}\{Q_\lambda(x)=d\}$ is the polynomial $f_d(x)$ (up to a scalar multiple). In [**29**], the light-traffic regime is considered, i.e., when $\lambda\downarrow 0$, in the case $L=2$. By doing a Taylor expansion, one can see that

$$\mathbb{P}\{Q_\lambda(x)=0\}=1+o(1),\ \mathbb{P}\{Q_\lambda(x)\geq d\}=\Big(\frac{n\lambda}{\mu}\Big)^d f_d(x)+o(\lambda^d)$$

(see relation (13) in [**29**]). Therefore, with $x^*=(1,\dots,1)/|E|$ denoting the uniform probability vector, we have

$$\lim_{\lambda\downarrow 0}\frac{\mathbb{P}\{Q_\lambda(x^*)\geq d\}}{\mathbb{P}\{Q_\lambda(x)\geq d\}}=\lim_{\lambda\downarrow 0}\frac{f_d(x^*)+o(1)}{f_d(x)+o(1)}.$$

Hence, if the polynomial f_d attains its minimum at the uniform distribution x^*, then one has

$$\lim_{\lambda\downarrow 0}\frac{\mathbb{P}\{Q_\lambda(x^*)\geq d\}}{\mathbb{P}\{Q_\lambda(x)\geq d\}}\leq 1.$$

This indicates that in the light-traffic regime, the system occupancy is minimized when selecting uniformly at random the assignments to the servers of the job replicas. This thus motivates the task of showing

$$f_d(x^*)=\min_{x\in\Delta_m}f_d(x),$$

for all integers $n\geq 2$, $d=3$ and $L=2$.

11.3. Preliminaries on the Terwilliger algebra

A crucial ingredient in proving $H(p_d)\succeq 0$ on Δ_m will be showing that $H(p_d)$ decomposed into matrices that (after some reduction) lie in the Terwilliger algebra of the binary Hamming cube. We begin with introducing the definition of the Terwilliger algebra $\mathcal{A}_n$ of the binary Hamming cube on n elements.

Definition 11.5 (**Terwilliger algebra of the binary Hamming cube**). *Let $\mathcal{P}([n])$ denote the collection of all subsets of the set $[n]$. For every triple of nonnegative integers i,j,t we define the $2^n\times 2^n$ matrix $D^t_{i,j}$, indexed by $\mathcal{P}([n])$, with entries*

$$\big(D^t_{i,j}\big)_{S,T}=\begin{cases}1 & if\ |S|=i,|T|=j,|S\cap T|=t,\\ 0 & else\end{cases}.$$

for sets $S,T\in\mathcal{P}([n])$. Then, the Terwilliger algebra of the binary Hamming cube*, denoted by $\mathcal{A}_n$, is defined as the (real) span of all these matrices:*

$$\mathcal{A}_n=\Big\{\sum_{i,j,t\geq 0}x^t_{i,j}D^t_{i,j}:x^t_{i,j}\in\mathbb{R}\Big\}.$$

It is easy to see that $\mathcal{A}_n$ is a *matrix $*$-algebra*, i.e., $\mathcal{A}_n$ is closed under taking linear combinations, matrix multiplications, and transposition. One way to see this is by realizing that the matrices $D^t_{i,j}$ are exactly the indicator matrices of the orbits of pairs in $\mathcal{P}([n])\times\mathcal{P}([n])$ under the element-wise action of the symmetric group $\mathrm{Sym}(n)$.

All matrix $*$-algebras can be block-diagonalized by Artin-Wedderburn theory (see [**166**], see also [**13**] for a proof).

THEOREM 11.6 (Artin-Wedderburn). *Let $\mathcal{A}$ be a matrix $*$-algebra. Then, there exist nonnegative integers d and $m_1, \dots, m_d$ and a $*$-algebra isomorphism*

$$\varphi\colon \mathcal{A} \to \bigoplus_{k=1}^{d} \mathbb{C}^{m_k \times m_k}.$$

The important property here is that φ is an algebra isomorphism. Hence, we know that this isomorphism maintains positive semidefiniteness: for any matrix $A \in \mathcal{A}$, we have $A \succeq 0 \iff \varphi(A) \succeq 0$. Moreover, the matrix $\varphi(A)$ is block-diagonal, with d diagonal blocks of sizes $m_1, \dots, m_d$. This is a crucial property that can be exploited to get a more efficient way of encoding positive semidefiniteness of matrices in $\mathcal{A}$.

The explicit block-diagonalization of the Terwilliger algebra $\mathcal{A}_n$ was given by Schrijver [**144**].

THEOREM 11.7 (Schrijver [**144**]). *The Terwilliger algebra $\mathcal{A}_n$ can be block-diagonalized into $\lfloor \frac{n}{2} \rfloor + 1$ blocks, of sizes $m_k = n - 2k + 1$ for $k = 0, \dots, \lfloor \frac{n}{2} \rfloor$. The algebra isomorphism φ sends the matrix*

$$A = \sum_{i,j,t=0}^{n} x_{i,j}^{t} D_{i,j}^{t}$$

to the block-matrix $\varphi(A) = \oplus_{k=0}^{\lceil n/2 \rceil} B_k$, where the matrix $B_k \in \mathbb{R}^{m_k \times m_k}$ is given by

$$B_k := \left(\binom{n-2k}{i-k}^{-\frac{1}{2}} \binom{n-2k}{j-k}^{-\frac{1}{2}} \sum_{t} \beta_{i,j,k}^{t} x_{i,j}^{t} \right)_{i,j=k}^{n-k} \tag{11.3}$$

for $k = 0, 1, \dots, \lfloor \frac{n}{2} \rfloor$. Here, for any nonnegative integers i, j, t, k, we set

$$\beta_{i,j,k}^{t} := \sum_{\ell=0}^{n} (-1)^{\ell-t} \binom{\ell}{t} \binom{n-2k}{n-k-\ell} \binom{n-k-\ell}{i-\ell} \binom{n-k-\ell}{j-\ell}. \tag{11.4}$$

In particular, we have

$$\sum_{i,j,t=0}^{n} x_{i,j}^{t} D_{i,j}^{t} \succeq 0 \iff B_k \succeq 0 \quad \text{for } k = 0, 1, \dots, \lfloor \frac{n}{2} \rfloor. \tag{11.5}$$

CHAPTER 12

Convexity of p_d

The crux for proving Theorem 11.2 is showing that the polynomials p_d are convex over the standard simplex. We explain why this is the case in Lemma 12.1, after which we devote the rest of this chapter to proving convexity by showing that the Hessians $H(p_d)$ are PSD. First, in a special case (Section 12.1), then in the general case (Section 12.2), which is significantly more technical.

Convexity implies optimality at the barycenter. To show p_d is optimal at the barycenter x^* of Δ_m, it suffices to show that p_d is convex over Δ_m. This relies on a symmetry argument that exploits the fact that the polynomial p_d is invariant under the permutations of the edge set E.

Lemma 12.1. *Assume the polynomial p_d is convex on the simplex Δ_m. Then*

$$p_d(x^*) = \min_{x \in \Delta_m} p_d(x).$$

Proof. For any tuple $(e_1, \dots, e_d) \in E^d$, the coefficient of the monomial $x_{e_1} \cdots x_{e_d}$ in p_d is $1/|e_1 \cup \ldots \cup e_d|$, which depends only on the cardinality of the set $e_1 \cup \ldots \cup e_d$.

Any permutation $\sigma \in \mathrm{Sym}(n)$ of $[n]$ induces a permutation of E (still denoted σ) by setting $\sigma(e) = \{j_{\sigma(1)}, \dots, j_{\sigma(L)}\}$ for $e = \{j_1, \dots, j_L\} \in E$. In turn, σ acts on Δ_m by setting $\sigma(x) = (x_{\sigma(e)})_{e \in \Delta_m}$ for $x = (x_e)_{e \in E} \in \Delta_m$. Observe that p_d is invariant under this action of permutations $\sigma \in \mathrm{Sym}(n)$. Indeed, for any $\sigma \in \mathrm{Sym}(n)$, we have

$$\begin{aligned} \sigma(p_d)(x) = p_d(\sigma(x)) \quad &= \textstyle\sum_{(e_1,\dots,e_d) \in E^d} \frac{1}{|e_1 \cup \ldots \cup e_d|} x_{\sigma(e_1)} \cdots x_{\sigma(e_d)} \\ &= \textstyle\sum_{(f_1,\dots,f_d) \in E^d} \frac{1}{|\sigma^{-1}(f_1) \cup \ldots \cup \sigma^{-1}(f_d)|} x_{f_1} \cdots x_{f_d} \\ &= \textstyle\sum_{(f_1,\dots,f_d) \in E^d} \frac{1}{|f_1 \cup \ldots \cup f_d|} x_{f_1} \cdots x_{f_d} \\ &= p_d(x). \end{aligned}$$

Thus for any global minimizer $\widetilde{x} \in \Delta_m$ of p_d, and any permutation $\sigma \in \mathrm{Sym}(n)$, the permuted point $\sigma(\widetilde{x})$ belongs to Δ_m and $p_d(\widetilde{x}) = p_d(\sigma(\widetilde{x}))$. Consider the full symmetrization of $\widetilde{x}$, i.e.,

$$x^{\mathrm{sym}} := \frac{1}{n!} \sum_{\sigma \in \mathrm{Sym}(n)} \sigma(\widetilde{x}),$$

and observe that $x^{\mathrm{sym}} \in \Delta_m$ and $x^{\mathrm{sym}} = x^* = (1/m)(1, \dots, 1)$. Hence, x^* is a global minimizer of p_d in Δ_m because by convexity and the above we have

$$p_d(x^*) = p_d(x^{\mathrm{sym}}) \leq \frac{1}{n!} \sum_{\sigma \in \mathrm{Sym}(n)} p_d(\sigma(\widetilde{x})) = p_d(\widetilde{x}) = \min_{x \in \Delta_m} p_d(x). \qquad \square$$

We are left with the task of showing that the polynomial p_d is convex over the simplex Δ_m or, equivalently, that its Hessian matrix is PSD, i.e.,

$$H(p_d)(x) = \Big(\frac{\partial^2}{\partial x_e \partial x_f} p_d(x)\Big)_{e,f\in E} \succeq 0 \ (x \in \Delta_m).$$

12.1. If $d = 2$ and $L = 2$, then p_d is convex

Consider the polynomial

$$p_2(x) = \sum_{e,f\in E} \frac{1}{|e \cup f|} x_e x_f,$$

where $E = \{e \subseteq [n] : |e| = 2\}$. We show that the Hessian matrix $H(p_2)$ of p_2 is PSD. Observe that $H(p_2) = 2M$, where M is the matrix indexed by E with entries

$$M_{e,f} = \frac{1}{|e \cup f|} \qquad \text{for } e, f \in E. \tag{12.1}$$

The matrix M can be expressed as a linear combination of the matrices A_2, A_3, A_4, which are also indexed by E, and have entries

$$(A_s)_{e,f} = \begin{cases} 1 & \text{if } |e \cup f| = s, \\ 0 & \text{otherwise.} \end{cases} \quad (s = 2, 3, 4).$$

Some inspection will reveal that $A_2 = I$ and $A_2 + A_3 + A_4 = J$. Observe now that

$$M = \frac{1}{2}I + \frac{1}{3}A_3 + \frac{1}{4}A_4 = \frac{1}{4}I + \frac{1}{12}A_3 + \frac{1}{4}J = \frac{1}{12}I + \frac{1}{4}J + \frac{1}{12}(A_3 + 2I). \tag{12.2}$$

Thus, if we can show that $A_3 + 2I \succeq 0$, then $M \succeq 0$ follows, and p_2 is convex. The reader can now verify by direct inspection that $A_3 + 2I = \Gamma_n \Gamma_n^T \succeq 0$, where

$$\Gamma_n = \big(|e \cap \{i\}|\big)_{e\in E,\ i\in[n]}.$$

Remark 12.2. *Note that the matrices $A_2 = I, A_3, A_4$ generate the Bose-Mesner algebra of the Johnson scheme J_2^n, with length n and weight 2, and thus the matrix M belongs to this Bose-Mesner algebra (see* [**50**] *for details on the Johnson scheme). For arbitrary degree $d \geq 3$ and edge size $L = 2$, one could proceed to show that the Hessian matrix of p_d is convex by using a similar symmetry reduction based on the Bose-Mesner algebra of the Johnson scheme J_2^p for suitable values of p. However, for general edge size $L \geq 3$, we will need to use a richer algebra, namely the Terwilliger algebra of the Hamming cube. Hence, in the rest of the section, we will treat the general case $d \geq 2$ and $L \geq 2$.*

12.2. If $d \geq 2$ and $L \geq 2$, then p_d is convex over the simplex

Let $E = \{e \subseteq [n] : |e| = L\}$, $d \geq 2$, and $L \geq 2$. We repeat the definition of polynomial p_d from relation (11.2):

$$p_d(x) = \sum_{(e_{i_1},\ldots,e_{i_d})\in E^d} \frac{1}{|e_{i_1} \cup \ldots \cup e_{i_d}|} x_{e_{i_1}} \cdots x_{e_{i_d}}. \tag{12.3}$$

12.2.1. Characterization of the coefficients of the polynomial p_d. We begin by getting the explicit coefficients of the polynomial p_d expressed in the standard monomial basis. The basic fact we will now use is that the coefficients depend only on the set of distinct edges that are present in the tuple $(e_{i_1}, \ldots, e_{i_d}) \in E^d$ and not on their multiplicities.

Recall that $m = |E|$ and label all the edges as $e_1, \ldots, e_m$ so that $E = \{e_1, \ldots, e_m\}$. For a d-tuple $\underline{e} := (e_{i_1}, \ldots, e_{i_d}) \in E^d$ with $i_1, \ldots, i_d \in [m]$ define the m-tuple

$$\alpha(\underline{e}) = \Big(\big|\{j \in [d] : i_j = \ell\}\big|\Big)_{\ell \in [m]} \in \mathbb{N}^m_d.$$

Then we have $|\alpha(\underline{e})| = d$ and

$$x_{e_{i_1}} \cdots x_{e_{i_d}} = x_{e_1}^{\alpha(\underline{e})_1} \cdots x_{e_m}^{\alpha(\underline{e})_m} = x^{\alpha(\underline{e})}.$$

Moreover, all d-tuples $\underline{e} = (e_{i_1}, \ldots, e_{i_d}) \in E^d$ with $\operatorname{supp}(\alpha(\underline{e})) = \operatorname{supp}(\alpha)$, for some fixed $\alpha \in \mathbb{N}^m_d$, have the same associated coefficients, namely,

$$c_\alpha := \frac{1}{|e_{i_1} \cup \ldots \cup e_{i_d}|}. \tag{12.4}$$

As an example, for $d = n = m = 3$, $\alpha = (1, 0, 1)$:

$$c_\alpha = \frac{1}{|e_1 \cup e_3 \cup e_1|} = \frac{1}{|e_3 \cup e_1 \cup e_1|}.$$

Lemma 12.3. *The polynomial p_d from (12.3) can be reformulated as follows:*

$$p_d(x) = \sum_{\alpha \in \mathbb{N}^m_d} c_\alpha \frac{d!}{\alpha!} x^\alpha. \tag{12.5}$$

Proof. Using the definition of the coefficients c_α, we can rewrite p_d as

$$p_d(x) = \sum_{\alpha \in \mathbb{N}^m_d} \Big(\sum_{\substack{\underline{e} = (e_{i_1}, \ldots, e_{i_d}) \in E^d: \\ \alpha(\underline{e}) = \alpha}} \frac{1}{|e_{i_1} \cup \ldots \cup e_{i_d}|} \Big) x^\alpha = \sum_{\alpha \in \mathbb{N}^m_d} \Big(\sum_{\underline{e} \in E^d : \alpha(\underline{e}) = \alpha} c_\alpha \Big) x^\alpha,$$

which is equal to $\sum_{\alpha \in \mathbb{N}^m_d} c_\alpha \frac{d!}{\alpha!} x^\alpha$. Here, for this last equality, we use the monomial theorem, which claims the identity

$$\left(\sum_{i=1}^m x_i\right)^d = \sum_{\alpha \in \mathbb{N}^m_d} \frac{d!}{\alpha!} x^\alpha,$$

or, equivalently, the number of d-tuples $\underline{e} \in E^d$ for which $\alpha(\underline{e}) = \alpha$ is equal to $\frac{d!}{\alpha!}$. □

12.2.2. Decomposing the Hessian of p_d.

Lemma 12.4. *The Hessian of the polynomial p_d is the matrix*

$$H(p_d)(x) := \Big(\frac{\partial^2 p_d(x)}{\partial x_{e_i} \partial x_{e_j}}\Big)_{i,j=1}^m = \sum_{\gamma \in \mathbb{N}^m_{d-2}} \frac{d!}{\gamma!} x^\gamma M_\gamma,$$

where, for any $\gamma \in \mathbb{N}^m_{d-2}$, $M_\gamma := (c_{\gamma + u_i + u_j})_{i,j=1}^m$.

PROOF. The result follows from looking at the partial derivatives of p_d

$$\frac{\partial p_d(x)}{\partial x_{e_i}} = \sum_{\alpha\in\mathbb{N}^m_d:\alpha_i\geq 1} \frac{d!}{(\alpha-u_i)!}c_\alpha x^{\alpha-u_i} = \sum_{\beta\in\mathbb{N}^m_{d-1}} \frac{d!}{\beta!}c_{\beta+u_i}x^\beta,$$

and the second-order partial derivatives

$$\frac{\partial^2 p(x)}{\partial x_{e_j}\partial x_{e_i}} = \sum_{\beta\in\mathbb{N}^m_{d-1}:\beta_j\geq 1} \frac{d!}{(\beta-u_j)!}c_{\beta+u_i}x^{\beta-u_j} = \sum_{\gamma\in\mathbb{N}^m_{d-2}} c_{\gamma+u_i+u_j}\frac{d!}{\gamma!}x^\gamma.$$

□

Hence, if we can show that the matrices M_γ in Lemma 12.4 are all PSD, then it follows directly that $H(p_d)(x)\succeq 0$ for any x in the standard simplex.

We proceed with two successive PSD preserving reductions of M_γ into smaller matrices. After the final reduction, these smaller matrices will be shown to be block-diagonalizeable as a consequence of belonging to the Terwilliger algebra. The resulting block-diagonalized matrices will be shown to be PSD by giving their explicit Cholesky factorization.

First reduction. For any integer-valued vector $\gamma\in\mathbb{N}^m$, define its (edge) *support* as the set $S_\gamma=\{e\in E:\gamma_e\geq 1\}$ and let

$$W_\gamma = \bigcup_{e\in S_\gamma} e$$

denote the subset of elements of $V=[n]$ that are covered by some edge in the support of γ. Then, for any $i,j\in[m]$, the support of $\gamma+u_i+u_j$ is the set $S_\gamma\cup\{e_i,e_j\}$ and we have

$$(M_\gamma)_{e_i,e_j} = c_{\gamma+u_i+u_j} = \frac{1}{|W_\gamma\cup e_i\cup e_j|}.$$

Hence the matrix M_γ depends only on the set W_γ (and not on the specific choice of the sequence γ). This justifies defining the matrices

$$M_W = \Big(\frac{1}{|W\cup e\cup f|}\Big)_{e,f\in E} \tag{12.6}$$

for any set $W\subseteq V=[n]$. Hence, for any $\gamma\in\mathbb{N}^m_{d-2}$, we have:

$$M_\gamma = M_{W_\gamma}. \tag{12.7}$$

Summarizing, we have shown the following result:

LEMMA 12.5. *Assume that the matrices M_W from (12.6) are positive semidefinite for all $W\subseteq V$ with $|W|\geq L$ (if $d\geq 3$) and $|W|\leq L(d-2)$. Then the polynomial p_d is convex over the standard simplex.*

If $d=2$, then there is only one matrix to check, namely the matrix $M_\emptyset$ (for $W=\emptyset$). Note that the matrix $M_\emptyset$ coincides with the matrix in (12.1), so we already know it is positive semidefinite when $L=2$. However, if $d\geq 3$, one needs to check all the matrices of the form M_W in (12.6).

Second reduction. We now link these matrices M_W to the Terwilliger algebra. Observe that in the matrix M_W, there are identical rows and columns, and the second reduction consists simply in removing duplicate rows and columns. Set $p := |W|$ and $U := V \setminus W$, so that $|U| = n - p$. In addition, set

$$F := \{e \setminus W : e \in E\} = \{e \subseteq U : L - p \leq |e| \leq L\}, \tag{12.8}$$

which consists of the intersections with U of the edges in E. Then, $F = E$ when $p = 0$ and the condition $|e| \geq L - p$ is redundant when $p \geq L$. Now we consider the following matrix M_p, which is indexed by F, with entries

$$(M_p)_{e,f} = \frac{1}{p + |e \cup f|} \quad \text{for } e, f \in F. \tag{12.9}$$

Note that for $p = 0$ the matrix M_0 coincides with the matrix $M_\emptyset$ in (12.6) (and with the matrix in (12.1)). The next lemma shows that M_p is obtained from M_W by deleting duplicate rows and columns.

Lemma 12.6. *Let $L \geq 2$ and $d \geq 2$. Consider the matrices M_W in (12.6) and M_p in (12.9). The following assertions are equivalent:*

(i) *$M_W \succeq 0$ for all $W = e_1 \cup \ldots \cup e_{d-2}$ with $e_1, \ldots, e_{d-2} \in E$.*
(ii) *$M_p \succeq 0$ for all $p \leq L \cdot (d-2)$ such that $p \geq L$ if $d \geq 3$.*

Proof. If $d = 2$, the result holds since $M_0 = M_\emptyset$ as observed above. So, assume now $d \geq 3$. Let $W = e_1 \cup \ldots \cup e_{d-2}$, where $e_1, \ldots, e_{d-2} \in E$, and set $p = |W|$. Consider the partition of the set E into

$$E = \bigcup_{i=0}^{L} E_i, \; E_i := \{e \in E : |e \setminus W| = i\}.$$

With respect to this partition of its index set, the matrix M_W has the following block-form:

$$M_W = \left(\begin{array}{c|c|c|c} M_W^{0,0} & M_W^{0,1} & \cdots & M_W^{0,L} \\ \hline M_W^{1,0} & M_W^{1,1} & \cdots & M_W^{1,L} \\ \hline \vdots & \vdots & \ddots & \vdots \\ \hline M_W^{L,0} & M_W^{L,1} & \cdots & M_W^{L,L} \end{array}\right),$$

where the block $M_W^{i,j}$ has its rows indexed by E_i and its columns by E_j. Note that, if two edges $e, e' \in E$ satisfy $e \setminus W = e' \setminus W$, then the two rows of M_W indexed by e and e' coincide: for any $f \in E$ we have

$$\left(M_W^{i,j}\right)_{e,f} = \frac{1}{|W| + |(e \cup f) \setminus W|} = \frac{1}{|W| + |(e' \cup f) \setminus W|} = \left(M_W^{i,j}\right)_{e',f}.$$

In fact, after removing these duplicate rows (and columns) and keeping only one copy for each subset of $U = V \setminus W$, we obtain the matrix

$$\left(\begin{array}{c|c|c|c} M_p^{0,0} & M_p^{0,1} & \cdots & M_p^{0,L} \\ \hline M_p^{1,0} & M_p^{1,1} & \cdots & M_p^{1,L} \\ \hline \vdots & \vdots & \ddots & \vdots \\ \hline M_p^{L,0} & M_p^{L,1} & \cdots & M_p^{L,L} \end{array}\right),$$

which coincides with the matrix M_p in (12.9). Indeed, the above matrix is indexed by the set F in (12.8) and its block-form is with respect to the partition

$$F = \bigcup_{i=0}^{L} F_i, \;\; F_i := \{e \subseteq U : |e| = i\}.$$

So the block $M_p^{i,j}$ has its rows indexed by F_i, its columns indexed by F_j, and its entries are

$$(M_p^{i,j})_{e,f} = \frac{1}{p+|e \cup f|} = \frac{1}{p+i+j-|e\cap f|} \quad \text{for } e \in F_i, f \in F_j. \tag{12.10}$$

As the matrix M_p arise from M_W by removing its duplicate rows and columns, it is clear that

$$M_W \succeq 0 \iff M_p \succeq 0.$$

This concludes the proof. □

The next section shows that the matrices M_p are positive semidefinite for all $0 \le p \le n$ by exploiting their link to Terwilliger algebras.

12.2.3. The matrices M_p are PSD. Fix an integer $0 \le p \le n$ and consider the matrix M_p in (12.9), which has a block-form with blocks as in (12.10). We will show that $M_p \succeq 0$ because it belongs to the Terwilliger algebra $\mathcal{A}_{n-p}$ (introduced in Section 11.3).

Observe that relation (12.10) provides the explicit correspondence between the blocks $M_p^{i,j}$ of M_p and the generating matrices $D_{i,j}^t$ of the algebra $\mathcal{A}_{n-p}$:

$$M_p = \sum_{i=0}^{L}\sum_{j=0}^{L}\sum_{t=0}^{\min\{i,j\}} \frac{1}{p+i+j-t} D_{i,j}^t = \sum_{i=0}^{L}\sum_{j=0}^{L}\sum_{t=0}^{\min\{i,j\}} x_{i,j}^t D_{i,j}^t,$$

after setting

$$x_{i,j}^t = \frac{1}{p+i+j-t}. \tag{12.11}$$

Showing that $M_p \succeq 0$***.*** Since $M_p \in A_{n-p}$ we can use Theorem 11.7 to show that $M_p \succeq 0$ provided we can show that $B_k \succeq 0$, where the matrices B_k are defined in (11.3), with n now replaced with $n-p$.

Fix the integers p and k. We now proceed to explicitly show that $B_k \succeq 0$. To simplify the notation we introduce the following parameters

$$a(i) := \binom{n-p-2k}{i-k}^{-\frac{1}{2}}, \quad b(\ell, i) := \binom{n-p-k-\ell}{i-\ell}, \quad c(\ell) := \binom{n-p-2k}{n-p-k-\ell},$$

which are defined for any integers $i, \ell \in \mathbb{Z}$. Note that we may omit the obvious bounding conditions on i and ℓ since the corresponding parameters are zero if these conditions are not satisfied; for instance, $a(i) = 0$ if $i < k$ and $b(\ell, i) = 0$ if $\ell > i$.

Using this new notation we have that

$$B_k = \left(a(i)a(j) \sum_{t=0}^{min\{i,j\}} \beta_{i,j,k}^t x_{i,j}^t \right)_{i,j=k}^{n-p-k}, \tag{12.12}$$

where

$$\beta_{i,j,k}^t := \sum_{\ell=0}^{n-p} (-1)^{\ell-t} \binom{\ell}{t} c(\ell) b(\ell,i) b(\ell,j). \tag{12.13}$$

Lemma 12.7. *We have*

$$\sum_{t=0}^{\min\{i,j\}} \beta_{i,j,k}^t x_{i,j}^t = \sum_{\ell=0}^{\min\{i,j\}} c(\ell) b(\ell,i) b(\ell,j) \int_0^1 g(\ell,z) z^{i+j} dz,$$

where $g(\ell,z) := z^{p-1}(\frac{1-z}{z})^\ell$ for $z \in (0,1]$.

Proof. First, we exchange the summations in t and ℓ to obtain

$$\sum_{t=0}^{\min\{i,j\}} \beta_{i,j,k}^t x_{i,j}^t = \sum_{\ell=0}^{\min\{i,j\}} c(\ell) b(\ell,i) b(\ell,j) \left(\sum_{t=0}^{\ell} \frac{1}{p+i+j-t} (-1)^{\ell-t} \binom{\ell}{t} \right). \tag{12.14}$$

Observe that, for any integer $i \geq 1$, we have $\frac{1}{i} = \int_0^1 z^{i-1} dz$, which permits us to give an integral reformulation for the scalars $x_{i,j}^t$ in (12.11) as follows:

$$\frac{1}{p+i+j-t} = \int_0^1 z^{p+i+j-t-1} dz.$$

Using this integral representation we can reformulate the inner summation appearing in (12.14) as follows:

$$\begin{aligned}
\sum_{t=0}^{\ell} \frac{1}{p+i+j-t} (-1)^{\ell-t} \binom{\ell}{t} &= \sum_{t=0}^{\ell} (-1)^{\ell-t} \binom{\ell}{t} \int_0^1 z^{p+i+j-t-1} dz \\
&= \int_0^1 z^{p+i+j-1} (-1)^\ell \left(\sum_{t=0}^{\ell} \left(-\frac{1}{z}\right)^t \binom{\ell}{t} \right) dz \\
&\overset{(*)}{=} \int_0^1 z^{p+i+j-1} (-1)^\ell \left(1 - \frac{1}{z}\right)^\ell dz \\
&= \int_0^1 z^{p+i+j-1} (-1)^\ell \left(\frac{z-1}{z}\right)^\ell dz \\
&= \int_0^1 z^{p-1} \left(\frac{1-z}{z}\right)^\ell z^{i+j} dz = \int_0^1 g(\ell,z) z^{i+j} dz.
\end{aligned}$$

The equality marked with (*) follows from use of the binomial theorem. This concludes the proof. □

Lemma 12.8. *We have $B_k \succeq 0$.*

Proof. Note that in the result of Lemma 12.7, since $b(\ell,i)b(\ell,j)=0$ if $\ell > \min\{i,j\}$, we may replace the summation on ℓ from $0 \le \ell \le \min\{i,j\}$ to $0 \le \ell \le n-p$. This implies:

$$\begin{aligned} B_k &= \Big(a(i)a(j)\sum_{t=0}^{n-p}\beta^t_{i,j,k}x^t_{i,j}\Big)_{i,j=k}^{n-p-k} \\ &= \int_0^1 \Big(\sum_{\ell=0}^{n-p} g(\ell,z)c(\ell)\underbrace{(z^i a(i)b(\ell,i))}_{=:h(\ell,z,i)}\underbrace{(z^j a(j)b(\ell,j))}_{=:h(\ell,z,j)}\Big)_{i,j=k}^{n-p-k} dz \\ &= \sum_{\ell=0}^{n-p}\int_0^1 \underbrace{g(\ell,z)c(\ell)}_{\ge 0}\underbrace{\Big(h(\ell,z,i)h(\ell,z,j)\Big)_{i,j=k}^{n-p-k}}_{\succeq 0} dz \succeq 0 \end{aligned}$$

Here, we used that, for any $\ell \in [0, n-p]$, the function $g(\ell,z) \ge 0$ on $(0,1]$. □

Concluding the proof that p_d is convex. We summarize the chain of equivalences we have built with the following schematic:

$$\begin{aligned} H(p_d)(x) \succeq 0 &\overset{Lem.\ 12.4}{\iff} M_\gamma \succeq 0 \ (\gamma \in \mathbb{N}^m_{d-2}) \\ &\overset{(12.7)}{\iff} M_{W_\gamma} \succeq 0 \ (\gamma \in \mathbb{N}^m_{d-2}) \\ &\overset{Lem.\ 12.6}{\iff} M_p \succeq 0 \ (p \le L\cdot(d-2)) \\ &\overset{Th.11.7}{\iff} B_k \succeq 0 \ (k = 0,1,\dots,\lfloor\frac{n}{2}\rfloor), \end{aligned}$$

where, the final statement, i.e., $B_k \succeq 0$ $(k = 0,1,\dots,\lfloor\frac{n}{2}\rfloor)$, is valid via Lemma 12.8. Thus, p_d is convex over the simplex, and Theorem 11.2 is proved.

CHAPTER 13

Investigating the polynomials f_d

We consider the second class of polynomials f_d from (11.1), namely

$$f_d(x) = \sum_{(e_1,\ldots,e_d)\in E^d} \prod_{i=1}^{d} \frac{x_{e_i}}{|e_1 \cup \ldots \cup e_i|}.$$

Recall our task was to decide whether

$$\min_{x\in\Delta_m} f_d(x) \overset{?}{=} f_d(x^*),$$

where $x^* = \frac{1}{m}(1,\ldots,1)$ is the barycenter of the simplex Δ_m. This statement is true if f_d is convex over Δ_m. To see this, observe that Lemma 12.1 can easily be extended to the polynomial f_d. So, the tasks shift to proving the convexity of f_d over the simplex.

Conjecture 13.1. *For any integers $n, L, d \geq 2$, the polynomial f_d is convex over the simplex Δ_m.*

Note that if this conjecture holds true, then, via a similar argument to what we used in Lemma 12.1, Conjecutre 11.1 follows. For degree $d = 2$, we have $f_2 = \frac{1}{L}p_2$, and thus we know from Theorem 11.2 that f_2 is convex. In Section 13.1, we prove that Conjecture 13.1 holds for degree $d = 3$ and edge size $L = 2$. In Section 13.2, we provide numerical justification for why we think Conjecture 13.1 holds for more values of n, L, and d.

In what follows, we decompose in the monomial basis the Hessian $H(f_d)$ of f_d into a family of well-structured polynomial matrices Q_γ (see Lemma 13.2). Then we give a recursive reformulation of the matrices Q_γ linking them to the matrices M_γ (from Lemma 12.4) that constitute the Hessian $H(p_d)$ of p_d (see Lemma 13.5). In Section 13.1, we show that the matrices Q_γ are PSD in the case when $d = 3$ and $L = 2$, thereby proving Conjecture 13.1 for this special case.

Understanding the general case ($d > 3$ and $L > 2$) is technically involved. New tools for exploiting the symmetry structure in the matrices Q_γ are required as the Terwilliger algebra does not capture the structure. This goes beyond the scope of this thesis, and we leave it for further research. In recent work, Polak [**131**] has carried out a symmetry reduction, which enables him to show that Conjecture 13.1 holds in the case when $d \leq 8$ and $L = 2$.

13.1. Computing the Hessian of f_d

We begin with expressing the polynomial f_d in the standard monomial basis:

$$f_d(x) = \sum_{\alpha \in \mathbb{N}^m_d} x^\alpha \sum_{\substack{\underline{e}=(e_1,\dots,e_d)\in E^d \\ \alpha(\underline{e})=\alpha}} \prod_{i=1}^{d} \frac{1}{|e_1 \cup \ldots \cup e_i|} = \sum_{\alpha \in \mathbb{N}^m_d} b_\alpha x^\alpha,$$

where we set

$$b_\alpha := \sum_{\substack{\underline{e}=(e_1,\dots,e_d)\in E^d \\ \alpha(\underline{e})=\alpha}} \prod_{i=1}^{d} \frac{1}{|e_1 \cup \ldots \cup e_i|}. \tag{13.1}$$

Lemma 13.2. *The Hessian $H(f_d)$ of the polynomial f_d is given by*

$$H(f_d)(x) = \sum_{\gamma \in \mathbb{N}^m_{d-2}} x^\gamma Q_\gamma, \tag{13.2}$$

where, for each $\gamma \in \mathbb{N}^m_{d-2}$, the symmetric $m \times m$ matrix Q_γ is defined entry-wise by

$$(Q_\gamma)_{ij} := \left\{ \begin{array}{ll} (\gamma_i+1)(\gamma_j+1)b_{\gamma+u_i+u_j} & i \neq j \\ (\gamma_i+1)(\gamma_i+2)b_{\gamma+2u_i} & i = j \end{array} \right. .$$

Proof. Direct verification. □

A recursive reformulation for the coefficients of the polynomial f_d. Fix $\alpha \in \mathbb{N}^m_d$. Looking at the coefficients b_α, we see there are $\frac{d!}{\alpha!}$ distinct tuples $\underline{e}$ such that $\alpha(\underline{e}) = \alpha$. For any such sequence $\underline{e} = (e_{i_1}, \dots, e_{i_d})$ with $i_1, \dots, i_d \in [m]$, $\alpha = \alpha(\underline{e})$ means that, for any $\ell \in [m]$, α_ℓ is the number of occurrences of ℓ within the multiset $\{i_1, \dots, i_d\}$; so $\alpha_\ell \geq 1$ if $\ell \in \{i_1, \dots, i_d\}$ and $\alpha_\ell = 0$ if $\ell \notin \{i_1, \dots, i_d\}$.

For instance, for $\underline{e} = (e_1, e_2, e_3, e_2, e_1)$, $d = 5$, $m = 4$, we have $(i_1, \dots, i_5) = (1, 2, 3, 2, 1)$ and $\alpha(\underline{e}) = (2, 2, 1, 0)$.

Using this fact, we can rewrite b_α to be a summation over $[m]$ as opposed to summing over edge tuples $\underline{e}$.

Lemma 13.3. *For any $\alpha \in \mathbb{N}^m_d$ we have*

$$b_\alpha = c_\alpha \sum_{k \in [m] : \alpha_k \geq 1} b_{\alpha - u_k},$$

where c_α was defined in (12.4).

PROOF. To reformulate b_α we exploit the fact that b_α enjoys some invariance property under permutations of $[d]$, namely

$$\begin{aligned}
b_\alpha &= \sum_{\substack{\underline{e}=(e_{i_1},\dots,e_{i_d})\in E^d:\\ \alpha(\underline{e})=\alpha}} \prod_{k=1}^{d} \frac{1}{|e_{i_1}\cup\ldots\cup e_{i_k}|} \\
&= \frac{1}{d!}\sum_{\sigma\in\mathrm{Sym}(d)} \sum_{\substack{\underline{e}=(e_{i_1},\dots,e_{i_d})\in E^d:\\ \alpha(\underline{e})=\alpha}} \prod_{k=1}^{d} \frac{1}{|e_{i_{\sigma(1)}}\cup\ldots\cup e_{i_{\sigma(k)}}|} \\
&= \frac{1}{d!}\sum_{\substack{\underline{e}=(e_{i_1},\dots,e_{i_d})\in E^d\\ \alpha(\underline{e})=\alpha}} \underbrace{\sum_{\sigma\in\mathrm{Sym}(d)} \prod_{k=1}^{d} \frac{1}{|e_{i_{\sigma(1)}}\cup\ldots\cup e_{i_{\sigma(k)}}|}}_{=:S}. \qquad (13.3)
\end{aligned}$$

Observe that the inner summation S in (13.3) does not depend on the choice of the sequence $\underline{e}$ such that $\alpha(\underline{e})=\alpha$; thus we may consider it fixed, denoted as $(e_{i_1},\dots,e_{i_d})$. Since there are $\frac{d!}{\alpha!}$ possible choices for selecting this sequence, using relation (13.3), we can reformulate b_α as follows:

$$b_\alpha = \frac{1}{d!}\frac{d!}{\alpha!}\sum_{\sigma\in\mathrm{Sym}(d)}\prod_{k=1}^{d}\frac{1}{|e_{i_{\sigma(1)}}\cup\ldots\cup e_{i_{\sigma(k)}}|} = \frac{1}{\alpha!}\sum_{\sigma\in\mathrm{Sym}(d)}\prod_{k=1}^{d}\frac{1}{|e_{i_{\sigma(1)}}\cup\ldots\cup e_{i_{\sigma(k)}}|}.$$

Next, we pull out the factor $\frac{1}{|e_{i_1}\cup\ldots\cup e_{i_d}|}=c_\alpha$ which occurs for $k=d$ and get

$$\begin{aligned}
b_\alpha &= \frac{c_\alpha}{\alpha!}\sum_{r=1}^{d}\sum_{\sigma\in\mathrm{Sym}(d):\sigma(d)=r}\prod_{k=1}^{d-1}\frac{1}{|e_{i_{\sigma(1)}}\cup\ldots\cup e_{i_{\sigma(k)}}|} \\
&= \frac{c_\alpha}{\alpha!}\sum_{r=1}^{d} b_{\alpha-u_{i_r}}(\alpha-u_{i_r})! \\
&= c_\alpha\sum_{r=1}^{d}\frac{b_{\alpha-u_{i_r}}}{\alpha_{i_r}} \\
&\overset{(*)}{=} c_\alpha\sum_{k\in[m]:\alpha_k\geq 1} b_{\alpha-u_k}.
\end{aligned}$$

Here, in the last equality marked (*), we use the fact that α_k of the elements in the multiset $\{i_1,\dots,i_d\}$ are equal to k. □

We now give a recursive reformulation for the matrices Q_γ. Begin by defining a new parameter $\widehat{b}_\alpha := \alpha!\; b_\alpha$, for which we have, via Lemma 13.3, that

$$\widehat{b}_\alpha = \alpha!\; b_\alpha = \alpha!\; c_\alpha\sum_{k:\alpha_k\geq 1} b_{\alpha-u_k} = \alpha!\; c_\alpha\sum_{k:\alpha_k\geq 1}\frac{\widehat{b}_{\alpha-u_k}}{\alpha-u_k!} = c_\alpha\sum_{k:\alpha_k\geq 1}\alpha_k\widehat{b}_{\alpha-u_k}.$$

LEMMA 13.4. *For any $\gamma\in\mathbb{N}^m_{d-2}$ we have $Q_\gamma = \frac{1}{\gamma!}(\widehat{b}_{\gamma+u_i+u_j})_{i,j=1}^m$.*

PROOF. By direct verification, we have, for $i\neq j$ that

$$(Q_\gamma)_{ij} = (\gamma_i+1)(\gamma_j+1)b_{\gamma+u_i+u_j} = \widehat{b}_{\gamma+u_i+u_j}(\gamma_i+1)(\gamma_j+1)/(\gamma+u_i+u_j)! = \widehat{b}_{\gamma+u_i+u_j}/\gamma!$$

and, for $i = j$, we have

$$(Q_\gamma)_{ii} = (\gamma_i + 1)(\gamma_i + 2)b_{\gamma+2u_i} = \widehat{b}_{\gamma+2u_i}(\gamma_i + 1)(\gamma_i + 2)/(\gamma + 2u_i)! = \widehat{b}_{\gamma+2u_i}/\gamma!.$$

□

Lemma 13.5. *For $d \geq 3$ and $\gamma \in \mathbb{N}^m_{d-2}$ we have*

$$Q_\gamma = \underbrace{(c_{\gamma+u_i+u_j})^m_{i,j=1}}_{M_\gamma} \circ \Big(\sum_{k \in [m]: \gamma_k \geq 1} Q_{\gamma-u_k} + \underbrace{\frac{1}{\gamma!}(\widehat{b}_{\gamma+u_i} + \widehat{b}_{\gamma+u_j})^m_{i,j=1}}_{=:R_\gamma} \Big)$$

$$= M_\gamma \circ \Big(\sum_{k \in [m]: \gamma_k \geq 1} Q_{\gamma-u_k} + R_\gamma \Big),$$

where the matrices M_γ were introduced in Lemma 12.4.

Proof. Combining Lemmas 13.3 and 13.5 we obtain

$$\begin{aligned}
(Q_\gamma)_{ij} &= \frac{1}{\gamma!}\widehat{b}_{\gamma+u_i+u_j} = \frac{1}{\gamma!}c_{\gamma+u_i+u_j} \sum_{k:(\gamma+u_i+u_j)_k \geq 1} \widehat{b}_{\gamma+u_i+u_j-u_k}(\gamma + u_i + u_j)_k \\
&= \frac{1}{\gamma!}c_{\gamma+u_i+u_j}\Big(\sum_{k \neq i,j: \gamma_k \geq 1} \widehat{b}_{\gamma+u_i+u_j-u_k}\gamma_k + \widehat{b}_{\gamma+u_j}(\gamma_i + 1) + \widehat{b}_{\gamma+u_j}(\gamma_i + 1)\Big) \\
&= \frac{1}{\gamma!}c_{\gamma+u_i+u_j}\Big(\sum_{k: \gamma_k \geq 1} \widehat{b}_{\gamma-u_k+u_i+u_j}\gamma_k + \widehat{b}_{\gamma+u_i} + \widehat{b}_{\gamma+u_j}\Big) \\
&= c_{\gamma+u_i+u_j}\Big(\sum_{k: \gamma_k \geq 1} \frac{\widehat{b}_{\gamma-u_k+u_i+u_j}}{(\gamma - u_k)!} + \frac{1}{\gamma!}(\widehat{b}_{\gamma+u_i} + \widehat{b}_{\gamma+u_j})\Big) \\
&= c_{\gamma+u_i+u_j}\Big(\sum_{k: \gamma_k \geq 1} (Q_{\gamma-u_k})_{ij} + \frac{1}{\gamma!}(\widehat{b}_{\gamma+u_i} + \widehat{b}_{\gamma+u_j})\Big),
\end{aligned}$$

which shows the claim. □

Showing $Q_\gamma \succeq 0$ $(\gamma \in \mathbb{N}^m_{d-2})$ ***when*** $d = 3$***,*** $L = 2$***.*** In view of Lemma 13.2, it suffices to show that the matrix Q_γ is positive semidefinite for any $\gamma \in \mathbb{N}^m_1$. Up to symmetry, it suffices to show that $Q_\gamma \succeq 0$ for $\gamma = u_1$. In view of Lemma 13.5 we have

$$Q_{u_1} = \underbrace{(c_{u_1+u_i+u_j})^m_{i,j=1}}_{=M_{u_1}} \circ (Q_0 + \underbrace{(\widehat{b}_{u_1+u_i} + \widehat{b}_{u_1+u_j})^m_{i,j=1}}_{=R_{u_1}}).$$

Earlier, in Lemma 12.6 and Section 12.2.3, we showed that $M_{u_1} \succeq 0$ because $M_2 \succeq 0$. Hence, we endeavor now to show that $Q_0 + R_{u_1} \succeq 0$, which will imply that $Q_{u_1} \succeq 0$, and conclude the proof of Theorem 11.3.

Describing the entries of the matrix $Q_0 + R_{u_1}$***.*** By definition, the entries of Q_0 (case $\gamma = 0$) are

$$(Q_0)_{ii} = 2b_{2u_i} = \frac{2}{L}, \quad (Q_0)_{ij} = b_{u_i+u_j} = \frac{2}{|e_i \cup e_j|} \quad \text{for } i \neq j \in [m].$$

Moreover, $\widehat{b}_{2u_1} = 2b_{2u_1} = \frac{2}{L}$ and $\widehat{b}_{u_1+u_i} = b_{u_1+u_i} = \frac{2}{|e_1 \cup e_i|}$ for $i \geq 2$. Using this, we obtain that

$$Q_0 + R_{u_1} = 2 \cdot \Big(\frac{1}{|e_1 \cup e_j|} + \frac{1}{|e_i \cup e_j|} + \frac{1}{|e_1 \cup e_i|} \Big)_{i,j=1}^{m} =: 2B,$$

where we define the matrix B as

$$B := \Big(\frac{1}{|e \cup f|} + \frac{1}{|e_1 \cup e|} + \frac{1}{|e_1 \cup f|} \Big)_{e,f \in E}. \tag{13.4}$$

Proposition 13.6. *For $L = 2$, we have $B \succeq 0$.*

Before proceeding to the proof, let us make a few observations. Note that the matrix B can be decomposed as

$$B = \underbrace{\Big(\frac{1}{|e \cup f|} \Big)_{e,f \in E}}_{=M_0} + \underbrace{\Big(\frac{1}{|e_1 \cup e|} + \frac{1}{|e_1 \cup f|} \Big)_{e,f \in E}}_{=:R}.$$

As we are in the case $L = 2$, the matrix M_0 is the matrix M from (12.1), and is thus positive semidefinite. On the other hand, the matrix R is not positive semidefinite. In fact, R has rank 2 and a negative eigenvalue. One can infer from the results in Section 12.1 that $\lambda_{\min}(M_0) = 1/12$, while one can check that $\lambda_{\min}(R) < -1/12 = -0.0833...$ when $n \geq 6$ (see Table 1 below). Hence in general, one cannot argue that $B \succeq 0$ by simply looking at the smallest eigenvalues of its summands M_0 and R.

On a very high level, we will show positive semidefiniteness of the matrix B by observing that it has a simple block structure, which we can exploit by taking several successive Schur complements; in this way, we obtain well-structured matrices that can be directly shown to be positive semidefinite.

Proof of Proposition 13.6. To fix ideas we let e_1 be the edge $e_1 = \{1, 2\}$ and to simplify notation we set $p = n - 2$ and $q = \binom{n-2}{2}$. Then the index set of B can be partitioned into $\{e_1\} \cup I_1 \cup I_2 \cup I_0$, setting $I_k = \{\{k, i\} : 3 \leq i \leq n\}$ for $k = 1, 2$, and $I_0 = \{\{i, j\} : 3 \leq i < j \leq n\}$. So $|I_1| = |I_2| = p$ and $|I_0| = q$. With respect to this partition, one can verify that the matrix B has the following block-form:

$$B = \begin{array}{c} \\ e_1 \\ I_1 \\ I_2 \\ I_0 \end{array} \begin{array}{c|c|c|c} e_1 & I_1 & I_2 & I_0 \\ \frac{3}{2} & \frac{7}{6}J_{1,p} & \frac{7}{6}J_{1,p} & J_{1,q} \\ \hline \frac{7}{6}J_{p,1} & J_p + \frac{1}{6}I_p & \frac{11}{12}J_p + \frac{1}{12}I_p & \frac{5}{6}J_{p,q} + \frac{1}{12}\Gamma^T \\ \hline \frac{7}{6}J_{p,1} & \frac{11}{12}J_p + \frac{1}{12}I_p & J_p + \frac{1}{6}I_p & \frac{5}{6}J_{p,q} + \frac{1}{12}\Gamma^T \\ \hline J_{q,1} & \frac{5}{6}J_{q,p} + \frac{1}{12}\Gamma & \frac{5}{6}J_{q,p} + \frac{1}{12}\Gamma & M + \frac{1}{2}J_q \end{array}.$$

Here, M is the matrix from (12.1) (replacing n by $p = n - 2$). We have shown in Section 12.1 (see relation (12.2)) that M can be decomposed as

$$M = \frac{1}{12}I_q + \frac{1}{4}J_q + \frac{1}{12}\Gamma\Gamma^T,$$

where $\Gamma = \Gamma_p$ is the $\binom{p}{2} \times p$ matrix whose (f, i)th entry is $|\{i\} \cap f|$. We now proceed to show that the matrix B is positive semidefinite. Note that its lower right diagonal block indexed by the set I_0 is positive semidefinite (since $M \succeq 0$). Our strategy is now to 'eliminate' the three borders indexed by the sets $\{e_1\}$, I_1, and I_2 successively, one by one, by taking Schur complements, until reaching a final matrix (indexed by I_0) whose positive semidefiniteness can be seen directly. To do the Schur complement operations, we will need to invert matrices of the form $aI + bJ$.

LEMMA 13.7. *For $a, b \in \mathbb{R}$ such that $a+pb \neq 0$, the matrix aI_p+bJ_p is nonsingular with inverse*

$$(aI_p + bJ_p)^{-1} = \frac{1}{a}\Big(I_p - \frac{b}{pb+a}\Big)J_p.$$

We now eliminate the three borders of B indexed by $\{e_1\}$, I_1, and I_2 by taking successive Schur complements with respect to the upper left corner.

The first Schur complement. We take a first Schur complement with respect to the upper left corner of B (indexed by e_1). We call $\widetilde{B}_1$ the resulting matrix, which reads

$$\widetilde{B}_1 := \left(\begin{array}{c|c|c} J_p + \frac{1}{6}I_p & \frac{11}{12}J_p + \frac{1}{12}I_p & \frac{5}{6}J_{p,q} + \frac{1}{12}\Gamma^T \\ \hline \frac{11}{12}J_p + \frac{1}{12}I_p & J_p + \frac{1}{6}I_p & \frac{5}{6}J_{p,q} + \frac{1}{12}\Gamma^T \\ \hline \frac{5}{6}J_{q,p} + \frac{1}{12}\Gamma & \frac{5}{6}J_{q,p} + \frac{1}{12}\Gamma & \frac{1}{12}I_q + \frac{1}{12}\Gamma\Gamma^T + \frac{3}{4}J_q \end{array}\right)$$

$$-\frac{2}{3}\begin{pmatrix} \frac{7}{6}J_{p,1} \\ \frac{7}{6}J_{p,1} \\ J_{q,1} \end{pmatrix}\begin{pmatrix} \frac{7}{6}J_{1,p} & \frac{7}{6}J_{1,p} & J_{1,q} \end{pmatrix}$$

$$= \left(\begin{array}{c|c|c} \frac{5}{54}J_p + \frac{1}{6}I_p & \frac{1}{108}J_p + \frac{1}{12}I_p & \frac{1}{18}J_{p,q} + \frac{1}{12}\Gamma^T \\ \hline \frac{1}{108}J_p + \frac{1}{12}I_p & \frac{5}{54}J_p + \frac{1}{6}I_p & \frac{1}{18}J_{p,q} + \frac{1}{12}\Gamma^T \\ \hline \frac{1}{18}J_{q,p} + \frac{1}{12}\Gamma & \frac{1}{18}J_{q,p} + \frac{1}{12}\Gamma & \frac{1}{12}I_q + \frac{1}{12}\Gamma\Gamma^T + \frac{1}{12}J_q \end{array}\right).$$

Setting $B_1 = 6\widetilde{B}_1$, we obtain $B \succeq 0 \Longleftrightarrow \widetilde{B}_1 \succeq 0 \Longleftrightarrow B_1 \succeq 0$, where

$$B_1 = \left(\begin{array}{c|c|c} \frac{5}{9}J_p + I_p & \frac{1}{18}J_p + \frac{1}{2}I_p & \frac{1}{3}J_{p,q} + \frac{1}{2}\Gamma^T \\ \hline \frac{1}{18}J_p + \frac{1}{2}I_p & \frac{5}{9}J_p + I_p & \frac{1}{3}J_{p,q} + \frac{1}{2}\Gamma^T \\ \hline \frac{1}{3}J_{q,p} + \frac{1}{2}\Gamma & \frac{1}{3}J_{q,p} + \frac{1}{2}\Gamma & \frac{1}{2}I_q + \frac{1}{2}\Gamma\Gamma^T + \frac{1}{2}J_q \end{array}\right).$$

The second Schur complement. We now take the Schur complement with respect to the upper left corner of B_1 (indexed by I_1), where we use Lemma 13.7 to invert it:

$$(I_p + 5/9J_p)^{-1} = I_p - 5/(5p+9)J_p.$$

After taking this Schur complement the resulting matrix $\widetilde{B}_2$ reads:

$$\widetilde{B}_2 = \left(\begin{array}{c|c} \frac{5}{9}J_p + I_q & \frac{1}{3}J_{p,q} + \frac{1}{2}\Gamma^T \\ \hline \frac{1}{3}J_{q,p} + \frac{1}{2}\Gamma & \frac{1}{2}I_q + \frac{1}{2}\Gamma\Gamma^T + \frac{1}{2}J_q \end{array}\right)$$

$$- \begin{pmatrix} \frac{1}{18}J_p + \frac{1}{2}I_p \\ \frac{1}{3}J_{q,p} + \frac{1}{2}\Gamma \end{pmatrix} \Big(I_p - \frac{5}{(5p+9)}J_p\Big) \Big(\frac{1}{18}J_p + \frac{1}{2}I_p \quad \frac{1}{3}J_{p,q} + \frac{1}{2}\Gamma^T\Big)$$

$$= \left(\begin{array}{c|c} \frac{3}{4}I_p + \frac{11p+23}{4(5p+9)}J_p & \frac{1}{4}\Gamma^T + \frac{3p+7}{2(5p+9)}J_{p,q} \\ \hline \frac{1}{4}\Gamma + \frac{3p+7}{2(5p+9)}J_{q,p} & \frac{1}{2}I_q + \frac{1}{4}\Gamma\Gamma^T + \frac{3p+7}{2(5p+9)}J_q \end{array}\right).$$

Setting $B_2 = 4\widetilde{B}_2$ we obtain $B \succeq 0 \iff B_1 \succeq 0 \iff B_2 \succeq 0$, where

$$B_2 = \left(\begin{array}{c|c} 3I_p + \frac{11p+23}{5p+9}J_p & \Gamma^T + \frac{2(3p+7)}{5p+9}J_{p,q} \\ \hline \Gamma + \frac{2(3p+7)}{5p+9}J_{q,p} & 2I_q + \Gamma\Gamma^T + \frac{2(3p+7)}{5p+9}J_q \end{array}\right).$$

The third and final Schur complement. Inverting the top left block of B_2 via Lemma 13.7 gives

$$\Big(3I_p + \frac{11p+23}{5p+9}J_p\Big)^{-1} = \frac{1}{3}I_p - \frac{(11p+23)}{3(11p^2+38p+27)}J_p.$$

Taking the third and final Schur complement with respect to this block in B_2 we get the matrix

$$B_3 := 2I_q + \Gamma\Gamma^T + \frac{2(3p+7)}{5p+9}J_q$$
$$-\Big(\Gamma^T + \frac{2(3p+7)}{5p+9}J_{q,p}\Big)\Big(\frac{1}{3}I_p - \frac{(11p+23)}{3(11p^2+38p+27)}J_p\Big)\Big(\Gamma^T + \frac{2(3p+7)}{5p+9}J_{p,q}\Big)$$
$$= 2I_q + \frac{2}{3}\Gamma\Gamma^T + \frac{2(9p+25)}{3(11p+27)}J_q.$$

It is now clear that $B_3 \succeq 0$. This implies that $B_2 \succeq 0$ and thus $B \succeq 0$, which concludes the proof of Proposition 13.6.

Why the proof of Proposition 13.6 is hard to extend to $L \geq 3$. The biggest hurdle lies in the richness of the possible intersections between edges of large sizes. More concretely, recall that the $(e, f)^{\text{th}}$ entry of the matrix B in (13.4) depends on $|e \cup f|$, $|e \cup e_1|$ and $|f \cup e_1|$. So, one has to take into account how the two edges e, f pairwise interact within e_1 and outside of it, which becomes technically involved when $|e_1| = L$ is large. The matrix B has an increasingly involved block structure when L grows. In addition, some of the blocks in B may have a form that requires an additional symmetry reduction to become amenable.

13.2. Numerical justification for convexity of f_d

We have carried out some numerical experiments for a range of values of d, L, n and verified that the matrices Q_γ are positive semidefinite for all $\gamma \in \mathbb{N}^n_{d-2}$ in these cases. Hence, for these values, the polynomial f_d is convex, and Conjecture 13.1 holds. Recall from Lemma 13.5 that the matrix Q_γ can be decomposed as

$$Q_\gamma = M_\gamma \circ \Big(\underbrace{\sum_{k\in[m]:\gamma_k\geq 1} Q_{\gamma-u_k} + R_\gamma}_{=:B_\gamma} \Big) = M_\gamma \circ (B_\gamma + R_\gamma).$$

By the results in Section 12.2, we already know that the matrix M_γ is positive semidefinite. Hence, it now suffices to show that the matrix $B_\gamma + R_\gamma$ is positive semidefinite for each $\gamma \in \mathbb{N}^n_{d-2}$. We did this in the previous section for the case $d = 3$ (and $L = 2$). We have computed the minimum eigenvalues of the matrices Q_γ, B_γ, and R_γ for different values of n, d and L and give this information for the case $L = 2$ in Table 1 below (for $d = 3$). In Appendix A of [**27**], extensive tables are provided for $d \geq 4$.

d	L	n	γ	$\lambda_{min}(Q_\gamma)$	$\lambda_{min}(B_\gamma)$	$\lambda_{min}(R_\gamma)$
3	2	3	[[1, 2]]	0.0556	0.1667	-0.0236
3	2	4	[[1, 2]]	0.0347	0.0833	-0.0478
3	2	5	[[1, 2]]	0.0347	0.0833	-0.0729
3	2	6	[[1, 2]]	0.0347	0.0833	-0.0987
3	2	7	[[1, 2]]	0.0347	0.0833	-0.1249
3	2	8	[[1, 2]]	0.0347	0.0833	-0.1514

Table 1. Case $d = 3, L = 2$

For recent progress on this problem, we refer to Polak [**131**], who proved that all the matrices Q_γ are positive semidefinite in the case $d \leq 8$ and $L = 2$. One of the difficulties is that one needs to enumerate the distinct patterns for $\gamma \in \mathbb{N}^m_{d-2}$, i.e., the number of multigraphs with $d - 2$ edges. As mentioned earlier in Example 11.4, this number is given by the OEIS sequence A050535 [**1**], and it grows quickly with d.

Discussion

Some background on symmetry. Symmetry is a widely used ingredient in optimization, in particular in semidefinite optimization and algebraic questions involving polynomials. We mention a few landmark examples as background information. Symmetry can be used to formulate equivalent, more compact reformulations for semidefinite programs. The underlying mathematical fact is Artin-Wedderburn theory, which shows that matrix $*$-algebras can be block-diagonalized (see Theorem 11.6).

A well-known early example is the linear programming reformulation from [**143**] for the Lovász theta number of Hamming graphs, showing the link to the Delsarte bound and Bose-Mesner algebras of Hamming schemes [**49, 50**]. Symmetry is used more generally to give tractable reformulations for the semidefinite bounds arising from the next levels of Lasserre's hierarchy in [**144**] (which gives the explicit block-diagonalization for the Terwilliger algebra of Hamming schemes, see Theorem 11.7) and, e.g., in [**75**], [**74**], [**109**], [**114**]. For more examples and a broad exposition about symmetry in semidefinite programming, we refer, e.g., to [**10, 44**] and further references therein. Symmetry is also a crucial ingredient in the study of algebraic questions about polynomials, like representations in terms of sums of squares and in polynomial optimization. We refer to [**71**] for a broad exposition and, e.g., to [**138**] (for compact reformulations of Lasserre relaxations of symmetric polynomial optimization problems), [**137**] (for methods to reduce the number of variables in programs involving symmetric polynomials), and the recent works [**133, 134**] (which consider symmetric polynomials with variables indexed by the k-subsets hypercube (as in our case) and uncover links with the theory of flag algebras by Razborov [**135**]).

Current status of Conjecture 13.1. As of the time of writing this thesis, Conjecture 13.1 has not been proven to hold for the general case: $d \geq 9$ and $L \geq 2$.

Bibliography

[1] The On-Line Encyclopedia of Integer Sequences. 2021.

[2] M. R. Abdalmoaty, D. Henrion, and L. Rodrigues. Measures and LMIs for optimal control of piecewise-affine systems. In *2013 European Control Conference (ECC)*, pages 3173–3178, 2013.

[3] J. Agler, W. Helton, S. McCullough, and L. Rodman. Positive semidefinite matrices with a given sparsity pattern. *Linear Algebra and its Applications*, 107:101–149, 1988.

[4] N. I. Akhiezer. *The Classical Moment Problem.* Hafner Publishing Company, New York, 2019/04/01 edition, 1965.

[5] E. D. Andersen and K. D. Andersen. The MOSEK interior point optimizer for linear programming: An implementation of the homogeneous algorithm. In H. Frenk, K. Roos, T. Terlaky, and S. Zhang, editors, *High Performance Optimization*, volume 33, pages 197–232. Springer US, Boston, MA, 2000.

[6] B. Aracıoğlu, F. Demircan Keskin, and H. Uçak. Mean–Variance–Skewness–Kurtosis approach to portfolio optimization: An application in Istanbul stock exchange. *Ege Academic Review*, 11:9–17, 2011.

[7] S. Arnborg, D. G. Corneil, and A. Proskurowski. Complexity of finding embeddings in a k-tree. *SIAM Journal on Algebraic Discrete Methods*, 8(2):277–284, 1987.

[8] G. Averkov. Optimal size of linear matrix inequalities in semidefinite approaches to polynomial optimization. *SIAM Journal on Applied Algebra and Geometry*, 3(1):128–151, 2019.

[9] B. Ghaddar, J. Marecek, and M. Mevissen. Optimal Power Flow as a Polynomial Optimization Problem. *IEEE Transactions on Power Systems*, 31(1):539–546, 2016.

[10] C. Bachoc, D. C. Gijswijt, A. Schrijver, and F. Vallentin. Invariant semidefinite programs. In M. F. Anjos and J. B. Lasserre, editors, *Handbook on Semidefinite, Conic and Polynomial Optimization*, pages 219–269. Springer US, New York, 2012.

[11] F. Barioli. Completely positive matrices with a book-graph. *Linear Algebra and its Applications*, 277(1):11–31, 1998.

[12] F. Barioli and A. Berman. The maximal cp-rank of rank k completely positive matrices. *Linear Algebra and its Applications*, 363:17–33, 2003.

[13] G. Barker, L. Eifler, and T. Kezlan. A non-commutative spectral theorem. *Linear Algebra and its Applications*, 20(2):95–100, 1978.

[14] L. B. Beasley and T. J. Laffey. Real rank versus nonnegative rank. *Linear Algebra and its Applications*, 431(12):2330–2335, 2009.

[15] A. Beck. *First-Order Methods in Optimization.* Society for Industrial and Applied Mathematics, Philadelphia, 2017.

[16] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[17] A. Ben-Tal and A. Nemirovski. *Lectures on Modern Convex Optimization.* Society for Industrial and Applied Mathematics, Philadelphia, 2001.

[18] A. Berman and N. Shaked-Monderer. *Completely Positive Matrices.* WORLD SCIENTIFIC, 2003.

[19] M. J. Best. *Portfolio Optimization.* Chapman and Hall, New York, 1 edition, 2010.

[20] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah. Julia: A fresh approach to numerical computing. *SIAM Review*, 59(1):65–98, 2017.

[21] H. L. Bodlaender and A. M. Koster. Treewidth computations I. Upper bounds. *Information and Computation*, 208(3):259–275, 2010.

[22] I. M. Bomze, W. Schachinger, and R. Ullrich. From seven to eleven: Completely positive matrices with high cp-rank. *Linear Algebra and its Applications*, 459:208–221, 2014.

[23] I. M. Bomze, W. Schachinger, and R. Ullrich. New lower bounds and asymptotics for the cp-rank. *SIAM Journal on Matrix Analysis and Applications*, 36(1):20–37, 2015.

[24] S. Boyd and L. Vandenberghe. *Convex Optimization.* Cambridge University Press, Cambridge, 2004.

[25] G. Braun, S. Fiorini, S. Pokutta, and D. Steurer. Approximation limits of linear programs (beyond hierarchies). *Mathematics of Operations Research*, 40(3):756–772, 2015.

[26] C. Bron and J. Kerbosch. Algorithm 457: Finding all cliques of an undirected graph. *Communications of The Acm*, 16(9):575–577, 1973.

[27] D. Brosch, M. Laurent, and A. Steenkamp. Optimizing hypergraph-based polynomials modeling job-occupancy in queuing with redundancy scheduling. *SIAM Journal on Optimization*, 31(3):2227–2254, 2021.

[28] S. Burer. On the copositive representation of binary and continuous nonconvex quadratic programs. *Mathematical Programming*, 120(2):479–495, 2009.

[29] E. Cardinaels, S. Borst, and J. S. van Leeuwaarden. Power-of-two sampling in redundancy systems: The impact of assignment constraints. *Operations Research Letters*, 50(6):699–706, 2022.

[30] V. Chandrasekaran and P. Shah. Relative entropy relaxations for signomial optimization. *SIAM Journal on Optimization*, 26(2):1147–1173, 2016.

[31] L. Chen and D. Ž. Đoković. Qubit-qudit states with positive partial transpose. *Physical Review A*, 86(6):062332, 2012.

[32] L. Chen and D. Ž. Đoković. Dimensions, lengths, and separability in finite-dimensional quantum systems. *Journal of Mathematical Physics*, 54(2):022201, 2013.

[33] T. Chen, J. B. Lasserre, V. Magron, and E. Pauwels. Semialgebraic optimization for lipschitz constants of ReLU networks. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 19189–19200. Curran Associates, Inc., 2020.

[34] T. Chen, J. B. Lasserre, V. Magron, and E. Pauwels. Semialgebraic representation of monotone deep equilibrium models and applications to certification. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 27146–27159. Curran Associates, Inc., 2021.

[35] M.-D. Choi. Positive linear maps. In *Operator Algebras and Applications*, volume 38 of *Proc. Sympos. Pure Math.*, pages 583–590. 1982.

[36] A. Cichocki, R. Zdunek, A. H. Phan, and S.-I. Amari. *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-Way Data Analysis and Blind Source Separation.* Wiley Publishing, 2009.

[37] R. W. Cottle, J.-S. Pang, and R. E. Stone. *The Linear Complementarity Problem.* Society for Industrial and Applied Mathematics, 2009.

[38] X. Cui, X. Sun, S. Zhu, R. Jiang, and D. Li. Portfolio optimization with nonparametric value at risk: A block coordinate descent method. *INFORMS Journal on Computing*, 30(3):454–471, 2018.

[39] R. E. Curto and L. A. Fialkow. *Solution of the Truncated Complex Moment Problem for Flat Data*, volume 119. American Mathematical Society, 1996.

[40] R. E. Curto and L. A. Fialkow. The truncated complex K-moment problem. *Transactions of the American Mathematical Society*, 352(6):2825–2855, 2000.

[41] D. Henrion and J.-B. Lasserre. Convergent relaxations of polynomial matrix inequalities and static output feedback. *IEEE Transactions on Automatic Control*, 51(2):192–202, 2006.

[42] E. de Klerk and M. Laurent. A survey of semidefinite programming approaches to the generalized problem of moments and their error analysis. In C. Araujo, G. Benkart, C. E. Praeger, and B. Tanbay, editors, *World Women in Mathematics 2018: Proceedings of the First World Meeting for Women in Mathematics (WM)*, pages 17–56. Springer International Publishing, Cham, 2019.

[43] E. de Klerk and D. Pasechnik. Approximation of the stability number of a graph via copositive programming. *SIAM Journal on Optimization*, 12(4):875–892, 2002.

[44] E. de Klerk, D. V. Pasechnik, and A. Schrijver. Reduction of symmetric semidefinite programs using the regular *-representation. *Mathematical Programming*, 109(2):613–624, 2007.

[45] E. de Klerk and F. Vallentin. On the turing model complexity of interior point methods for semidefinite programming. *SIAM J. on Optimization*, 26(3):1944–1961, 2016.

[46] G. de las Cuevas, T. Drescher, and T. Netzer. Separability for mixed states with operator Schmidt rank two. *Quantum*, 3:203, 2019.

[47] G. de las Cuevas and T. Netzer. Mixed states in one spatial dimension: Decompositions and correspondence with nonnegative matrices. *Journal of Mathematical Physics*, 2020.

[48] L. T. DeCarlo. On the meaning and use of kurtosis. *Psychological Methods*, 2:292–307, 1997.

[49] P. Delsarte. An algebraic approach to the association schemes of coding theory. Phiips 602 Research Reports Supplements 10, Philips Research Laboratories, 1973.

[50] P. Delsarte and V. I. Levenshtein. Association schemes and coding theory. *IEEE Trans. Inf. Theor.*, 44(6):2477–2504, 2006.

[51] V. DeMiguel, L. Garlappi, and R. Uppal. Optimal versus naive diversification: How inefficient is the 1/N portfolio strategy? *The Review of Financial Studies*, 22(5):1915–1953, 2009.

[52] P. J. C. Dickinson and M. Dür. Linear-time complete positivity detection and decomposition of sparse matrices. *SIAM Journal on Matrix Analysis and Applications*, 33(3):701–720, 2012.

[53] P. J. C. Dickinson and L. Gijben. On the computational complexity of membership problems for the completely positive cone and its dual. *Computational Optimization and Applications*, 57:403–415, 2014.

[54] R. Diestel. *Graph Theory.* Springer Berlin, Heidelberg, 2017.

[55] D. P. Divincenzo, B. M. Terhal, and Ashish V. Thapliyal. Optimal decompositions of barely separable states. *Journal of Modern Optics*, 47(2-3):377–385, 2000.

[56] D. P. Doane and L. E. Seward. Measuring skewness: A forgotten statistic? *Journal of Statistics Education*, 19(2):1–18, 2011.

[57] M. Dressler, J. Nie, and Z. Yang. Separability of Hermitian tensors and PSD decompositions. *Linear and Multilinear Algebra*, 70(21):6581–6608, 2022.

[58] J. H. Drew, C. R. Johnson, and R. Loewy. Completely positive matrices associated with M-matrices. *Linear and Multilinear Algebra*, 37(4):303–310, 1994.

[59] I. Dunning, J. Huchette, and M. Lubin. JuMP: A modeling language for mathematical optimization. *Siam Review*, 59:295–320, 2017.

[60] Editors of Encyclopaedia Britannica. S&P 500. *Encyclopedia Britannica*, 2023, February 10.

[61] M. Ehrgott. *Multicriteria Optimization.* Springer Berlin, Heidelberg, 2 edition, 2005.

[62] A. Einstein, B. Podolsky, and N. Rosen. Can quantum-mechanical description of physical reality be considered complete? *Physical Review*, 47(10):777–780, 1935.

[63] P. Erdös, A. W. Goodman, and L. Pósa. The representation of a graph by set intersections. *Canadian Journal of Mathematics*, 18:106–112, 1966.

[64] H. Fawzi. The set of separable states has no finite semidefinite representation except in dimension 3×2. *Communications in Mathematical Physics*, 386(3):1319–1335, 2021.

[65] H. Fawzi, J. Gouveia, P. A. Parrilo, R. Z. Robinson, and R. R. Thomas. Positive semidefinite rank. *Mathematical Programming*, 153(1):133–177, 2015.

[66] H. Fawzi, J. Gouveia, P. A. Parrilo, J. Saunderson, and R. R. Thomas. Lifting for simplicity: Concise descriptions of convex sets. *SIAM Review*, 64(4):866–918, 2022.

[67] H. Fawzi and P. A. Parrilo. Self-scaled bounds for atomic cone ranks: Applications to nonnegative rank and cp-rank. *Mathematical Programming*, 158(1):417–465, 2016.

[68] S. Fiorini, S. Massar, S. Pokutta, H. R. Tiwary, and R. de Wolf. Exponential lower bounds for polytopes in combinatorial optimization. *Journal of The ACM*, 62(2), 2015.

[69] K. Gardner, S. Zbarsky, S. Doroudi, M. Harchol-Balter, E. Hyytiä, and A. Scheller-Wolf. Queueing with redundant requests: Exact analysis. *Queueing Systems*, 83(3-4):227–259, 2016.

[70] M. Garstka, M. Cannon, and P. Goulart. A clique graph based merging strategy for decomposable SDPs. *21st IFAC World Congress*, 53(2):7355–7361, 2020.

[71] K. Gatermann and P. A. Parrilo. Symmetry groups, semidefinite programs, and sums of squares. *Journal of Pure and Applied Algebra*, 192(1):95–128, 2004.

[72] S. Gharibian. Strong NP-hardness of the quantum separability problem. *Quantum Information & Computation*, 10:343–360, 2010.

[73] D. Gijswijt. *Matrix Algebras and Semidefinite Programming Techniques for Codes*. PhD thesis, 2005.

[74] D. Gijswijt, H. D. Mittelmann, and A. Schrijver. Semidefinite code bounds based on quadruple distances. *IEEE Transactions on Information Theory*, 58:2697–2705, 2010.

[75] D. Gijswijt, L. Schrijver, and H. Tanaka. New upper bounds for nonbinary codes based on the Terwilliger algebra and semidefinite programming. *Journal of Combinatorial Theory - Series A*, 113(8):1719–1731, 2006.

[76] N. Gillis. *Nonnegative Matrix Factorization*. Society for Industrial and Applied Mathematics, Philadelphia, 2020.

[77] N. Gillis and F. Glineur. On the geometric interpretation of the nonnegative rank. *Linear Algebra and its Applications*, 437(11):2685–2712, 2012.

[78] G. H. Golub and C. F. van Loan. *Matrix Computations*. The Johns Hopkins University Press, New York, 3 edition, 1996.

[79] G. Gonçalves, P. Wanke, and Y. Tan. A higher order portfolio optimization model incorporating information entropy. *Intelligent Systems with Applications*, 15:200101, 2022.

[80] S. Gribling, D. de Laat, and M. Laurent. Lower bounds on matrix factorization ranks via noncommutative polynomial optimization. *Foundations of Computational Mathematics*, 19(5):1013–1070, 2019.

[81] S. Gribling, M. Laurent, and A. Steenkamp. Bounding the separable rank via polynomial optimization. *Linear Algebra and its Applications*, 648:1–55, 2022.

[82] L. Gurvits. Classical deterministic complexity of Edmonds' problem and quantum entanglement. In *Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing*, STOC '03, pages 10–19, New York, 2003.

[83] M. Hall. *Combinatorial Theory*. John Wiley & Sons, 1988.

[84] D. Henrion, M. Korda, and J. B. Lasserre. *The Moment-SOS Hierarchy*, volume 4 of *Series on Optimization and Its Applications*. World Scientific (Europe), 2020.

[85] D. Henrion and J.-B. Lasserre. Detecting global optimality and extracting solutions in GloptiPoly. In D. Henrion and A. Garulli, editors, *Positive Polynomials in Control*, pages 293–310. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005.

[86] M. Horodecki, P. Horodecki, and R. Horodecki. Separability of mixed states: Necessary and sufficient conditions. *Physics Letters A*, 223(1):1–8, 1996.

[87] P. Horodecki. Separability criterion and inseparable mixed states with positive partial transposition. *Physics Letters A*, 232(5):333–339, 1997.

[88] S. Iliman and T. De Wolff. Amoebas, nonnegative polynomials and sums of squares supported on circuits. *Research in the Mathematical Sciences*, 3(1):1–35, 2016.

[89] T. M. Inc. MATLAB version: 9.13.0 (R2022b). The MathWorks Inc., 2022.

[90] J. Lofberg. YALMIP : A toolbox for modeling and optimization in MATLAB. In *2004 IEEE International Conference on Robotics and Automation (IEEE Cat. No.04CH37508)*, pages 284–289, 2004.

[91] C. Josz and D. K. Molzahn. Lasserre hierarchy for large scale polynomial optimization in real and complex variables. *SIAM Journal on Optimization*, 28(2):1017–1048, 2018.

[92] E. Jurczenko, B. Maillet, and P. Merlin. Hedge fund portfolio selection with higher-order moments: A nonparametric mean-variance-skewness-kurtosis efficient frontier. In *Multi-moment Asset Allocation and Pricing Models*, pages 51–66. 2012.

[93] O. Kallenberg. *Foundations of Modern Probability*. Probability Theory and Stochastic Modelling. Springer Cham, 3 edition, 2021.

[94] P.-M. Kleniati, P. Parpas, and B. Rustem. Partitioning procedure for polynomial optimization. *Journal of Global Optimization*, 48(4):549–567, 2010.

[95] P.-M. Kleniati and B. Rustem. Portfolio decisions with higher order moments. Working Papers 021, COMISEF, 2009.

[96] I. Klep, V. Magron, and J. Povh. Sparse noncommutative polynomial optimization. *Mathematical Programming*, pages 1–41, 2021.

[97] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009.

[98] M. Korda. Stability and performance verification of dynamical systems controlled by neural networks: Algorithms and complexity. *IEEE Control Systems Letters*, 6:3265–3270, 2021.

[99] M. Korda and C. N. Jones. Stability and performance verification of optimization-based controllers. *Automatica*, 78:34–45, 2017.

[100] M. Korda, M. Laurent, V. Magron, and A. Steenkamp. Exploiting ideal-sparsity in the generalized moment problem with application to matrix factorization ranks. *Mathematical Programming*, 2023.

[101] A. Kraus and R. H. Litzenberger. Skewness Preference and the Valuation of Risk Assets. *The Journal of Finance*, 31(4):1085–1100, 1976.

[102] A. Kroó. On Markov Inequality for Multivariate Polynomials. In *Proceedings of the 11th Conference on Approximation Theory*, Modern Methods in Mathematics, Gatlinburg, 2005. Nashboro Press.

[103] K. Lai, L. Yu, and S. Wang. Mean-variance-skewness-kurtosis-based portfolio optimization. *First International Multi-Symposiums on Computer and Computational Sciences (IMSCCS'06)*, 2:292–297, 2006.

[104] J. B. Lasserre. Convergent SDP-relaxations in polynomial optimization with sparsity. *SIAM Journal on Optimization*, 17(3):822–843, 2006.

[105] J. B. Lasserre. A semidefinite programming approach to the generalized problem of moments. *Mathematical Programming*, 112(1):65–92, 2008.

[106] J. B. Lasserre. *Moments, Positive Polynomials and Their Applications*, volume 1 of *Series on Optimization and Its Applications*. Imperial College Press, 2009.

[107] J. B. Lasserre. *An Introduction to Polynomial and Semi-Algebraic Optimization*. Cambridge Texts in Applied Mathematics. Cambridge University Press, Cambridge, 2015.

[108] J. B. Lasserre and Y. Emin. Semidefinite relaxations for Lebesgue and Gaussian measures of unions of basic semialgebraic sets. *Mathematics of Operations Research*, 44(4):1477–1493, 2019.

[109] M. Laurent. Strengthened semidefinite programming bounds for codes. *Mathematical Programming*, 109(2):239–261, 2007.

[110] M. Laurent. Sums of squares, moment matrices and optimization over polynomials. In M. Putinar and S. Sullivant, editors, *Emerging Applications of Algebraic Geometry*, pages 157–270. Springer New York, New York, 2009.

[111] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, 1999.

[112] X. Li and P. Zhang. High order portfolio optimization problem with transaction costs. *Modern Economy*, 10(6):1507–1525, 2019.

[113] Y. Li and G. Ni. Separability discrimination and decomposition of m-partite quantum mixed states. *Physical Review A*, 102(1):012402, 2020.

[114] B. Litjens, S. Polak, and L. Schrijver. Semidefinite bounds for nonbinary codes based on quadruples. *Designs, Codes and Cryptography*, 84:87–100, 2017.

[115] V. Magron, M. Forets, and D. Henrion. Semidefinite approximations of invariant measures for polynomial systems. *Discrete and Continuous Dynamical Systems - B*, 24(12):6745–6770, 2019.

[116] V. Magron and J. Wang. Sparse polynomial optimization: Theory and practice. *ArXiv*, abs/2208.11158, 2022.

[117] D. Maringer and P. Parpas. Global optimization of higher order moments in portfolio selection. *Journal of Global Optimization*, 43(2):219–230, 2009.

[118] H. Markowitz. Portfolio Selection. *The Journal of Finance*, 7(1):77–91, 1952.

[119] R. W. Melicher and E. Norton. *Introduction to Finance: Markets, Investments, and Financial Management*. John Wiley & Sons, Inc, 16 edition, 2016.

[120] M. Mhiri and J.-L. Prigent. International portfolio optimization with higher moments. *International journal of economics and finance*, 2:157–169, 2010.

[121] C. Moreira Costa, D. Kreber, and M. Schmidt. An alternating method for cardinality-constrained optimization: A computational study for the best subset selection and sparse portfolio problems. *INFORMS Journal on Computing*, 34(6):2968–2988, 2022.

[122] 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.

[123] R. Murray, V. Chandrasekaran, and A. Wierman. Signomial and polynomial optimization via relative entropy and partial dualization. *Mathematical Programming Computation*, 13(2):257–295, 2021.

[124] M. Nakata. A numerical evaluation of highly accurate multiple-precision arithmetic version of semidefinite programming solver: SDPA-GMP, -QD and -DD. In *2010 IEEE International Symposium on Computer-Aided Control System Design*, pages 29–34, 2010.

[125] Y. Nesterov and A. Nemirovskii. *Interior-Point Polynomial Algorithms in Convex Programming*. Society for Industrial and Applied Mathematics, 1994.

[126] J. Nie. The $\mathcal{A}$-truncated K-moment problem. *Foundations of Computational Mathematics*, 14(6):1243–1276, 2014.

[127] J. Nie. Symmetric tensor nuclear norms. *SIAM Journal on Applied Algebra and Geometry*, 1(1):599–625, 2017.

[128] J. Nie and X. Zhang. Positive maps and separable matrices. *SIAM Journal on Optimization*, 26(2):1236–1256, 2016.

[129] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, Cambridge, 2010.

[130] A. Peres. Separability Criterion for Density Matrices. *Physical Review Letters*, 77(8):1413–1415, 1996.

[131] S. C. Polak. Symmetry reduction to optimize a graph-based polynomial from queueing theory. *SIAM Journal on Applied Algebra and Geometry*, 6(2):243–266, 2022.

[132] M. Putinar. Positive Polynomials on Compact Semi-algebraic Sets. *Indiana University Mathematics Journal*, 42(3):969–984, 1993.

[133] A. Raymond, J. Saunderson, M. Singh, and R. R. Thomas. Symmetric sums of squares over K-subset hypercubes. *Mathematical Programming*, 167(2):315–354, 2018.

[134] A. Raymond, M. Singh, and R. R. Thomas. Symmetry in Turán sums of squares polynomials from flag algebras. *Algebraic Combinatorics*, 1(2):249–274, 2018.

[135] A. A. Razborov. Flag algebras. *Journal of Symbolic Logic*, 72(4):1239–1282, 2007.

[136] B. Reznick. Extremal PSD forms with few terms. *Duke Mathematical Journal*, 45(2):363–374, 1978.

[137] C. Riener. On the degree and half-degree principle for symmetric polynomials. *Journal of Pure and Applied Algebra*, 216(4):850–856, 2012.

[138] C. Riener, T. Theobald, L. J. Andrén, and J. B. Lasserre. Exploiting symmetries in SDP-relaxations for polynomial optimization. *Mathematics of Operations Research*, 38(1):122–141, 2013.

[139] S. Kullback and R. A. Leibler. On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951.

[140] L. Salce and P. Zanardo. Completely positive matrices and positivity of least squares solutions. *Linear Algebra and its Applications*, 178:201–216, 1993.

[141] P. A. Samuelson. The fundamental approximation theorem of portfolio analysis in terms of means, variances and higher moments. In W. Ziemba and R. Vickson, editors, *Stochastic Optimization Models in Finance*, pages 215–220. Academic Press, 1975.

[142] K. Schmüdgen. The K-moment problem for compact semi-algebraic sets. *Mathematische Annalen*, 289(1):203–206, 1991.

[143] Schrijver, A. A comparison of the Delsarte and Lovász bounds. *IEEE Transactions on Information Theory*, 25(4):425–429, 1979.

[144] Schrijver, A. New code upper bounds from the Terwilliger algebra and semidefinite programming. *IEEE Transactions on Information Theory*, 51(8):2859–2866, 2005.

[145] N. Shaked-Monderer and A. Berman. *Copositive and Completely Positive Matrices*. WORLD SCIENTIFIC, 2019.

[146] N. Shaked-Monderer, I. M. Bomze, F. Jarre, and W. Schachinger. On the cp-rank and minimal cp factorizations of a completely positive matrix. *SIAM Journal on Matrix Analysis and Applications*, 34:355–368, 2013.

[147] P. Sheng-zhi and W. Fu-sheng. Semidefinite programming relaxation for portfolio selection with higher order moments. *2011 International Conference on Management Science & Engineering 18th Annual Conference Proceedings*, pages 99–104, 2011.

[148] N. D. Sidiropoulos and R. Bro. On the uniqueness of multilinear decomposition of N-way arrays. *Journal of Chemometrics*, 14(3):229–239, 2000.

[149] J. C. Singleton and J. Wingender. Skewness persistence in common stock returns. *The Journal of Financial and Quantitative Analysis*, 21(3):335–341, 1986.

[150] A. Steenkamp. Convex scalarizations of the mean-variance-skewness-kurtosis problem in portfolio selection, 2023.

[151] A. Steenkamp. Matrix factorization ranks via polynomial optimization. In M. Kočvara, B. Mourrain, and C. Riener, editors, *Polynomial Optimization, Moments, and Applications*, pages 135–162. Springer, 2023.

[152] N. Taleb. *Statistical Consequences of Fat Tails: Real World Preasymptotics, Epistemology, and Applications*. Technical Incerto. STEM Academic Press, 2020.

[153] G. Tang and P. Shah. Guaranteed tensor decomposition: A moment approach. In F. Bach and D. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, pages 1491–1500. PMLR, 2015.

[154] V. Tchakaloff. Formules de cubature mécaniques à coéfficients non négatifs. *Bulletin des Sciences Mathématiques*, 81:123–134, 1957.

[155] S. Telen, M. V. Barel, and J. Verschelde. A robust numerical path tracking algorithm for polynomial homotopy continuation. *Siam Journal On Scientific Computing*, 42:A3610–A3637, 2019.

[156] A. Uhlmann. Entropy and optimal decompositions of states relative to a maximal commutative subalgebra. *Open Systems & Information Dynamics*, 5(3):209–228, 1998.

[157] S. A. Vavasis. On the complexity of nonnegative matrix factorization. *SIAM Journal on Optimization*, 20:1364–1377, 2009.

[158] D. Wackerly, W. Mendenhall, and R. Scheaffer. *Mathematical Statistics with Applications*. Cengage Learning, 2014.

[159] H. Waki, S. Kim, M. Kojima, and M. Muramatsu. Sums of squares and semidefinite program relaxations for polynomial optimization problems with structured sparsity. *SIAM Journal on Optimization*, 17(1):218–242, 2007.

[160] J. Wang and V. Magron. A second order cone characterization for sums of nonnegative circuits. In *Proceedings of the 45th International Symposium on Symbolic and Algebraic Computation*, pages 450–457, 2020.

[161] J. Wang, V. Magron, and J.-B. Lasserre. Chordal-TSSOS: A moment-SOS hierarchy that exploits term sparsity with chordal extension. *SIAM Journal on Optimization*, 31(1):114–141, 2021.

[162] J. Wang, V. Magron, and J.-B. Lasserre. TSSOS: A moment-SOS hierarchy that exploits term sparsity. *SIAM Journal on Optimization*, 31(1):30–58, 2021.

[163] J. Wang, V. Magron, J. B. Lasserre, and N. H. A. Mai. CS-TSSOS: Correlative and term sparsity for large-scale polynomial optimization. *ACM Trans. Math. Softw.*, 48(4), 2022.

[164] W. Wang and M. Á. Carreira-Perpiñán. Projection onto the probability simplex: An efficient algorithm with a simple proof, and an application. *ArXiv*, abs/1309.1541, 2013.

[165] J. Watrous. *The Theory of Quantum Information*. Cambridge University Press, Cambridge, 2018.

[166] J. H. M. Wedderburn. *Lectures on Matrices*. Dover Publications Inc., New York, 1964.

[167] S. Woronowicz. Positive maps of low dimensional matrix algebras. *Reports on Mathematical Physics*, 10(2):165–183, 1976.

[168] S. Xiang and S. Xiang. Notes on completely positive matrices. *Linear Algebra and its Applications*, 271(1):273–282, 1998.

[169] M. Yamashita, K. Fujisawa, and M. Kojima. Implementation and evaluation of SDPA 6.0 (SemiDefinite programming algorithm 6.0). *Optimiz. Methods Software*, pages 491–505, 2003.

[170] M. Yamashita, K. Fujisawa, K. Nakata, M. Nakata, M. Fukuda, K. Kobayashi, and K. Goto. A high-performance software package for semidefinite programs: SDPA7. Technical report, 2010.

[171] M. Yannakakis. Expressing combinatorial optimization problems by linear programs. In *STOC '88*, 1988.

[172] Y. Zheng and G. Fantuzzi. Sum-of-squares chordal decomposition of polynomial matrix inequalities. *Mathematical Programming*, pages 1–38, 2021.

[173] Y. Zheng, G. Fantuzzi, and A. Papachristodoulou. Chordal and factor-width decompositions for scalable semidefinite and polynomial optimization. *Annual Reviews in Control*, 52:243–279, 2021.

[174] R. Zhou and D. P. Palomar. Solving high-order portfolios via successive convex approximation algorithms. *IEEE Transactions on Signal Processing*, 69:892–904, 2021.

CENTER DISSERTATION SERIES

CentER for Economic Research, Tilburg University, the Netherlands

No.	Author	Title	ISBN	Published
672	Joobin Ordoobody	The Interplay of Structural and Individual Characteristics	978 90 5668 674 1	February 2022
673	Lucas Avezum	Essays on Bank Regulation and Supervision	978 90 5668 675 8	March 2022
674	Oliver Wichert	Unit-Root Tests in High-Dimensional Panels	978 90 5668 676 5	April 2022
675	Martijn de Vries	Theoretical Asset Pricing under Behavioral Decision Making	978 90 5668 677 2	June 2022
676	Hanan Ahmed	Extreme Value Statistics using Related Variables	978 90 5668 678 9	June 2022
677	Jan Paulick	Financial Market Information Infrastructures: Essays on Liquidity, Participant Behavior, and Information Extraction	978 90 5668 679 6	June 2022
678	Freek van Gils	Essays on Social Media and Democracy	978 90 5668 680 2	June 2022
679	Suzanne Bies	Examining the Effectiveness of Activation Techniques on Consumer Behavior in Temporary Loyalty Programs	978 90 5668 681 9	July 2022
680	Qinnan Ruan	Management Control Systems and Ethical Decision Making	978 90 5668 682 6	June 2022
681	Lingbo Shen	Essays on Behavioral Finance and Corporate Finance	978 90 5668 683 3	August 2022
682	Joshua Eckblad	Mind the Gales: An Attention-Based View of Startup Investment Arms	978 90 5668 684 0	August 2022
683	Rafael Greminger	Essays on Consumer Search	978 90 5668 685 7	August 2022
684	Suraj Upadhyay	Essay on policies to curb rising healthcare expenditures	978 90 5668 686 4	September 2022

No.	Author	Title	ISBN	Published
685	Bert-Jan Butijn	From Legal Contracts to Smart Contracts and Back Again: An Automated Approach	978 90 5668 687 1	September 2022
686	Sytse Duiverman	Four essays on the quality of auditing: Causes and consequences	978 90 5668 688 8	October 2022
687	Lucas Slot	Asymptotic Analysis of Semidefinite Bounds for Polynomial Optimization and Independent Sets in Geometric Hypergraphs	978 90 5668 689 5	September 2022
688	Daniel Brosch	Symmetry reduction in convex optimization with applications in combinatorics	978 90 5668 690 1	October 2022
689	Emil Uduwalage	Essays on Corporate Governance in Sri Lanka	978 90 5668 691 8	October 2022
690	Mingjia Xie	Essays on Education and Health Economics	978 90 5668 692 5	October 2022
691	Peerawat Samranchit	Competition in Digital Markets	978 90 5668 693 2	October 2022
692	Jop Schouten	Cooperation, allocation and strategy in interactive decision-making	978 90 5668 694 9	December 2022
693	Pepijn Wissing	Spectral Characterizations of Complex Unit Gain Graphs	978 90 5668 695 6	November 2022
694	Joris Berns	CEO attention, emotion, and communication in corporate financial distress	978 90 5668 696 3	November 2022
695	Tom Aben	The (long) road towards smart management and maintenance: Organising the digital transformation of critical infrastructures	978 90 5668 697 0	December 2022
696	Gülbike Mirzaoğlu	Essays in Economics of Crime Prevention and Behavior Under Uncertainty	978 90 5668 698 7	February 2023
697	Suwei An	Essays on incentive contracts, M&As, and firm risk	978 90 5668 699 4	February 2023
698	Jorgo Goossens	Non-standard Preferences in Asset Pricing and Household Finance	978 90 5668 700 7	February 2023

No.	Author	Title	ISBN	Published
699	Santiago Bohorquez Correa	Risk and rewards of residential energy efficiency	978 90 5668 701 4	April 2023
700	Gleb Gertsman	Behavioral Preferences and Beliefs in Asset Pricing	978 90 5668 702 1	May 2023
701	Gabriella Massenz	On the Behavioral Effects of Tax Policy	978 90 5668 703 8	May 2023
702	Yeqiu Zheng	The Effect of Language and Temporal Focus on Cognition, Economic Behaviour, and Well-Being	978 90 5668 704 5	May 2023
703	Michela Bonani	Essays on Innovation, Cooperation, and Competition Under Standardization	978 90 5668 705 2	June 2023
704	Fabien Ize	The Role of Transparency in Fairness and Reciprocity Issues in Manager-Employee Relationships	978 90 5668 706 9	June 2023
705	Kristel de Nobrega	Cyber Defensive Capacity and Capability: A Perspective from the Financial Sector of a Small State	978 90 5668 707 6	July 2023
706	Christian Peters	The Microfoundations of Audit Quality	978 90 5668 708 3	June 2023
707	Felix Kirschner	Conic Optimization with Applications in Finance and Approximation Theory	978 90 5668 709 0	July 2023
708	Zili Su	Essays on Equity Incentive and Share Pledging in China	978 90 5668 710 6	September 2023
709	Rafael Escamilla	Managing the Nanostore Supply Chain: Base-of-the-Pyramid Retail in Emerging Markets	978 90 5668 711 3	September 2023
710	Tomas Jankauskas	Essays in Empirical Finance	978 90 5668 712 0	August 2023
711	Tung Nguyen Huy	Fostering Sustainable Land Management in Sub-Saharan Africa: Evidence from Ghana and Burkina Faso	978 90 5668 713 7	September 2023
712	Daniel Karpati	Essays in Finance & Health	978 90 5668 714 4	September 2023
713	Mylène Struijk	IT Governance in the Digital Era: Insights from Meta-Organizations	978 90 5668 715 1	September 2023

No.	Author	Title	ISBN	Published
714	Albert Rutten	Essays on Work and Retirement	978 90 5668 716 8	November 2023
715	Yan Liu	Essays on Credit Rating Agencies in China	978 90 5668 717 5	October 2023
716	Xiaoyue Zhang	Input Distortions and Industrial Upgrading in China	978 90 5668 718 2	September 2023
717	Andries van Beek	Solutions in multi-actor projects with collaboration and strategic incentives	978 90 5668 719 9	October 2023
718	Andries Steenkamp	Polynomial Optimization: Matrix Factorization Ranks, Portfolio Selection, and Queueing Theory	978 90 5668 720 5	October 2023

Inspired by Leonhard Euler's belief that every event in the world can be understood in terms of maximizing or minimizing a specific quantity, this thesis delves into the realm of mathematical optimization. The thesis is divided into four parts, with optimization acting as the unifying thread.

Part 1 introduces a particular class of optimization problems called generalized moment problems (GMPs) and explores the moment method, a powerful tool used to solve GMPs. We introduce the new concept of ideal sparsity, a technique that aids in solving GMPs by improving the bounds of their associated hierarchy of semidefinite programs.

Part 2 focuses on matrix factorization ranks, in particular, the nonnegative rank, the completely positive rank, and the separable rank. These ranks are extensively studied using the moment method, and ideal sparsity is applied (whenever possible) to enhance the bounds on these ranks and speed-up their computation.

Part 3 centers around portfolio optimization and the mean-variance-skewness-kurtosis (MVSK) problem. Multi-objective optimization techniques are employed to uncover Pareto optimal solutions to the MVSK problem. We show that most linear scalarizations of the MVSK problem result in specific convex polynomial optimization problems which can be solved efficiently.

Part 4 explores hypergraph-based polynomials emerging from queueing theory in the setting of parallel-server systems with job redundancy policies. By exploiting the symmetry inherent in the polynomials and some classical results on matrix algebras, the convexity of these polynomials is demonstrated, thereby allowing us to prove that the polynomials attain their optima at the barycenter of the simplex.

Andries Steenkamp (Pretoria, South Africa, 1992) received his bachelor's degree in Mathematics (with distinction) from the University of Pretoria in 2015. He obtained his master's degree in Mathematics from ETH Zurich in 2018. In 2019, he joined Centrum Wiskunde & Informatica (CWI) in Amsterdam as a PhD candidate in the Marie Skłodowska-Curie Training Network POEMA.

ISBN: 978 90 5668 720 5
DOI: 10.26116/pep2-4m10

TILBURG SCHOOL OF ECONOMICS
AND MANAGEMENT

CentER Dissertation Series