Mixed-Projection Conic Optimization:
A New Paradigm for Modeling Rank Constraints

Dimitris Bertsimas
Sloan School of Management and Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA, USA.
ORCID: 0000-0002-1985-1003
dbertsim@mit.edu

Ryan Cory-Wright
Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA, USA,
ORCID: 0000-0002-4485-0619
ryancw@mit.edu

Jean Pauphilet
Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA, USA,
ORCID: 0000-0001-6352-0984
jpauph@mit.edu

We propose a framework for modeling and solving low-rank optimization problems to certifiable optimality.
We introduce symmetric projection matrices that satisfy $Y^2 = Y$, the matrix analog of binary variables that satisfy $z^2 = z$, to model rank constraints. By leveraging regularization and strong duality, we prove that this modeling paradigm yields tractable convex optimization problems over the non-convex set of orthogonal projection matrices. Furthermore, we design outer-approximation algorithms to solve low-rank problems to certifiable optimality, compute lower bounds via their semidefinite relaxations, and provide near optimal solutions through rounding and local search techniques. We implement these numerical ingredients and, for the first time, solve low-rank optimization problems to certifiable optimality. Our algorithms also supply certifiably near-optimal solutions for larger problem sizes and outperform existing heuristics, by deriving an alternative to the popular nuclear norm relaxation which generalizes the perspective relaxation from vectors to matrices. All in all, our framework, which we name Mixed-Projection Conic Optimization, solves low-rank problems to certifiable optimality in a tractable and unified fashion.

Key words: rank minimization, semidefinite optimization, global optimization, discrete optimization, outer-approximation, regularization, perspective relaxation, matrix completion, nuclear norm

Subject classifications: programming: non-linear: quadratic, algorithms, applications

Area of review: Optimization

1. Introduction

Many central problems in optimization, machine learning, and control theory are equivalent to optimizing a low-rank matrix over a convex set. For instance, low-rank constraints successfully model notions of minimal complexity, low dimensionality, or orthogonality in a system. However, while rank constraints offer unparalleled modeling flexibility, no generic code currently solves these problems to certifiable optimality at even moderate sizes. This state of affairs has led influential works on low-rank optimization (Candès and Plan 2010, Recht et al. 2010) to characterize low-rank optimization as intractable and advocate convex relaxations or heuristics which do not enjoy assumption-free optimality guarantees.
The manner in which rank constrained optimization is regarded today is reminiscent of how mixed-integer conic optimization (MICO), which can model NP-complete problems, was originally considered. After decades of research effort, however, algorithms and software for MICO are now widely available (see, e.g., Bonami et al. 2008, Coey et al. 2020) and solve large instances of disparate non-convex problems such as best subset selection (Bertsimas and Van Parys 2020) or the Traveling Salesperson Problem (Padberg and Rinaldi 1991) to certifiable optimality. Unfortunately, rank constraints cannot be represented using MICO (Lubin et al. 2019, Lemma 4.1) and do not benefit from these advances.

In this work, we characterize the complexity of rank constrained optimization and propose a new, more general framework, which we term Mixed-Projection Conic Optimization (MPCO). Our proposal generalizes MICO, by replacing binary variables \( z \) which satisfy \( z^2 = z \) with symmetric orthogonal projection matrices \( Y \) which satisfy \( Y^2 = Y \), and offers the following advantages over existing state-of-the-art methods: First, it supplies certificates of (near) optimality for low-rank problems. Second, it demonstrates that some of the best ideas in MICO, such as decomposition methods, cutting-planes, relaxations, and random rounding schemes, admit straightforward extensions to MPCO. Finally, we implement a near-optimal rounding strategy and a globally optimal cutting-plane algorithm that improve upon the state-of-the-art for matrix completion and sensor location problems. We hope that MPCO gives rise to exciting new challenges for the optimization community to tackle.

1.1. Scope of the Framework

Formally, we consider the problem:

\[
\min_{X \in \mathbb{R}^{n \times m}} \lambda \cdot \text{Rank}(X) + \langle C, X \rangle \quad \text{s.t.} \quad AX = B, \quad \text{Rank}(X) \leq k, \quad X \in \mathcal{K},
\]

where \( \lambda \) (resp. \( k \)) prices (bounds) the rank of \( X \), \( (A, B) \in \mathbb{R}^{\ell \times n} \times \mathbb{R}^{\ell \times m} \) defines an affine subspace, and \( \mathcal{K} \) is a proper cone in the sense of Boyd and Vandenberghe (2004), i.e., closed, convex, solid and pointed. Observe that Problem (1) offers significant modeling flexibility, as it allows arbitrary conic constraints on \( X \). As a result, linear, convex quadratic, semidefinite, exponential, and power constraints and objectives can be captured by letting \( \mathcal{K} \) be an appropriate product of the non-negative orthant and the second order, semidefinite, exponential, and power cones.

We now present some central problems from the optimization and machine learning literature which admit low-rank formulations and fall within our framework.

1.1.1. Low-Rank Matrix Completion

Given a sub-sample \( (A_{i,j} : (i,j) \in \mathcal{I} \subseteq [n] \times [p]) \) of a matrix \( A \in \mathbb{R}^{n \times p} \), the matrix completion problem is to recover the entire matrix, by assuming \( A \) is low rank and seeking a rank-\( k \) matrix \( X \) which approximately fits the observed values. This problem arises in recommender system applications at Netflix and Amazon (see Figure 1) and admits the formulation:

\[
\min_{X \in \mathbb{R}^{n \times p}} \frac{1}{2} \sum_{(i,j) \in \mathcal{I}} (X_{i,j} - A_{i,j})^2 \quad \text{s.t.} \quad \text{Rank}(X) \leq k.
\]
A low-rank matrix completion problem which arises in recommender systems. 100 users have rated 20% of the 100 movies available in a library, and the system imputes the ratings of the other 80% of movies, by assuming that the customer-rating pairs depend on 5 features and imputing a rank-5 matrix.

Note that convex quadratic objectives can be rewritten as linear objectives over second-order cones (Boyd and Vandenberghe 2004). Therefore, Problem (2) is certainly a special case of Problem (1). Since there are \((n+p)k\) degrees of freedom in a singular value decomposition of a rank-\(k\) matrix \(X \in \mathbb{R}^{n \times p}\), Problem (2) is not well-defined unless \(|I| \geq (n+p)k\). Nonetheless, this condition is often satisfied in practice, since high-dimensional matrices are approximately low rank (Udell and Townsend 2019).

1.1.2. Minimum Dimension Euclidean Distance Embedding

Given a set of pairwise distances \(d_{i,j}\), the Euclidean Distance Embedding (EDM) problem is to determine the lowest dimensional space which the distances can be embedded in, such that the distances correspond to Euclidean distances. This problem arises in protein folding, network sensor location, and satellite ranging applications among others (Biswas and Ye 2004, Liberti et al. 2014). By Blekherman et al. (2012) Theorem 2.49, a set of distances \(d_{i,j}\) can be embedded in a Euclidean space of dimension \(k\) if and only if there exists some Gram matrix \(G \succeq 0\) of rank \(k\) such that \(d_{i,j}^2 = G_{i,i} + G_{j,j} - 2G_{i,j}\), on all pairs \((i,j)\) where \(d_{i,j}\) is supplied. Denoting \(D_{i,j} = d_{i,j}^2\), we write these constraints in matrix form, \(D = \text{Diag}(G)e^\top + e\text{Diag}(G)^\top - 2G\), where the equality is implicitly imposed only for pairs \((i,j)\) where \(d_{i,j}\) is supplied. This is equivalent to:

\[
\min_{G \in S_n^+} \text{Rank}(G) \quad \text{s.t.} \quad \text{Diag}(G)e^\top + e\text{Diag}(G)^\top - 2G = D. \tag{3}
\]

Given a solution \(G\), we can obtain the matrix of coordinates of the underlying points \(X\) (up to a rotation and translation of the points) by performing a Cholesky decomposition of \(G\), \(G = XX^\top\). Post decomposition, \(X\) is a \(n \times k\) rectangular matrix which contains the coordinates of the underlying points.

1.1.3. Quadratically Constrained Quadratic Optimization

A quadratically constrained quadratic optimization problem (QCQOP) seeks an \(x \in \mathbb{R}^n\) which solves:

\[
\min_{x \in \mathbb{R}^n} x^\top Q_0 x + q_0^\top x \quad \text{s.t.} \quad x^\top Q_i x + q_i^\top x \leq r_i \quad \forall i \in [m], \tag{4}
\]

where \(Q_0, Q_i, q_0, q_i, r_i\) are given problem data. We assume that \(Q_0, Q_i\) are symmetric matrices, but do not assume that they are positive semidefinite. Therefore, this problem is non-convex, and encompasses
binary quadratic optimization (Goemans and Williamson 1995) and alternating current optimal power flow problems (Lavaei and Low 2011). The fundamental difficulty in Problem (4) is the potential non-convexity of the outer product $xx^\top$. However, we can isolate this non-convexity by introducing a rank-one matrix $X$ to model the outer product $xx^\top$. This leads to the following reformulation:

$$\min_{x \in \mathbb{R}^n, X \in S_n} \langle Q_0, X \rangle + \langle q_0, x \rangle \quad \text{s.t.} \quad \langle Q_i, X \rangle + \langle q_i, x \rangle \leq r_i \quad \forall i \in [m], \quad \text{Rank} \left( \begin{bmatrix} 1 \\ x^\top \\ X \end{bmatrix} \right) = 1. \quad (5)$$

We have established that QCQOPs are rank constrained problems. Notably however, the converse is also true: rank constrained problems with linear, second-order cone, or semidefinite constraints are QCQOPs. Indeed, the constraint $\text{Rank}(X) \leq k$ is equivalent to requiring that $X = UV^\top: U \in \mathbb{R}^{n \times k}, V \in \mathbb{R}^{m \times k}$, i.e., imposing $m \times n$ non-convex quadratic equalities. As modern solvers such as Gurobi can now solve non-convex QCQOPs to global optimality, this QCQOP formulation can be used to solve low-rank problems, although it is not particularly scalable; we expand on this point in Section 4.1.2.

1.1.4. Minimal Degree Sum-of-Squares Decomposition of a Polynomial Many central problems in optimization and control can be addressed by optimizing over the space of globally non-negative polynomials. As separating over this space exactly is NP-hard (Murty and Kabadi 1987) and requires invoking computationally expensive results from real algebraic geometry such as Stengle’s Positivstellensatz (see, e.g., Blekherman et al. 2012, Section 3.4.3), non-negative polynomial optimization is typically addressed by taking a safe inner approximation, namely the set of polynomials which are a sum of squares (Lasserre 2001, Parrilo 2003). For the sake of both interpretability and tractability, a desirable attribute is to obtain a polynomial composed of a sum of at most $k$ squares, where $k$ is small. Recalling that a polynomial $p(z)$ of degree $2d$ is a sum-of-squares (SOS) if and only if $p(x) = z^\top Q z$, where $z = [1, x_1, \ldots, x_n, x_1 x_2, \ldots, x_n^d]$ and $Q$ is a PSD matrix (Parrilo 2003), the minimal SOS decomposition of a polynomial is given by:

$$\min_{Q \succeq 0} \text{Rank}(Q) \quad \text{s.t.} \quad p(x) = z^\top Q z, \quad \text{where} \quad z = [1, x_1, \ldots, x_n, x_1 x_2, \ldots, x_n^d]. \quad (6)$$

This formulation allows us to optimize over the space of low-complexity SOS polynomials.

1.1.5. Further Problems We briefly summarize some further applications of rank constraints.

Sparse Plus Low-Rank Decomposition In many engineering applications, one wishes to decompose a matrix $A \in \mathbb{R}^{n \times m}$ into a sparse matrix $X$ plus a low-rank matrix $Y$ (Chandrasekaran et al. 2011). This is equivalent to solving the following problem, where $\lambda > 0$ balances the importance of rank/sparsity:

$$\min_{X, Y \in \mathbb{R}^{n \times m}} \|X\|_0 + \lambda \cdot \text{Rank}(Y) \quad \text{s.t.} \quad X + Y = A. \quad (7)$$

Low-Rank Factor Analysis An important problem in statistics, psychometrics and economics is to decompose a covariance matrix $\Sigma$ into a low-rank matrix $X$ plus a diagonal matrix $\Phi$, as explored by Saunderson et al. (2012) and references therein. This corresponds to solving:

$$\min_{X, \Phi \in S_n^+} \|\Sigma - \Phi - X\|_F^2 \quad \text{s.t.} \quad \text{Rank}(X) \leq k, \Phi_{i,j} = 0 \quad \forall i,j \in [n]: i \neq j. \quad (8)$$
Bilinear Matrix Inequalities A central problem in control theory is the bilinear matrix inequality problem (Boyd et al. 1994). Among other problems, designing stable controllers and output feedback stabilization problems are special cases of this problem. We seek controls $x, y$ which solve
\[
\min_{x \in \mathbb{R}^m, y \in \mathbb{R}^n} b^T x + c^T y \quad \text{s.t.} \quad F_0 + \sum_{i=1}^m x_i F_i + \sum_{j=1}^n y_j G_j + \sum_{i=1}^m \sum_{j=1}^n x_i y_j H_{i,j} \succeq 0,
\] (9)
where $F_0, F_i, G_j, H_{i,j}$ are symmetric matrices which characterize the systems’ dynamics. This problem can be addressed using a rank constraint, by letting a rank-one matrix model the product $xy^T$.

Non-Negative Matrix Optimization Several important problems in combinatorial optimization, statistics and computer vision (see, e.g., Yannakakis 1991, Burer 2009, Moitra 2016), reduce to optimizing over the space of low-rank matrices with non-negative factors. Formally, this problem is:
\[
\min_{X \in \mathbb{R}^{n \times m}, U \in \mathbb{R}^{n \times k}_+, V \in \mathbb{R}^{k \times m}_+} \langle C, X \rangle \quad \text{s.t.} \quad \langle A_i, X \rangle = b_i \quad \forall i \in [m], \ X = UV^T.
\] (10)

1.2. Background and Literature Review

Our work arises at the intersection of three complementary areas of the low-rank optimization literature: (a) global optimization algorithms for non-convex quadratically constrained problems, (b) the interplay of convex relaxations and their dual side, randomized rounding methods, and (c) heuristics which provide high-quality solutions to non-convex problems in an efficient fashion.

1.2.1. Global Optimization Techniques A broad class of global optimization algorithms have been proposed for QCQOPs, since McCormick (1976) observed that convex envelopes of non-convex regions supply globally valid lower bounds. This observation gives rise to a numerical strategy where one recursively partitions the QCQOP’s feasible region into subregions, constructs convex envelopes for each subregion and uses these envelopes to construct iteratively improving lower bounds. This approach is known as spatial branch-and-bound; see Kocuk et al. (2016, 2018) for modern implementations in alternating current optimal power flow systems.

In a complementary direction, several branch-and-cut methods (Audet et al. 2000, Linderoth 2005, Chen et al. 2017) have been proposed for solving non-convex QCQOPs, by borrowing decomposition schemes from the mixed-integer nonlinear optimization literature (Duran and Grossmann 1986, Quesada and Grossmann 1992). While often efficient in practice, a common theme in these methods is that the more efficient decomposition schemes used for MINLOs cannot be applied out-of-the-box, because they may fail to converge to a globally optimal solution (see Grossmann 2002, for a counterexample). As a result, non-convex problems need to be preprocessed in an expensive fashion. This preprocessing step has inhibited the use of global optimization methods for low-rank problems; indeed, we are not aware of any works which apply branch-and-cut techniques to solve low-rank problems to certifiable optimality.

By taking a purely algebraic view of rank constraints, several algebraic geometry techniques have been proposed for addressing low-rank SDOs. Among others, d’Aspremont (2003) proposed reformulating a
low-rank constraint as a system of polynomial equations which can be addressed via the sum-of-squares hierarchy (Lasserre 2001, Parrilo 2003). Unfortunately, while the first 1–2 levels of the hierarchy successfully supply high-quality bounds for low-rank problems, solving low-rank problems exactly requires the full hierarchy, which scales notoriously poorly in practice. More recently, Naldi (2018) proposed a semi-algebraic reformulation of rank-constrained SDOs, which can be optimized over via Gröbner basis computation (see Cox et al. 2013, for a general theory). Unfortunately, their approach only scales to successfully solve low-rank problems where $n = 5$ and $r = 3$ in 1,000s of seconds. Indeed, as observed by Recht et al. (2010), it seems unlikely that an algebraic approach can solve low-rank SDOs exactly when $n > 10$.

1.2.2. Convex Relaxations and Random Rounding Methods for Low-Rank Problems

A number of authors have studied convex relaxations of low-rank optimization problems, since Fazel (2002) observed that the nuclear norm of a matrix is the convex envelope of a rank constraint on the set of matrices with spectral norm at most $M$, i.e.,

$$\text{Conv} \left( \{ \mathbf{X} \in \mathbb{R}^{n \times m} \text{ s.t. } \|\mathbf{X}\|_{\sigma} \leq M, \text{Rank} (\mathbf{X}) \leq k \} \right) = \{ \mathbf{X} \in \mathbb{R}^{n \times m} \text{ s.t. } \|\mathbf{X}\|_{\sigma} \leq M, \|\mathbf{X}\|_* \leq kM \}.$$  \hspace{1cm} (11)

Because the epigraph of a nuclear norm is semidefinite representable, this approach gives rise to semidefinite relaxations of low-rank problems which can be computed in polynomial time. In addition, Candès and Plan (2010), Recht et al. (2010) have established rigorous probabilistic conditions under which a nuclear norm relaxation recovers the optimal solution to Problem (1) with high probability under appropriate assumptions on the problem data. Unfortunately, these assumptions are expensive to verify and often not satisfied in practice. As a result, most modern approaches to low-rank optimization are currently heuristic in nature, see Section 1.2.3, and do not incorporate convex relaxations.

A second, complementary, line of work aims to supply certifiably near-optimal solutions to low-rank problems, by rounding their semidefinite relaxations. Initiated by Goemans and Williamson (1995) in the context of binary quadratic optimization, who established that randomly rounding an SDO relaxation supplies a $0.878$-approximation, it has evolved into a very successful framework for solving rank-one optimization problems; see Nemirovski et al. (1999) for a unified approach in the rank-one case. Unfortunately, this line of work has a key drawback. Namely, existing rounding methods do not address rank-$k$ problems such as matrix completion, due to the analytic difficulty of constructing a rounding mechanism which preserves both feasibility and near-optimality in the rank-$k$ case.

In a similar direction, a line of work has developed general conditions under which rank constraints are redundant in low-rank SDOs, which are special cases of (1). Specifically, for the low-rank SDO:

$$\min_{\mathbf{X} \in \mathbb{S}^n_+} \langle \mathbf{C}, \mathbf{X} \rangle \text{ s.t. } \langle \mathbf{A}_i, \mathbf{X} \rangle = b_i \text{ for } i \in [m], \text{ Rank} (\mathbf{X}) \leq k,$$  \hspace{1cm} (12)

Pataki (1998) has established that if $k(k+1) \leq 2(m+1)$ then the rank constraint is redundant, and Barvinok (2002, Chapter V.6.1) (see also Ai et al. (2008), for a constructive version of Barvinok’s result)
has established that if $k \epsilon^2 \geq 8 \ln(4(m+1))$ then the SDO admits a rank-$k$ solution, up to a relative error of $\epsilon$. In the same vein, a second line of work has developed conditions under which low-rank problems admit exact semidefinite reformulations (Ben-Tal and Teboulle 1996, Ben-Tal and Den Hertog 2014, Wang and Kılınç-Karzan 2019, Cifuentes et al. 2020).

1.2.3. Heuristic Methods

Due to the computational difficulty of solving Problem (1) to certifiable optimality, and the analytic difficulty of deriving a high-quality randomized rounding procedure, a variety of heuristic methods have been proposed for solving Problem (1), originating with methods for solving low-rank linear matrix inequalities in the optimal control literature (Boyd et al. 1994).

Although slow and somewhat ad-hoc in their original implementations, heuristic methods were moved front-and-center by the works of Fazel (2002), Burer and Monteiro (2003, 2005). Fazel (2002) observed that low-rank positive semidefinite matrices lie on the boundary of the PSD cone, and used this observation to justify a “log-det” heuristic, where a rank minimization objective is replaced with the function $\log \det(X + \delta I)$. Burer and Monteiro (2003, 2005) proposed implicitly modeling a rank constraint $\text{Rank}(X) \leq k$ by applying the non-linear reformulation $X = UV^T$, where $U, V \in \mathbb{R}^{n \times k}$ and eliminating $X$, to obtain a problem which is non-convex in $(U, V)$. Although originally solved using augmented Lagrangian techniques, modern implementations of the Burer-Monterio heuristic typically use alternating minimization (Jain et al. 2013), to enhance the methods scalability and ensure convergence towards a second-order critical point.

The modern era of heuristics methods for low-rank matrix optimization was initiated by matrix completion and the Netflix competition (Bell and Koren 2007). The magnitude of the data made available, challenged the aforementioned methods and led to new techniques such as Wen et al. (2012), who proposed a nonlinear successive over-relaxation approach that scales to $10,000 \times 10,000$ matrices, Recht and Ré (2013), who proposed a stochastic gradient descent method which scales to $1,000,000 \times 1,000,000$ matrices; see Udell et al. (2016), Nguyen et al. (2019) for reviews of heuristic approaches.

1.3. Contributions and structure

The key contributions of the paper are threefold: First, we propose using orthogonal projection matrices which satisfy $Y^2 = Y$, the matrix analogue of binary variables which satisfy $z^2 = z$, to model low-rank constraints via the non-linear equation $X = YX$. Under this lens, low-rank problems admit reformulations as optimization problems where some decision variables comprise a projection matrix. We term this family of problems Mixed-Projection Conic Optimization (MPCO), in reference to mixed-integer optimization. To our knowledge, our approach is the first mathematical framework which solves low-rank optimization problems to certifiable optimality. Second, by leveraging regularization and strong duality we rewrite low-rank optimization problems as saddle-point problems over the space of orthogonal projection matrices, and propose an outer-approximation method to solve the saddle-point problem to certifiable optimality. Third, by analyzing the saddle-point problem, we derive new convex relaxations...
and rounding schemes which provide certifiably near optimal solutions in polynomial time in theory and rapidly in practice.

The rest of the paper is laid out as follows:

In Section 2, we show that projection matrices are a natural generalization of binary vectors to matrices. Inspired by a common tactic in cardinality constrained optimization, namely introducing binary variables to encode the support of the decision vector, we propose introducing a projection matrix to encode the image of the decision matrix and thereby model rank. We also investigate the complexity of low-rank optimization problems and show that rank minimization is as hard as the existential theory of the reals (i.e., deciding whether a semi-algebraic set is non-empty), and thus in PSPACE.

In Section 3, we derive the MPCO formulations of the aforementioned rank optimization problems. By introducing a constraint on the spectral norm of $X$ or a penalty on its Frobenius norm - the matrix analogs of big-$M$ constraints and perspective formulations (Günlük and Linderoth 2012) respectively, we leverage strong duality, reformulate Problem (1) as a saddle-point problem, and prove the resulting optimization problem admits a convex objective.

We propose numerical algorithms to solve these MPCO problem to provable (near) optimality in Section 4, by extending some of the most successful techniques from MICO. First, we propose an outer-approximation scheme for solving Problem (1) exactly. Then, we obtain valid lower-bounds from solving its convex relaxations and propose an alternating minimization algorithm to do so. In addition, we prove that a singular value decomposition (SVD) followed by greedily rounding the eigenvalues provides certifiably near-optimal solutions in polynomial time. Finally, we propose a local-search strategy to improve the quality of the greedily rounded solution.

In Section 5, we implement and numerically evaluate our proposed algorithms. On examples from matrix completion and sensor location, we demonstrate that methods proposed in this paper solve instances of Problem (1) to certifiable optimality in minutes for $n$ in the tens. To our knowledge, our work is the first to demonstrate that moderately sized rank constrained problems can be solved to provable optimality in a tractable fashion. For $n$ in the hundreds, our proposal scales and provides in minutes solutions of higher quality than existing heuristics, such as nuclear norm minimization.

**Notation:** We let non-bold face characters such as $b$ denote scalars, lowercase bold-faced characters such as $x$ denote vectors, uppercase bold-faced characters such as $X$ denote matrices, and calligraphic uppercase characters such as $Z$ denote sets. We let $[n]$ denote the set of running indices $\{1,...,n\}$. We let $e$ denote a vector of all 1’s, $0$ denote a vector of all 0’s, and $I$ denote the identity matrix.

We also use an assortment of matrix operators. We let $\sigma_i(X)$ denote the $i$th largest singular value of a matrix $X$, $\langle \cdot, \cdot \rangle$ denote the Euclidean inner product between two vectors or matrices of the same dimension, $X^\dagger$ denote the Moore-Penrose pseudoinverse of a matrix $X$, $\| \cdot \|_F$ denote the Frobenius norm of a matrix, $\| \cdot \|_\sigma$ denote the spectral norm of a matrix, and $\| \cdot \|_*$ denote the nuclear norm of a matrix; see Horn and Johnson (1985) for a general theory of matrix operators.
Finally, we use a wide variety of convex cones. We let $S^n$ denote the $n \times n$ cone of symmetric matrices, $S^n_+$ denote the $n \times n$ positive semidefinite cone, $C^n_+ = \{ C \in \mathbb{R}^{n \times n} : C = DD^\top, D \in \mathbb{R}^{n \times n}_+ \}$ denote the $n \times n$ completely positive cone, and $DNN^n = S^n_+ \cap \mathbb{R}^{n \times n}_+ \subseteq C^n_+$ denote the doubly non-negative cone.

2. From Cardinality to Rank: A Unifying Nonlinear Perspective

Low rank constraints $\text{Rank}(X) \leq k$ are a natural generalization of cardinality constraints $\|x\|_0 \leq k$ from vectors to matrices. Indeed, if $X$ is a diagonal matrix then $\text{Rank}(X) \leq k$ if and only if $\|X\|_0 \leq k$, and more generally $\text{Rank}(X) \leq k$ if and only if $\|\sigma(X)\|_0 \leq k$, where $\sigma(X)$ is the vector of singular values of $X$. However, while cardinality and rank constraints are intimately linked, they are addressed using different algorithms. Namely, we can solve cardinality constrained problems with 100,000s of variables to optimality (Bertsimas and Van Parys 2020), while low-rank problems are dramatically harder and have not yet been solved to certifiable optimality for $n > 10$ (Naldi 2018).

In our opinion, the difference between the community’s understanding of cardinality and rank constraints has arisen because of two algorithmic barriers. The first barrier is that rank constraints belong to a harder complexity class. The second barrier arises because cardinality constraints can be represented using binary variables, while rank constraints cannot (Lubin et al. 2019, Corollary 4.1). This presents a challenge for researchers, who have developed scalable methods for cardinality constraints by exploiting advances in mixed-integer optimization (MIO), but cannot use these advances to address rank constraints. In this section, we question these barriers by precisely characterizing the complexity of low-rank problems and proposing a new framework for modeling rank that generalizes MIO.

2.1. On the Complexity of Rank-Constrained Optimization

Existing studies of Problem (1) typically claim that it is intractable, and support this claim by proving that it is NP-hard, by reduction from an NP-complete problem such as Boolean linear programming (see, e.g., Vandenberghe and Boyd 1996, Section 7.3). In our opinion, this argument needs to be revisited, for two separate reasons. First, NP-hardness is a worst-case analysis statement. In practice, NP-hard optimization problems are often tractable. For instance, the sparse regression can usually be solved to certifiable optimality with 100,000s of features in minutes (Bertsimas and Van Parys 2020). Second, there is no evidence that Problem (1) is even in NP. Indeed, Problem (1) cannot be represented using mixed-integer convex optimization (Lubin et al. 2019, Corollary 4.1), while all 21 of Karp’s NP-complete problems can, and the best known algorithms for Problem (1) run in EXPTIME (Chistov and Grigoriev 1984, Naldi 2018).

In this section, we provide a more complete characterization of Problem (1)’s complexity than is currently available in the literature. First, we demonstrate that it belongs to a different class than NP. In particular, it is existential theory of the reals-hard ($\exists \mathbb{R}$-hard; see Renegar (1992) for a general theory), i.e., as hard as any polynomial optimization problem, which implies that, if $\text{NP} \subseteq \exists \mathbb{R}$, Problem (1) is strictly harder than NP-complete problems. Second, we prove that Problem (1) is actually in $\exists \mathbb{R}$.
We now demonstrate that Problem (1) is existential theory of the reals complete (i.e., $\exists\mathbb{R}$-complete). We begin by reminding the reader of the definition of the $\exists\mathbb{R}$ complexity class (c.f. Schaefer 2013):

**Definition 1.** A decision problem belongs to the existential theory of the reals complexity class if it reduces to deciding whether a statement \((\exists x_1, ..., x_n) \phi(x_1, ..., x_n)\) is true or false, where \(\phi(\cdot)\) is a quantifier-free Boolean formula involving polynomials equalities and inequalities, for instance, deciding the emptiness of a semi-algebraic set. We say a problem is $\exists\mathbb{R}$-hard if any problem in $\exists\mathbb{R}$ reduces to it.

Note that 3-SAT $\in \exists\mathbb{R}$, so NP $\subseteq \exists\mathbb{R}$, and any statement in $\exists\mathbb{R}$ can be decided in PSPACE (Canny 1988), so $\exists\mathbb{R} \subseteq$ PSPACE. To establish that Problem (1) is $\exists\mathbb{R}$ hard, we require the following proposition, which is essentially a restatement of (Schaefer 2013, Theorem 3.1) in the language of optimization.

**Proposition 1.** Let \(G := (V,E)\) be a graph, and \(\ell(e)\) be the length of edge \(e\). Then, deciding if \(G\) can be embedded in \(\mathbb{R}^2\) is $\exists\mathbb{R}$ complete, even when all edges have unit length.

By reducing Proposition 1’s planar embedding problem to a Euclidean Distance Embedding problem, we obtain the following result (proof deferred to Appendix A.1):

**Theorem 1.** Problem (1) is $\exists\mathbb{R}$-hard.

Theorem 1 demonstrates that Problem (1) is, from a traditional complexity theory perspective, at least as hard as any problem in $\exists\mathbb{R}$. However, its complexity status remains unresolved. Indeed, while Candès and Plan (2010) have observed that Problem (1) is in EXPTIME, it seems likely that $\exists\mathbb{R} \subset$ EXPTIME. We now address this matter, by proving that if $K$ represents the semidefinite cone then Problem (1) is in $\exists\mathbb{R}$, and hence $\exists\mathbb{R}$-complete; note that the examples listed in Section 1.1 can all be rewritten as low-rank SDOs, so this result applies to all of the aforementioned examples (proof of theorem deferred to Appendix A.2).

**Theorem 2.** Let $K = S_n^+$ denote the $n \times n$ positive semidefinite cone. Then, Problem (1) is in $\exists\mathbb{R}$, and hence $\exists\mathbb{R}$-complete.

**Remark 1.** Since $\exists\mathbb{R} \subseteq$ PSPACE $\subseteq$ EXPTIME, this upper bound improves upon the EXPTIME bound on Problem (1)’s complexity stated by Recht et al. (2010), Candès and Plan (2010) among others. Moreover, it seems unlikely to us that this bound can be further improved without settling fundamental questions in complexity theory (e.g. characterizing NP vs. $\exists\mathbb{R}$ vs. PSPACE vs. EXPTIME).

**Remark 2.** Imposing an additional integrality constraint dramatically changes the complexity of Problem (1). Indeed, under the constraint $X \in \mathbb{Z}^{n \times n}$, we can use a reduction from Hilbert’s 10th problem to show that we cannot decide in finite time whether Problem (1)’s optimal objective is 0, even if we know the objective is binary; see Appendix A.3 for a proof.
2.2. Projection Matrices for Modeling Rank

As previously discussed, rank constraints can be seen as a generalization to the matrix case of cardinality constraints. For a vector \( \mathbf{x} \in \mathbb{R}^n \), the cardinality constraint \( \| \mathbf{x} \|_0 \leq k \) ensures that at most \( k \) coordinates of \( \mathbf{x} \) are non-zero, and can be modeled by introducing a vector of binary variables since

\[
\| \mathbf{x} \|_0 \leq k \iff \exists \mathbf{z} \in \{0, 1\}^n : \mathbf{e}^\top \mathbf{z} \leq k, \quad \mathbf{x} = \mathbf{z} \circ \mathbf{x}, \tag{13}
\]

where \( \mathbf{z} \circ \mathbf{x} \) denotes the component-wise product of \( \mathbf{z} \) and \( \mathbf{x} \). Actually, non-linear constraints of the form “\( \mathbf{x} = \mathbf{z} \circ \mathbf{x} \)” where \( \mathbf{z} \) is binary and \( \mathbf{x} \) is continuous occur in a variety of mixed-integer optimization problems, far beyond cardinality constrained optimization. Bertsimas et al. (2019a) observed that such non-linear constraints “\( \mathbf{x} = \mathbf{z} \circ \mathbf{x} \)” actually lead to tractable optimization problems, provided that the overall objective is appropriately regularized. In particular, big-\( M \) constraints (Glover 1975) and perspective reformulations (Günlük and Linderoth 2012) can be seen as appropriate regularizers. By building upon this observation and the work of several other authors (Fischetti et al. 2016, Bertsimas and Van Parys 2020), they successfully solve cardinality constrained problems at scale via a combination of branch-and-cut, randomized rounding and heuristic methods.

Unfortunately, rank constraints cannot be modeled using mixed-integer convex optimization (Lubin et al. 2019) and the aforementioned methodology cannot be applied. Therefore, we now propose a new framework to model rank in optimization problems. Instead of a binary vector \( \mathbf{z} \) to encode the support of \( \mathbf{x} \), we introduce a projection matrix \( \mathbf{Y} \) to capture the column space of \( \mathbf{X} \) and obtain a similar non-linear reformulation; we lay out several properties of projection matrices used throughout the paper in Appendix C. Precisely, the following proposition holds:

**Proposition 2.** For any \( \mathbf{X} \in \mathbb{R}^{n \times m} \), \( \mathrm{Rank}(\mathbf{X}) \leq k \iff \exists \mathbf{Y} \in \mathcal{Y}_n : \mathrm{tr}(\mathbf{Y}) \leq k, \quad \mathbf{X} = \mathbf{Y} \mathbf{X} \), where \( \mathcal{Y}_n := \{ \mathbf{P} \in \mathbb{S}^n : \mathbf{P}^2 = \mathbf{P} \} \) is the set of \( n \times n \) orthogonal projection matrices.

**Proof of Proposition 2** We prove the two implications successively.

- Let \( \mathbf{X} = \mathbf{U} \Sigma \mathbf{V}^\top \), with \( \mathbf{U} \in \mathbb{R}^{n \times k} \), \( \Sigma \in \mathbb{R}^{k \times k} \), \( \mathbf{V} \in \mathbb{R}^{m \times k} \), be a singular value decomposition of \( \mathbf{X} \) and define \( \mathbf{Y} = \mathbf{U} (\mathbf{U}^\top \mathbf{U})^{-1} \mathbf{U}^\top = \mathbf{U} \mathbf{U}^\top \). By construction, \( \mathbf{X} = \mathbf{Y} \mathbf{X} \), since \( \mathbf{U}^\top \mathbf{U} = \mathbf{I} \). Moreover, \( \mathrm{tr}(\mathbf{Y}) = \mathrm{rank}(\mathbf{Y}) = \mathrm{rank}(\mathbf{X}) \leq k \).

- Since \( \mathbf{X} = \mathbf{Y} \mathbf{X} \), \( \mathrm{rank}(\mathbf{X}) \leq \mathrm{rank}(\mathbf{Y}) = \mathrm{tr}(\mathbf{Y}) \leq k \). \( \square \)

**Remark 3.** In Proposition 2, the rank constraint is expressed via a trace constraint on \( \mathbf{Y} \), the orthogonal projection onto the image or column space of \( \mathbf{X} \). Alternatively, one could model the rank constraint via a matrix \( \mathbf{Y}' \in \mathcal{Y}_m \) such that \( \mathrm{tr}(\mathbf{Y}') \leq k \) and \( \mathbf{X} = \mathbf{X} \mathbf{Y}' \). In this case, \( \mathbf{Y}' \) encodes the projection onto the row space of \( \mathbf{X} \). In practice, one could introduce both \( \mathbf{Y} \) and \( \mathbf{Y}' \) and obtain tighter formulations, at the price of introducing additional notation. We explore this idea in Appendix A.4.

Proposition 2 suggests that projection matrices are to rank constraints what binary variables are to cardinality constraints. Indeed, similarities between the two are evident: binary variables \( z \) are
idempotent scalars which solve $z^2 = z$, while projection matrices $Y$ are idempotent matrices which solve $Y^2 = Y$. Also, if $X$ and $Y$ are diagonal, Proposition 2 recovers cardinality constrained optimization.

We note that the idea of using projection matrices to model low-rank constraints is not entirely new, as Peng and Xia (2005) have proposed reformulating $k$-means clustering as a semidefinite optimization problem over the set of orthogonal projection matrices. However, our proposal has several key points of difference. Namely, (1) we consider optimizing over projection matrices directly, while Peng and Xia (2005) use projection matrices as a vehicle to derive semidefinite relaxations, and (2) we use projection matrices to solve low-rank optimization problems that do not admit mixed-integer reformulations, while $k$-means clustering certainly admits a mixed-integer reformulation (Grötschel and Wakabayashi 1989).

Over the past decades, extensive efforts have been devoted to improving the scalability of mixed-integer optimization. We believe that similar achievements can be obtained for rank constrained problems by adapting techniques from MICO to MPCO. In this direction, Table 1 establishes a dictionary linking cardinality and rank constraints, and demonstrates that many of the techniques developed for binary convex optimization admit generalizations to MPCO, including the main results from our recent work (Bertsimas et al. 2019a). Note that we have not yet established most of the connections claimed in Table 1; this is the focus of the next two sections of the paper.

### Table 1

<table>
<thead>
<tr>
<th>Framework</th>
<th>Bertsimas et al. (2019a)</th>
<th>This paper</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parsimony concept</td>
<td>cardinality</td>
<td>rank</td>
</tr>
<tr>
<td>Non-convex outer set</td>
<td>binaries</td>
<td>orthogonal projection matrices</td>
</tr>
<tr>
<td>Strongly convex regularizer</td>
<td>$\ell^2_2$</td>
<td>Frobenius squared</td>
</tr>
<tr>
<td>Boundedness regularizer</td>
<td>$\ell_\infty$</td>
<td>spectral</td>
</tr>
<tr>
<td>Non-linear formulation</td>
<td>$x = x \circ z; z \in {0,1}^n$</td>
<td>$X = YX, Y \in \mathcal{Y}_n$</td>
</tr>
<tr>
<td>Big-M formulation</td>
<td>$-Mz \leq x \leq Mz$</td>
<td>$\begin{bmatrix} MY &amp; X \ X^T &amp; M I \end{bmatrix} \succeq 0$</td>
</tr>
<tr>
<td>Perspective formulation</td>
<td>$\begin{bmatrix} \theta_i \mid x_i \nu_i \end{bmatrix} \succeq 0$</td>
<td>$\begin{bmatrix} \theta \mid X \nu \end{bmatrix} \succeq 0$</td>
</tr>
<tr>
<td>Convex relaxation complexity</td>
<td>linear/second-order cone</td>
<td>semidefinite</td>
</tr>
<tr>
<td>Greedy rounding mechanism</td>
<td>coordinate-wise</td>
<td>singular value decomposition</td>
</tr>
</tbody>
</table>

### 3. Examples and a Saddle-Point Reformulation

In this section, we demonstrate that many central problems in the optimization, machine learning, statistics, and control literatures fit within our framework, and discuss the role played by regularization.

Throughout this paper, we let $\mathcal{Y}_n := \{P \in S^n : P^2 = P\}$ denote the set of $n \times n$ orthogonal projection matrices and $\mathcal{Y}^k_n := \{P \in S^n : P^2 = P, \text{tr}(P) \leq k\}$ denote projection matrices with rank at most $k$; see Appendix C for a collection of useful properties of projection matrices invoked throughout the paper.
3.1. Placing the Examples Within the Framework

By invoking Proposition 2, we now rewrite Problem (1) as the following mixed-projection conic problem:

$$\min_{Y \in \mathcal{Y}_n^{kn}} \min_{X \in \mathbb{R}^{n \times m}} \lambda \cdot \text{tr}(Y) + \langle C, X \rangle \quad \text{s.t.} \quad AX = B, \ X = YX, \ X \in \mathcal{K}. \quad (14)$$

Observe that Problem (14) has a two-stage structure which involves first selecting a low-rank projection matrix $Y$ and second selecting a matrix $X$ under the constraint $X = YX$. Moreover, selecting an optimal $X$ given $Y$ is easy, because it involves solving a conic optimization problem under the linear constraint $X = YX$, while selecting an optimal $Y$ is hard, because $\mathcal{Y}_n^{kn}$ is a non-convex set. Therefore, our modeling framework isolates the hardness of Problem (14) in $\mathcal{Y}_n^{kn}$. We now demonstrate that the problems discussed in the introduction can be cast in this form.

3.1.1. Low-Rank Matrix Completion

The low-rank matrix completion problem (2) involves constraining the rank of a matrix $X$. Therefore, this problem fits into our framework via

$$\min_{Y \in \mathcal{Y}_n^k} \min_{X \in \mathbb{R}^{n \times p}} \frac{1}{2} \sum_{(i,j)} (X_{i,j} - A_{i,j})^2 \quad \text{s.t.} \quad X = YX. \quad (15)$$

Note that although the above objective function is quadratic in $X$, it can be rewritten as a linear objective by introducing an epigraph variable and a second-order cone constraint. We have refrained from doing this here, as it’s more convenient to work with the quadratic objective directly.

3.1.2. Minimum Dimension Euclidean Distance Embedding

The minimum dimension EDM embedding problem (3) involves minimizing the rank of a positive semidefinite matrix $G$, under affine constraints. This fits in our framework via:

$$\min_{Y \in \mathcal{Y}_n^k} \min_{G \in \mathbb{S}_n^+} \text{tr}(Y) \quad \text{s.t.} \quad \text{Diag}(G)e^\top + e\text{Diag}(G)^\top - 2G = D, \ G = YG. \quad (16)$$

3.1.3. Quadratically Constrained Quadratic Optimization

We have seen that the quadratically constrained quadratic problem (5) is equivalent to minimizing the rank of its Gram matrix. Therefore, QCQOP is equivalent to:

$$\min_{Y \in \mathcal{Y}_n^k} \min_{G \in \mathbb{S}_n^+} \langle X, Q_0 \rangle + \langle q_0, x \rangle \quad \text{s.t.} \quad \langle X, Q_i \rangle + \langle q_i, x \rangle \leq r_i \quad \forall i \in [m], \ \left( \begin{array}{c} 1 \\ x^\top \\ x \end{array} \right) = Y \left( \begin{array}{c} 1 \\ x^\top \\ x \end{array} \right). \quad (17)$$

3.1.4. Minimal Degree Sum-of-Squares Decomposition

We have seen that computing the minimum degree SOS decomposition of a degree $2d$ polynomial is equivalent to minimizing the rank of its Gram matrix. Therefore, the problem admits the reformulation:

$$\min_{Y \in \mathcal{Y}_n^k} \min_{Q \in \mathbb{S}_n^+} \text{tr}(Y) \quad \text{s.t.} \quad p(x) = z^\top Qz, \ Q = YQ, \ \text{where} \ z = [1, x_1, \ldots, x_n, x_1x_2, \ldots, x_n^{d-1}]. \quad (18)$$
3.2. A Regularization Assumption

To cope with the non-linear constraints $X = YX$ in a tractable fashion, we augment the objective function in (14) with a regularization term. Namely, we consider

$$\min_{Y \in \mathcal{Y}_n^k} \min_{X \in \mathbb{R}^{n \times m}} \lambda \cdot \text{tr}(Y) + \langle C, X \rangle + \Omega(X) \quad \text{s.t.} \quad AX = B, \ X = YX, \ X \in \mathcal{K}. \quad (19)$$

where the regularization term $\Omega(X)$ satisfies the following assumption:

**Assumption 1.** In Problem (19), the regularization term $\Omega(X)$ is one of:

- A spectral norm penalty, $\Omega(X) = 0$ if $\|X\|_\sigma \leq M$ and $\Omega(X) = +\infty$ otherwise.
- A Frobenius norm penalty, $\Omega(X) = \frac{1}{2\gamma} \|X\|^2_F$.

The two regularizers are matrix analogues of the popular big-M constraints (constraints on the $\ell_\infty$ norm of the continuous variables) and ridge regularization (penalty on the $\ell_2^2$ norm) for vectors. In mixed-integer optimization, such regularization terms can efficiently cope with non-linear constraints between continuous and binary variables (Bertsimas et al. 2019a) and motivate our current approach. Practically speaking, regularization can be a natural component of the original problem (14), otherwise we advocate for introducing it artificially, for it leads to tractable algorithms with moderate impact on the resulting solution. For instance, if $M$ is large enough so that the optimal solution to Problem (14), $X^*$, satisfies $\|X^*\|_\sigma \leq M$, Problems (19) and (14) are equivalent. With the Frobenius norm penalty, the gap between Problem (19)’s and (14)’s objective is at most $\frac{1}{2\gamma} \|X^*\|^2_F$, which can certainly be bounded whenever $\text{tr}(X)$ is bounded, as often occurs in practice.

For ease of notation, we let

$$g(X) = \langle C, X \rangle + \begin{cases} 0, & \text{if } AX = B, \ X \in \mathcal{K}, \\ +\infty, & \text{otherwise}, \end{cases}$$

denote the unregularized second-stage cost for a given $X$. Therefore, Problem (19) can be written as:

$$\min_{Y \in \mathcal{Y}_n^k} f(Y) + \lambda \cdot \text{tr}(Y), \quad (20)$$

where $f(Y) := \min_{X \in \mathbb{R}^{n \times m}} g(X) + \Omega(X) \quad \text{s.t.} \quad X = YX \quad (21)$
yields a best choice of $X$ given $Y$.

Observe that both regularizers are coercive (i.e., “blow up” to $+\infty$ as $\|X\| \to \infty$), therefore render all unbounded solutions infeasible and ensure the compactness of the level sets of $X \mapsto g(X) + \Omega(X)$. This alleviates two of the major issues with conic duality (Ben-Tal and Nemirovski 2001, Theorem 2.4.1). Firstly, regularization ensures that optimal solutions to conic problems are attained (see Blekherman et al. 2012, Example 2.27, for a regularization-free counterexample). In our view, unattainable optimal solutions do not arise in practice, but rather indicate a modeling issue. From this perspective, introducing regularization is desirable. Secondly, regularization ensures that infeasibility of a conic system is **certifiable**\(^1\), i.e., there is either a feasible solution or a certificate of infeasibility. In general, such a

\(^1\)Unless the conic dual is also infeasible, this case is unimportant for our purposes, because it only arises when the original problem is itself infeasible for any $Y$, which can be checked a priori.
Table 2  Regularization penalties, conjugates and references to proofs.

<table>
<thead>
<tr>
<th>Penalty</th>
<th>$\Omega(X)$</th>
<th>$\Omega^*(\alpha, Y, V_{11}, V_{22})$</th>
<th>$\frac{\partial}{\partial V_{i,j}} \Omega^*(\alpha, Y)$</th>
</tr>
</thead>
</table>
| Spectral ($X$ rectangular)  | \[
\begin{cases}
0 & \text{if } \|X\|_\sigma \leq M \\
+\infty & \text{o.w.}
\end{cases}
\] | $\frac{M}{2} \langle Y, V_{11} \rangle + \frac{M}{2} \langle I_m, V_{22} \rangle$ | $\frac{M}{2} V_{11,i,j}$ |
| Spectral ($X$ symmetric)    | \[
\begin{cases}
0 & \text{if } \|X\|_\sigma \leq M \\
+\infty & \text{o.w.}
\end{cases}
\] | $M(\langle Y, V_{11} + V_{22} \rangle)$ | $M(V_{11} + V_{22})_{i,j}$ |
| Frobenius                  | $\frac{1}{2\gamma} \|X\|_F^2$ | $\frac{\gamma}{2} \langle \alpha, Y\alpha \rangle$ | $\frac{\gamma}{2} \langle \alpha_i, \alpha_j \rangle$ |

procedure is not possible because a conic system could be infeasible but asymptotically feasible, i.e.,

$\exists X : AX = B, X \in \mathcal{K}$ but $\exists \{X_t\}_{t=1}^\infty : X_t \in \mathcal{K}$ \(\forall t\) with $\|AX_t - B\| \to 0$.

Here, the regularization term ensures that the set of feasible $X$ (with objective at most $\theta_0 \in \mathbb{R}$) is a closed convex compact set. Therefore, $f(Y)$ cannot generate an asymptotically feasible problem.

### 3.3. A Saddle-Point Reformulation

We now reformulate Problem (19) as a saddle-point problem. This reformulation is significant because it proves that the regularization term $\Omega(X)$ drives the convexity and smoothness of $f(Y)$. To derive the problem’s dual, we require:

**Assumption 2.** For each subproblem (21) generated by $f(Y)$ where $Y \in \mathcal{Y}_n^k$, either the optimization problem is infeasible, or strong duality holds.

Assumption 2 holds under Slater’s constraint qualification (Boyd and Vandenberghe 2004, Section 5.2.3). By invoking Assumption 2, the following theorem reformulates (20) as a saddle-point problem:

**Theorem 3.** Suppose that Assumption 2 holds and $\Omega(\cdot)$ is either the spectral or Frobenius regularizer. Then, the following two optimization problems are equivalent:

\[
\begin{align*}
\min_{Y \in \mathcal{Y}_n^k} \min_{X \in \mathbb{R}^{n \times m}} & \quad \lambda \cdot \text{tr}(Y) + g(X) + \Omega(X) \quad \text{s.t.} \quad X = YX, \\
\min_{Y \in \mathcal{Y}_n^k} \max_{\alpha, V_{11}, V_{22}} & \quad \lambda \cdot \text{tr}(Y) + h(\alpha) - \Omega^*(\alpha, Y, V_{11}, V_{22}),
\end{align*}
\]

where $h(\alpha) := \inf_V g(V) - \langle V, \alpha \rangle = \max_{\Pi} \langle b, \Pi \rangle + \begin{cases} 0, & \text{if } C - \alpha - A^\top \Pi \in \mathcal{K}^*, \\
+\infty, & \text{otherwise}
\end{cases}$

is, up to a sign, the Fenchel conjugate of $g$ (see Boyd and Vandenberghe 2004, Chap. 3.3), $\mathcal{K}^* := \{W : \langle W, X \rangle \geq 0 \ \forall X \in \mathcal{K}\}$ denotes the dual cone to $\mathcal{K}$, and $\Omega^*(\alpha, Y, V_{11}, V_{22})$, as defined in Table 2, is notably linear in $Y$. 


Proof of Theorem 3  Let us fix \( Y \in Y_n^k \), and suppose that strong duality holds for the inner minimization problem which defines \( f(Y) \). Then, introducing \( V \in \mathbb{R}^{n \times m} \) such that \( V = YX \), we have

\[
f(Y) = \min_{X, V} g(V) + \Omega(X) \quad \text{s.t.} \quad V = YX.
\]

Let \( \alpha \) denote the dual variables associated with the coupling constraints \( V = YX \). The minimization problem is then equivalent to its dual problem, which is given by:

\[
f(Y) = \max_{\alpha} h(\alpha) + \min_{X} [\Omega(X) + \langle \alpha, YX \rangle],
\]

where \( h(\alpha) := \inf_{V} g(V) - \langle V, \alpha \rangle \) is, up to a sign, the Fenchel conjugate of \( g \). By a standard application of Fenchel duality, it follows that

\[
h(\alpha) = \max_{\Pi} \langle b, \Pi \rangle + \begin{cases} 0, & \text{if } C - \alpha - A^\top \Pi \in K^*, \\ +\infty, & \text{otherwise.} \end{cases}
\]

To obtain the final result, it suffices to explicitly derive \( \Omega^*(\alpha, Y, V_{11}, V_{22}) \) in the two cases. We present these cases in Table 2 (proofs of cases deferred to Appendix A.4).

Alternatively, under either penalty, if the inner minimization problem defining \( f(Y) \) is infeasible, then its dual problem is unbounded by weak duality. \( \square \)

By Theorem 3, when we evaluate \( f(\hat{Y}) \), one of two alternatives occur. The first alternative is that we have \( f(\hat{Y}) < +\infty \) and there is some optimal \( \alpha \). In this case, we construct the lower approximation

\[
f(Y) \geq f(\hat{Y}) + \langle H, Y - \hat{Y} \rangle,
\]

where \( H := -M(\alpha_+ + \alpha_-) \) under a spectral penalty and \( H := -\frac{\gamma}{2} \alpha \alpha^\top \) under a Frobenius penalty. The second alternative is that \( f(\hat{Y}) = +\infty \), in which case, by the conic duality theorem (see Ben-Tal and Nemirovski 2001, Chapter 2) there exists a \( (\alpha, \Pi) \) such that

\[
C - \alpha - A^\top \Pi \in K^*, \quad \text{and} \quad \langle b, \Pi \rangle > \langle -H, \hat{Y} \rangle.
\]

Under this alternative, we can separate \( \hat{Y} \) from the set of feasible \( Y \)'s by imposing the cut \( 0 \geq \langle b, \Pi \rangle + \langle H, Y \rangle \). Under either alternative, we obtain a globally valid first-order underestimator of the form

\[
zf(Y) \geq h + \langle H, Y - \hat{Y} \rangle,
\]

where \( z, h \) are defined as

\[
z = \begin{cases} 1, & \text{if } f(\hat{Y}) < +\infty, \\ 0, & \text{if } f(\hat{Y}) = +\infty, \end{cases} \quad \text{and} \quad h = \begin{cases} f(\hat{Y}), & \text{if } f(\hat{Y}) < +\infty, \\ \langle b, \Pi \rangle + \langle H, \hat{Y} \rangle, & \text{if } f(\hat{Y}) = +\infty. \end{cases}
\]

This observation suggests that a valid numerical strategy for minimizing \( f(Y) \) is to iteratively minimize and refine a piecewise linear underestimator of \( f(Y) \) defined by the pointwise supremum of a finite number of underestimators of the form \( zf(Y) \geq h + \langle H, Y - \hat{Y} \rangle \). Indeed, as we will see in Section 4, this strategy gives rise to the global optimization algorithm known as outer-approximation.

\(^2\)Weak duality implies that the dual problem is either unfeasible or unbounded. Since the feasible set of the maximization problem does not depend on \( Y \), it is always feasible, unless the original problem is itself infeasible. Therefore, we assume without loss of generality that it is unbounded.
Remark 4. In the unregularized case, i.e., $\Omega(X) = 0$, we can derive a similar reformulation:

$$\min_{Y \in \mathcal{Y}_n^k} \max_{\alpha \in \mathbb{R}^{n \times m}} \lambda \cdot \text{tr}(Y) + h(\alpha) \quad \text{s.t.} \quad Y\alpha = 0. \tag{27}$$

Under this lens, regularization of the primal problem is equivalent to a relaxation in the dual formulation: the hard constraint $Y\alpha = 0$ is penalized by $-\Omega^\star(\alpha, Y, V_{11}, V_{22})$.

3.4. Semidefinite Relaxations

To lower bound Problem (20)’s objective, we relax the non-convex constraint $Y \in \mathcal{Y}_n^k$ to

$$Y \in \text{Conv}(\mathcal{Y}_n^k) = \{Y \in S^n : 0 \preceq Y \preceq I, \text{tr}(Y) \leq k\}.$$  

This yields the saddle-point problem

$$\min_{Y \in \text{Conv}(\mathcal{Y}_n^k)} \max_{\alpha, V_{11}, V_{22} \in S^m} \lambda \cdot \text{tr}(Y) + h(\alpha) - \Omega^\star(\alpha, Y, V_{11}, V_{22}). \tag{28}$$

Problem (28) can in turn be reformulated as an SDO. Indeed, under Assumption 2, we obtain a semidefinite formulation by taking Problem (28)’s dual with respect to $\alpha$. Formally, we have the following results (proofs deferred to Appendix A.5 and A.6 respectively):

**Theorem 4.** Suppose that Assumption 2 holds. Then, strong duality holds between:

$$\min_{Y \in \text{Conv}(\mathcal{Y}_n^k)} \max_{\alpha \in \mathbb{R}^{n \times m}} \lambda \cdot \text{tr}(Y) + h(\alpha) - \frac{\gamma}{2} \langle \alpha, Y \alpha \rangle, \tag{29}$$

$$\min_{Y \in \text{Conv}(\mathcal{Y}_n^k)} \min_{X \in \mathbb{R}^{n \times m}, \theta \in S^n} \lambda \cdot \text{tr}(Y) + g(X) + \frac{1}{2\gamma} \text{tr}(\theta) \quad \text{s.t.} \quad \left( \theta X \right) \geq 0. \tag{30}$$

**Theorem 5.** Suppose that Assumption 2 holds. Then, strong duality holds between:

$$\min_{Y \in \text{Conv}(\mathcal{Y}_n^k)} \max_{\alpha \in S^n, V_{11}, V_{22} \succeq 0} \lambda \cdot \text{tr}(Y) + h(\alpha) - M \langle Y, V_{11} + V_{22} \rangle \quad \text{s.t.} \quad \alpha = V_{11} - V_{22}, \tag{31}$$

$$\min_{Y \in \text{Conv}(\mathcal{Y}_n^k)} \min_{X \in S^n} \lambda \cdot \text{tr}(Y) + g(X) \quad \text{s.t.} \quad -MY \preceq X \preceq MY. \tag{32}$$

We now offer some remarks on these bi-dual problems:

- We can derive a more general version of Theorem 5 without the symmetry assumption on $X$ in much the same manner, via the Schur complement lemma.

- Problem (30)’s formulation generalizes the perspective relaxation from vectors to matrices. This suggests that (30) is an efficient formulation for addressing rank constraints, as perspective formulations efficiently address cardinality constrained problems with conic quadratic (Günlük and Linderoth 2012) or power cone (Aktürk et al. 2009) objectives.

- If $g(X)$ is strongly convex, the above penalty can be tightened by first extracting a separable convex term, as explored in the cardinality constrained case by Dong et al. (2015), Han et al. (2020). For instance, if $g(X)$ is derived from a noisy matrix completion problem with complete observations, i.e.,

$$g(X) + \Omega(X) = \frac{\mu}{2} \|X - M\|_F^2 + \frac{1}{2\gamma} \|X\|_F^2,$$
then we can strengthen the relaxation by rewriting the objective as
\[ g(X) + \Omega(X) = -\mu \langle M, X \rangle + \frac{\mu}{2} \| M \|_F^2 + \frac{1}{2(1/\gamma + \mu)^{-1}} \| X \|_F^2, \]
and letting \((1/\gamma + \mu)^{-1}\) play the role of \(\gamma\).

### 3.5. Convex Penalty Interpretations of Relaxations

In this section, we consider instances where rank is penalized in the objective only and interpret the above convex relaxations as penalty functions, in the tradition of Fazel (2002), Recht et al. (2010). In the presence of the Frobenius penalty, our first result generalizes the reverse Huber penalty of Pilanci et al. (2015), Dong et al. (2015) from cardinality to rank objectives (proof deferred to Appendix A.8).

**Proposition 3.** Suppose that Assumption 2 holds. Then, the following problems are equivalent:

\[
\min_{Y \in \text{Conv}(Y_n)} \min_{X \in \mathbb{R}^{n \times m}, \theta \in S^n} \lambda \cdot \text{tr}(Y) + g(X) + \frac{1}{2\gamma} \text{tr}(\theta) \quad \text{s.t.} \quad \theta X Y^\top \succeq 0, \tag{33}
\]

\[
\min_{X \in \mathbb{R}^{n \times m}} g(X) + \sum_{i=1}^n \min \left( \sqrt{\frac{2\lambda}{\gamma}} \sigma_i(X), \lambda + \frac{\sigma_i(X)^2}{2\gamma} \right), \tag{34}
\]

Proposition 3 proposes an alternative to the nuclear norm penalty for approximately solving low-rank problems. This is significant, because many low-rank problems have constraints of the form \(X \succeq 0, \text{tr}(X) = k\) (e.g. sparse PCA, \(k\)-means clustering), and under these constraints a nuclear norm cannot encourage low-rank solutions (Zhang et al. 2013), while Proposition 3’s penalty can.

Our next results relate rank minimization problems with a spectral regularizer to the nuclear norm penalty, in both the square symmetric and the rectangular case (proofs deferred to Appendix A.7):

**Proposition 4.** Suppose that Assumption 2 holds. Then, the following problems are equivalent:

\[
\min_{Y \in \text{Conv}(Y_n)} \min_{X \in S^n} g(X) + \lambda \cdot \text{tr}(Y) \quad \text{s.t.} \quad -MY \preceq X \preceq MY, \tag{35}
\]

\[
\min_{X \in S^n} g(X) + \frac{\lambda}{M} \| X \|_* \quad \text{s.t.} \quad \| X \|_\sigma \leq M. \tag{36}
\]

**Proposition 5.** Suppose that Assumption 2 holds. Then, the following problems are equivalent:

\[
\min_{Y \in \text{Conv}(Y_n), Y' \in \text{Conv}(Y_m)} \min_{X \in \mathbb{R}^{n \times m}} g(X) + \frac{\lambda}{2} \text{tr}(Y) + \frac{\lambda}{2} \text{tr}(Y') \quad \text{s.t.} \quad \left( MY X^\top M' \right) \succeq 0, \tag{37}
\]

\[
\min_{X \in \mathbb{R}^{n \times m}} g(X) + \frac{\lambda}{M} \| X \|_* \quad \text{s.t.} \quad \| X \|_\sigma \leq M. \tag{38}
\]

### 4. Efficient Algorithmic Approaches

In this section, we present an efficient numerical approach to solve Problem (1) and its convex relaxations. The backbone is an outer-approximation strategy, embedded within a non-convex QCQOP branch-and-bound procedure to solve the problem exactly. We also propose rounding heuristics to find good feasible solutions, and semidefinite free methods for optimizing over its convex relaxations.
4.1. A Globally Optimal Cutting-Plane Method

The analysis in the previous section reveals that evaluating $f(Y)$ yields a globally valid first-order underestimator of $f(\cdot)$. Therefore, a numerically efficient strategy for minimizing $f(Y)$ is to iteratively minimize and refine a piecewise linear underestimator of $f(Y)$. This strategy is known as outer-approximation (OA), and was originally proposed by Duran and Grossmann (1986). OA iteratively constructs underestimators of the following form at each iterate $t+1$:

$$f_{t+1}(Y) = \max_{1 \leq i \leq t} \left\{ f(Y_i) + \langle H_i, Y - Y_i \rangle \right\}. \tag{39}$$

By iteratively minimizing $f_{t+1}(Y)$ and imposing the resulting cuts when constructing the next underestimator, we obtain a non-decreasing sequence of underestimators $f_t(Y_t)$ and non-increasing sequence of overestimators $\min_{i \in [t]} f(Y_i)$ which converge to an $\epsilon$-optimal solution within a finite number of iterations; see also Section 3.3 for details on cut generation. Indeed, since $\text{Conv}(\mathcal{Y}_n^{k})$ is a compact set and $f(\cdot)$ is an $L$-Lipschitz continuous function in $Y$, OA never visits a ball of radius $\frac{\epsilon}{L}$ twice.

We now formalize this numerical procedure in Algorithm 1, and state its convergence properties (proof of convergence deferred to Appendix A.9):

**Algorithm 1** An outer-approximation method for Problem (20)

**Require:** Initial solution $Y_1$

$t \leftarrow 1$

repeat

Compute $Y_{t+1}, \theta_{t+1}$ solution of

$$\min_{Y \in \mathcal{Y}^{k,\theta}} \theta + \lambda \cdot \text{tr}(Y) \quad \text{s.t.} \quad z_i \theta \geq h_i + \langle H_i, Y - Y_i \rangle \quad \forall i \in [t].$$

Compute $f(Y_{t+1}), H_{t+1}, z_{t+1}, d_{t+1}$

until $f(Y_t) - \theta_t \leq \epsilon$

return $Y_t$

**Theorem 6.** Suppose that Assumptions 1-2 hold, and that there exists some Lipschitz constant $L$ such that for any feasible $Y, Y' \in \text{Conv}(\mathcal{Y}_n^{k})$ we have: $|f(Y) - f(Y')| \leq L \|Y - Y'\|_F$, and for any feasibility cut $\langle H_i, Y - Y_i \rangle + h_i \leq 0$ we have $|\langle H_i, Y - Y' \rangle| \leq L \|Y - Y'\|_F$. Let $Y_t \in \text{Conv}(\mathcal{Y}_n^{k})$ be a feasible solution returned by the $t^{th}$ iterate of Algorithm 1, where

$$t \geq \left( \frac{Lk}{\epsilon} + 1 \right)^n.$$

Then, $Y_t$ is an $\epsilon$-optimal and $\epsilon$-feasible solution to Problem (19). Moreover, suppose that we set $\epsilon \to 0$. Then, any limit point of $\{Y_t\}_{t=1}^\infty$ solves (19).
4.1.1. Optimizing Over Orthogonal Projection Matrices  To successfully implement Algorithm 1, we need to repeatedly solve optimization problems of the form

$$\min_{Y \in \mathbb{Y}_{n}, \theta} \theta + \lambda \cdot \text{tr}(Y) \quad \text{s.t.} \quad z_i \theta \geq h_i + \langle H_i, Y - Y_i \rangle \quad \forall i \in [t],$$

which requires a tractable representation of $\mathbb{Y}_{n}$. As Gurobi 9.0 contains a globally optimal branch-and-bound method for general QCQOPs, we represent $Y$ by introducing a matrix $U \in \mathbb{R}^{n \times k}$ and requiring that $Y = UU^\top$ and $U^\top U = I$. This allows Algorithm 1 to be implemented by iteratively solving a sequence of QCQOPs and conic optimization problems. Moreover, to decrease the amount of branching required in each iteration of Algorithm 1, we impose an outer-approximation of the valid constraint $Y \succeq UU^\top$. Specifically, we strengthen the formulation by imposing second-order cone relaxations of the PSD constraint. First, we require that the 2 minors in $Y$ are non-negative, as proposed in Ahmadi and Majumdar (2019), Bertsimas and Cory-Wright (2020). Second, we require that the on-diagonal entries of $Y \succeq UU^\top$ are non-negative. Finally, we follow Atamtürk and Gomez (2019, Proposition 5) in taking a second-order cone approximation of the 2 minors in $Y \succeq UU^\top$. All told, we have:\n
$$\min_{Y \in \mathbb{S}^n, U \in \mathbb{R}^{n \times k}, \theta} \theta + \lambda \cdot \text{tr}(Y) \quad \text{s.t.} \quad z_i \theta \geq h_i + \langle H_i, Y - Y_i \rangle \quad \forall i \in [t],$$

$$Y = UU^\top, U^\top U = I, Y_{i,i}Y_{j,j} \geq Y_{i,j}^2 \forall i, j \in [n], Y_{i,i} \geq \sum_{t=1}^{k} U_{i,t}^2, \forall i \in [n], \text{tr}(Y) = k,$n

$$0 \geq \|U_i - U_j\|^2_2 + 2Y_{i,j} - Y_{i,i} - Y_{j,j}, \quad 0 \geq \|U_i - U_j\|^2_2 + 2Y_{i,j} - Y_{i,i} - Y_{j,j} \quad \forall i, j \in [n].$$

Finally, for a given $Y, U$, we strengthen this formulation by imposing second-order cone cuts of the form $\langle Y - UU^\top, uu^\top \rangle \geq 0$, where $u$ is the most negative eigenvector of $Y - UU^\top$, as proposed by Sherali and Fraticelli (2002).

As described, a linear optimization problem over the set of orthogonal projection matrices is solved at each iteration, hence building a new branch-and-bound tree each time. We refer to this implementation as a “multi-tree” method. Although inefficient if implemented naively, multi-tree methods benefit from gradually tightening the numerical tolerance of the solver as the number of cuts increases.

To improve the efficiency of Algorithm 1, one can integrate the entire procedure within a single branch-and-cut tree using lazy callbacks, as originally proposed in the context of MICO by Quesada and Grossmann (1992). Henceforth, we refer to this implementation as a “single-tree” method. However, the benefit from using multi-tree over single-tree is not straightforward for it depends on how the method is engineered. We benchmark both implementations in Section 5.4.

4.1.2. A Simple Benchmark  We now lay out a simple approach for solving low-rank problems exactly, which we will compare against in our numerical experiments. Rather than introducing an orthogonal projection matrix $Y$, we let $X = UV^\top$ where $U \in \mathbb{R}^{n \times k}$ and $V \in \mathbb{R}^{m \times k}$, and $U, V$ are both

3 It should be noted that this formulation is rather complicated because non-convex QCQOP solvers such as Gurobi currently do not model PSD constraints. If they did, we would supplant the second-order cone constraints with $Y \succeq UU^\top$ and thereby obtain a simpler master problem.
bounded in absolute value by big-M constraints of the form \(|U_{i,j}| \leq 1, |V_{i,j}| \leq M\). Assuming that the objective and constraints are QCQOP representable, as occurs for all of the examples mentioned in the introduction, this formulation can then be optimized over using Gurobi’s piecewise linear reformulation technique for general QCQOPs. Formally, a rank constraint \(\text{Rank}(X) \leq k\) leads to:

\[
\min_{X \in \mathbb{R}^{n \times m}, U \in \mathbb{R}^{n \times k}, V \in \mathbb{R}^{m \times k}} \Omega(X) + g(X)
\]

\[
\text{s.t. } X \geq UV^\top - \epsilon E, X \leq UV^\top + \epsilon E, \|U\|_2 \leq 1 \quad \forall i \in [n], \|V\|_\infty \leq M,
\]

where \(E\) is a matrix of all ones. Note however that, as we observe in Section 5, this approach is significantly less efficient than the previously described cutting-plane approaches.

### 4.2. Lower bounds via Semidefinite Relaxations

To certify optimality, high-quality lower bounds are of interest and can be obtained by relaxing the non-convex constraint \(Y \in Y^k_n\) to \(Y \in \text{Conv}(Y^k_n)\). In addition to a valid lower bound on Problem (20)’s objective, the optimal solution to the relaxation \(Y^*\) is a natural candidate for a random rounding strategy, for stringer convex relaxations lead to superior random rounding strategies. We will explore such rounding strategies in detail in the next section.

The convex relaxation yields the optimization problem (28) which can be solved using a cutting-plane method (see Section 4.2.1), an alternating minimization method (see Section 4.2.2) or reformulated as an SDO and solved as such. Since Algorithm 1 is also an outer-approximation scheme, solving the convex relaxation via a cutting-plane method has the additional benefit of producing valid linear lower-approximations of \(f(Y)\) to initialize Algorithm 1 with.

#### 4.2.1. Cutting-Plane Methods for Improving the Root Node Bound

As mentioned previously, Problem (28) can be solved by a cutting-plane method such as Kelley’s algorithm (see Kelley 1960), which is a continuous analog of Algorithm 1 that solves Problem (20) over \(\text{Conv}(Y^k_n)\), rather than \(Y^k_n\). The main benefit of such a cutting-plane method is that the cuts generated are valid for both \(\text{Conv}(Y^k_n)\) and \(Y^k_n\), and therefore can be used to initialize Algorithm 1 and ensure that its initial lower bound is equal to the semidefinite relaxation. As demonstrated by Fischetti et al. (2016) in the context of MICO and facility location problems, this approach often accelerates the convergence of decomposition schemes by orders of magnitude.

Because Kelley’s algorithm converges slowly in practice, we instead implement an accelerated variant, in-out, as recommended by Fischetti et al. (2016). We present pseudocode in Appendix D.1.

Figure 2’s left panel illustrates the convergence of Kelley’s method and the in-out method for solving the semidefinite relaxation of a noiseless matrix completion problem\(^4\). Note that in our plot of the in-out method on the continuous relaxation we omit the time required to first solve the SDO relaxation;

\(^4\) The data generation process is detailed in Section 5.2. Here, \(n = 100, p = 0.25, r = 1\), and \(\gamma = \frac{20}{p}\).
Figure 2  Convergence behavior of Kelley’s method and the in-out method for solving the semidefinite relaxation of a synthetic matrix completion instance where $n = 100$ (left), and lower bounds generated by a single-tree implementation of Algorithm 1 for a synthetic matrix completion instance where $n = 10$ (right).

this is negligible (38.4s) compared to the time required for either approach to solve the relaxation using cutting planes. Observe that the in-out method’s lower bound is both initially better and converges substantially faster to the optimal solution than Kelley’s method. This justifies our use of the in-out method over Kelley’s method for a stabilizing cut loop in numerical experiments.

Once the relaxation is solved, the generated cuts are used to initialize Algorithm 1. Figure 2’s right panel displays the convergence profile of the lower bound of Algorithm 1 initialized with cuts from Kelley’s or the in-out method (with a limit of 100 cuts). We use a single-tree implementation of Algorithm 1\(^5\) and again a noiseless matrix completion setting\(^6\). We also consider the impact of using the SOC inequalities $Y_{i,j}^2 \leq Y_{i,i}Y_{j,j}$ in the master problem formulation. Using the in-out method and imposing the SOC inequalities are both vitally important for obtaining high-quality lower bounds from Algorithm 1. Accordingly, we make use of both ingredients in our numerical experiments.

\subsection*{4.2.2. Solving the Semidefinite Relaxation at Scale via Alternating Minimization}

In preliminary numerical experiments, we found that modern IPM codes such as \texttt{Mosek} 9.0 cannot optimize over the Frobenius/nuclear norm penalties when $n > 200$ on a standard laptop. As real-world low-rank problems are often large-scale, we now explore more scalable alternatives for optimizing over these penalties. As scalable alternatives for the nuclear norm penalty have been studied, we focus on the Frobenius penalty, and refer to (Recht et al. 2010) for nuclear norm minimization. We begin our analysis with the following result (proof deferred to Appendix A.10):

\begin{itemize}
\item We warm-start the upper bound with greedy rounding and the Burer-Monterio local improvement heuristic described in Section 4.3. To mitigate against numerical instability, we opted to be conservative with our parameters, and therefore turned Gurobi’s heuristics off, set FuncPieceError and FuncPieceLength to their minimum possible values ($10^{-5}$ and $10^{-6}$), set the MIP gap to 1% and the time limit for each solve to one hour.
\item Here, $n = 10$, $p = 0.25$, $r = 1$, and $\gamma = \frac{2}{p}$.
\end{itemize}
Proposition 6. For any fixed $X_t$ in Problem (30), an optimal choice of $\theta$ is given by $\theta^\star = X_t^\top (Y^\star)^\dagger X_t$, where $Y^\star = \sum_{i=1}^n \rho_i^\star u_i u_i^\top$, $X_t = U \Sigma V^\top$ is an SVD of $X_t$, and $\rho^\star$ is an optimal solution to the following second order cone problem:

$$\min_{\rho \in [0,1]^n : \mathbf{e}^\top \rho \leq k \lambda \cdot \mathbf{e}^\top \rho + \sum_{i=1}^n \sigma(X_t)^2 \gamma \rho_i},$$

(42)

As optimizing over $X$ for a fixed $Y_t$ is straightforward, Proposition 6 suggests a viable approach for optimize over the Frobenius norm penalty is alternating minimization (AM; see Beck and Teboulle 2009, for a modern implementation). By specializing Beck and Teboulle (2009)'s implementation of AM to the Frobenius norm penalty, we obtain an efficient numerical strategy for obtaining an optimal solution to (30), which we present in Algorithm 2; we discuss implementation details in Appendix D.2.

Algorithm 2 An Accelerated Alternating Minimization Algorithm (c.f. Beck and Teboulle 2009)

Require: Initial solution $X_1, \tau_1 \leftarrow 1$

$t \leftarrow 1, T_{\text{max}}$

repeat

Compute $W^{t+1}$ solution of $\min_{Y \in \text{Conv}(Y^k_b)} g(X_t) + \frac{1}{2\gamma} \langle X_t X_t^\top, Y^\top \rangle$

Set $Y^{t+1} = W^t + \frac{\tau_t - 1}{\tau_{t+1}} (W_t - W_{t-1})$

Compute $V^{t+1}$ solution of $\min_{X \in \mathbb{R}^{n \times m}} g(X) + \frac{1}{2\gamma} \langle XX^\top, Y_t^\top \rangle$

Set $X^{t+1} = V^t + \frac{\tau_t - 1}{\tau_{t+1}} (V_t - V_{t-1})$

Set $\tau_{t+1} = \frac{1 + \sqrt{1 + 4\tau_t^2}}{2}$

If $t \mod 20 = 0$ compute dual bound at $Y^{t+1}$ via Equation (44).

$t \leftarrow t + 1$

until $t > T_{\text{max}}$ or duality gap $\leq \epsilon$

return $X_t, Y_t$

To confirm that Algorithm 2 has indeed converged (at least approximately) to an optimal solution, we require a dual certificate. As optimizing over the set of dual variables $\alpha$ for a fixed $Y_t$ does not supply such a bound, we now invoke strong duality to derive a globally valid lower bound. Formally, we have the following result (proof deferred to Appendix A.11):

Proposition 7. Suppose that Assumption 2 holds. Then, strong duality holds between:

$$\min_{\gamma \in \text{Conv}(Y^k_b)} \max_{\alpha \in \mathbb{R}^{n \times m}} h(\alpha) - \frac{7}{2} \sum_{i=1}^n \sum_{j=1}^n Y_{i,j} \langle \alpha_i, \alpha_j \rangle, (43)$$

$$\max_{\alpha \in \mathbb{R}^{n \times m}} h(\alpha) - \text{tr}(U) - kt \quad \text{s.t.} \quad U + \mathbb{I}_t \geq \frac{\gamma}{2} \alpha \alpha^\top. (44)$$

Proposition 7 demonstrates that Problem (1)'s semidefinite relaxation is equivalent to maximizing the dual conjugate $h(\alpha)$, minus the $k$ largest eigenvalues of $\frac{\gamma}{2} \alpha \alpha^\top$. Moreover, as proven in the special
case of sparse regression by Bertsimas et al. (2019b), one can show that if the \( k \)th and \( k+1 \)th largest eigenvalues of \( \alpha \alpha^T \) in a solution to (43) are distinct then Problem (43)’s lower bound is tight. In addition, as shown by Pilanci et al. (2015) in the context of sparse regression, if Problem (1) is a matrix completion problem, \( \gamma \) is sufficiently large, and the problem data satisfies a restricted isometry condition then with high probability Problem (43)’s lower bound is nearly tight (formal proof omitted, but essentially follows in the same manner as (Pilanci et al. 2015, Theorem 2)).

4.3. Upper Bounds via Greedy Rounding

We now propose a greedy rounding method for rounding \( Y^* \), an optimal \( Y \) in a semidefinite relaxation of Problem (19), to obtain certifiably near-optimal solutions to Problem (19) quickly. Rounding schemes for approximately solving low-rank optimization problems by rounding their SDO relaxations have received a great deal of attention since they were first proposed by Goemans and Williamson (1995). Our analysis is, however, more general than typically conducted when solving low-rank problems, as it involves rounding a projection matrix \( Y \), rather than rounding \( X \), and therefore is able to generalize to the rank-\( k \) case for \( k > 1 \), which has historically been challenging.

Observe that for any feasible \( Y \in \text{Conv}(Y_n) \), \( 0 \leq \lambda_i(Y) \leq 1 \) for each eigenvalue of \( Y \), and \( Y \) is a projection matrix if and only if its eigenvalues are binary. Combining this observation with the Lipschitz continuity of \( f(Y) \) in \( Y \) suggests that high-quality feasible projection matrices can be found in the neighborhood of a solution to the semidefinite relaxation, and a good method for obtaining them is to greedily round the eigenvalues of \( Y \). Namely, let \( Y^* \) denote a solution to the semidefinite relaxation (28), \( Y^* = U \Lambda^* U^T \) be a singular value decomposition of \( Y^* \), and \( \Lambda_{\text{greedy}} \) be a diagonal matrix obtained from rounding up (to 1) \( k \) of the highest diagonal coefficients of \( \Lambda^* \), and rounding down (to 0) the \( n-k \) others. We then let \( Y_{\text{greedy}} = U \Lambda_{\text{greedy}} U^T \). We now provide guarantees on the quality of the greedily rounded solution (proof deferred to Appendix A.12):

**Theorem 7.** Let \( Y^* \) denote a solution to the semidefinite relaxation (28), \( Y^* = U \Lambda U^T \) be a singular value decomposition of \( Y^* \), \( R \) denote the indices of strictly fractional diagonal entries in \( \Lambda \), and \( \alpha^*(Y) \) denote a best choice of \( \alpha \) for a given \( Y \). Suppose that for any \( Y \in Y^n_k \), we have \( \sigma_{\text{max}}(\alpha^*(Y)) \leq L \). Then, a greedy rounding of \( Y^* \), i.e., \( Y_{\text{greedy}} = U \Lambda_{\text{greedy}} U^T \), where \( \Lambda_{\text{greedy}} \) is a diagonal matrix such that \( \Lambda_{i,i} = 1 \) for \( k \) of the highest diagonal coefficients in \( \Lambda^* \), satisfies

\[
\frac{f(Y) - f(Y^*)}{2} \leq \sum_{\beta \geq 0, ||\beta||_1 \leq 1 \in R} (\Lambda_{i,i}^* - \Lambda_{i,i}) \beta_i, \quad \text{(45)}
\]

under the Frobenius penalty and

\[
\frac{f(Y) - f(Y^*)}{2} \leq M L |R| \max_{\beta \geq 0, ||\beta||_1 \leq 1 \in R} \sum_{\beta_i} (\Lambda_{i,i}^* - \Lambda_{i,i}) \beta_i, \quad \text{(46)}
\]

for the spectral penalty. Moreover, these bounds imply that \( 0 \leq f(Y_{\text{greedy}}) - f(Y^*) \leq \epsilon \), where \( \epsilon = M L \min(|R|, n-k) \) for the spectral penalty and \( \epsilon = \frac{1}{2} \min(|R|, n-k) L^2 \) for the Frobenius penalty.
Observe that this rounding technique is robust, because it minimizes the worst-case Lipschitz upper bound, under the assumption $\sigma_{\max}(\alpha^*(Y)) \leq L$ (i.e., we have no information about which coordinate has the largest Lipschitz upper bound). For instance, under Frobenius regularization the bound is
\[
 f(Y) - f(Y^*) \leq \frac{\gamma}{2}L^2|R| \max_{\beta \geq 0: \|\beta\|_1 \leq 1} \sum_{i \in \mathcal{R}} (\Lambda^*_i - \Lambda_i, i) \beta_i ,
\]
which is minimized by solving
\[
 \min_{\Lambda \in S^k_{++}} \max_{\beta \geq 0: \|\beta\|_1 \leq 1} \frac{\gamma}{2}L^2|R| \sum_{i \in \mathcal{R}} (\Lambda^*_i - \Lambda_i, i) \beta_i ,
\]
This interpretation suggests that greedy rounding never performs too badly.

To improve the greedily rounded solution, we implement a local search strategy which obtains even higher quality warm-starts. Namely, a variant of the popular Burer-Monterio (BM) heuristic (Burer and Monteiro 2003), which seeks low-rank solutions $X$ by applying a non-linear factorization $X = UV^T$, where $U \in \mathbb{R}^{n \times l}, V \in \mathbb{R}^{m \times k}$ and iteratively optimizing over $U$ for a fixed $V$ (resp. $V$ for a fixed $U$) until convergence to a local optima occurs. This strategy improves our greedily rounded solution because we initially set $U$ to be the square root of $Y_{\text{greedy}}$ and optimize over $V$; recall that if $Y$ is a projection matrix we have $Y = UU^T$ and $X = U\Sigma V^T$ for some singular value decomposition $U, \Sigma, V^T$.

5. Numerical Experiments

In this section, we evaluate the algorithmic strategies derived in the previous section, implemented in Julia 1.3 using JuMP.jl 0.20.1, Gurobi 9.0.1 to solve the non-convex QCQOP master problems, and Mosek 9.1 to solve the conic subproblems/continuous relaxations. Except where indicated otherwise, all experiments were performed on a Intel Xeon E5—2690 v4 2.6GHz CPU core using 32 GB RAM.

We evaluate the different ingredients of our numerical strategy on a matrix completion example: First, we solve the semidefinite relaxation by implementing Algorithm 2 and demonstrate its increased scalability over Mosek’s IPM in Section 5.1. From the solution of the relaxation, our rounding and local search heuristics then provide near-optimal solutions that outperform state-of-the-art heuristic methods, as discussed in Section 5.2. By computing the gap at the root node between the lower bound from the convex relaxation and the upper bound from the rounded solution, we show in Section 5.3 that increasing the amount of regularization in Problem (1) decreases both the relative gap and the problem’s complexity. Finally, we implement Algorithm 1, benchmark its performance and, for the first time, solve low-rank matrix completion and sensor location problems to certifiable optimality in Sections 5.4 - 5.5.

7 If we had this information then, as the proof of Theorem 7 suggests, we would greedily round to one $k$ of the indices with the largest values of $L_i A^*_i, i$.

8 We remark that Gurobi solves the non-convex QCQOP master problems by translating them to piecewise linear optimization problems. Since rank constraints are not MICO representable (Lubin et al. 2019) this introduces some error. To mitigate against this error, we set the Gurobi parameters FuncPieceError and FuncPieceLength to their minimum possible values ($10^{-6}$ and $10^{-5}$ respectively). Additionally, we set NonConvex to 2, and otherwise use default Gurobi/Mosek parameters.
5.1. Exploring the Scalability of the Convex Relaxations

In this section, we explore the relative scalability of Mosek’s interior point method and Algorithm 2. We consider convex relaxations of matrix completion problems. Similarly to Candès and Plan (2010), we generate two low-rank matrices \( M_L, M_R \in \mathbb{R}^{n \times r} \) with i.i.d. \( \mathcal{N}(0, 1) \) entries, and attempt to recover the matrix \( M = M_L M_R^\top \) given a proportion \( p \) of its observations. Here, we fix \( p = 0.25 \) and \( k = r = 5 \), vary \( n \), and set \( \gamma = \frac{20}{p} \). We solve the continuous relaxation

\[
\min_{X \in \mathbb{R}^{n \times n}, Y \in \text{Conv}(Y^k)} \frac{1}{2\gamma} \text{tr}(\theta) + \sum_{(i,j) \in \Omega} (X_{i,j} - A_{i,j})^2 \quad \text{s.t.} \quad \left( \theta, X^\top X \right) \succeq 0.
\]

Table 3 reports the time required by Algorithm 2 to obtain a solution with a relative duality gap of 0.1%; see Appendix D.2 for implementation details on the algorithm. To evaluate numerical stability, we also report the relative MSE of the greedily rounded solution; small-scale experiments \((n \leq 250)\) were run on a standard MacBook pro with 16GB RAM, while larger experiments were run on the previously described cluster with 100GB RAM.

<table>
<thead>
<tr>
<th>( n )</th>
<th>Relative MSE</th>
<th>Time (s)</th>
<th>Algorithm 2</th>
<th>Relative MSE</th>
<th>Time (s)</th>
<th>( n )</th>
<th>Algorithm 2</th>
<th>Relative MSE</th>
<th>Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>0.429</td>
<td>2.28</td>
<td>0.438</td>
<td>17.28</td>
<td></td>
<td>350</td>
<td>0.058</td>
<td>6.970</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>0.138</td>
<td>47.20</td>
<td>0.139</td>
<td>79.01</td>
<td>400</td>
<td>0.056</td>
<td>8.096</td>
<td></td>
<td></td>
</tr>
<tr>
<td>150</td>
<td>0.082</td>
<td>336.1</td>
<td>0.081</td>
<td>228.7</td>
<td>450</td>
<td>0.055</td>
<td>26.350</td>
<td></td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>0.0675</td>
<td>1,906</td>
<td>0.067</td>
<td>841.7</td>
<td>500</td>
<td>0.054</td>
<td>28.920</td>
<td></td>
<td></td>
</tr>
<tr>
<td>250</td>
<td>-</td>
<td>-</td>
<td>0.062</td>
<td>1,419</td>
<td>550</td>
<td>0.0536</td>
<td>39.060</td>
<td></td>
<td></td>
</tr>
<tr>
<td>300</td>
<td>-</td>
<td>-</td>
<td>0.059</td>
<td>2.897</td>
<td>600</td>
<td>0.0525</td>
<td>38.470</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Our results demonstrate the efficiency of Algorithm 2: the relative MSE is comparable to Mosek’s, but computational time does not explode with \( n \). Since it does not require solving any SDOs and avoids the computational burden of performing the Newton step in an IPM, Algorithm 2 scales beyond \( n = 600 \) \((1,440,000 \text{ decision variables})\), compared to \( n = 200 \) for IPMs \((80,000 \text{ decision variables})\).

5.2. Numerical Evaluation of Greedy Rounding on Matrix Completion Problems

In this section, we compare the greedy rounding method with state-of-the-art heuristic methods, and demonstrate that, by combining greedy rounding with the local search heuristic of (Burer and Monteiro 2003), our approach outperforms state-of-the-art heuristic methods and therefore should be considered as a viable and efficient warm-start for Algorithm 1.

We consider the previous matrix completion problems and assess the ability to recover the low-rank matrix \( M \) (up to a relative MSE of 1%), for varying fraction of observed entries \( p \) and various \( n \).
We compare the performance of four methods: the greedy rounding method, both with and without the local improvement heuristic from Burer and Monteiro (2003), against the local improvement heuristic alone (with a thresholded-SVD initialization point) and the nuclear norm approach. Specifically, the greedy rounding method takes the solution of the previous convex relaxation with $\gamma = \frac{500}{p}$ and rounds its singular values to generate a feasible solution $Y_{\text{greedy}}$. For the local improvement heuristic, we solve:

$$\min_{X \in \mathbb{R}^{n \times n}, U, V \in \mathbb{R}^{n \times k}} \frac{1}{2\gamma} \|X\|_2^2 + \sum_{(i,j) \in \Omega} (X_{i,j} - A_{i,j})^2 \quad \text{s.t.} \quad X = UV^T,$$

for $\gamma = \frac{500}{p}$ and $k = r$, and iteratively optimize over $U$ and $V$ using MOSEK. We provide an initial value for $U$ by either taking the first $k$ left-singular vectors of a matrix $A$ where unobserved entries are replaced by 0, or taking the square root of $Y_{\text{greedy}}$. For the nuclear norm regularization strategy, since our observations are noiseless, we solve: $\min_{X \in \mathbb{R}^{n \times n}} \|X\|_* \quad \text{s.t.} \quad X_{i,j} = A_{i,j} \quad \forall (i,j) \in \Omega$.

Figure 3 depicts the proportion of times the matrix was recovered exactly (averaged over 25 samples per pair of $(n,p,r)$), while Figure 4 depicts the relative average MSE over all instances. From this set of experiments, we make several observations: First, greedy rounding and the local improvement heuristic outperform nuclear norm minimization both in terms of average relative MSE and amount of data required to recover the matrix. Second, the local improvement heuristic improves upon greedy rounding. In terms of its ability to recover the underlying matrix exactly, it performs equally well with either initialization strategy. However, initialization with the greedy rounding supplies dramatically
lower average MSEs in instances where no approach recovers the true matrix exactly. This suggests that initialization strategies for the Burer-Monterio heuristic should be revisited and greedy rounding considered as a viable and more accurate alternative than selecting a random feasible point.

### 5.3. Exploring the Impact of Regularization on Problem Complexity

We now examine the impact of the regularization term $\frac{1}{2\gamma}\|X\|_F^2$ on the problem complexity, as captured by the relative in-sample duality gap between the semidefinite relaxation and the objective value of the greedy solution with a BM local improvement heuristic. We generate the problem data in the same manner as the previous experiment, and display results for four values of $\gamma$ in Figure 5. Observe that as $\gamma$ increases, both the duality gap and the problem’s complexity increase—this also applies for the Algorithm 1’s runtimes; see Section 5.4. This observation confirms similar results on the impact of regularization in mixed-integer conic optimization problems (c.f. Bertsimas et al. 2019a). Additionally, when $\gamma = \frac{500}{p}$ in Figure 5(d), the region where the in-sample duality gap is zero corresponds to cases where we exactly recover the underlying matrix with high probability, while a strictly positive duality gap corresponds to instances with partial recovery only (see Figure 3). This suggests a deep connection between tightness of the relaxation and statistical recovery.

### 5.4. Benchmarking Algorithm 1 on Synthetic Matrix Completion Problems

We now benchmark the performance of Algorithm 1 on medium-scale matrix completion problems.
We first compare the two different implementations of Algorithm 1, single- and multi-tree, with solving the problem directly as a QCQOP in Gurobi 9.0 (Section 4.1.2). In Algorithm 1, the lower bounds are warm-started with 200 cuts from the in-out method, and greedy rounding with local search improvement is used for the upper bounds. At the $t$th iteration, we impose a time limit of $10t$ seconds for generating the new cut so as to increase numerical precision as the solver progresses. We also impose a limit of 20 cuts for the multi-tree approach, a time limit of 3600s for the single-tree approach and Gurobi, and an optimality gap of 1\%\(^9\). Average runtime, number of nodes, and optimality gap are reported in Table 4. We observe that Algorithm 1 drastically improves upon Gurobi both in terms of computational time (reduced by up to two orders of magnitude) and accuracy (gap reduced from over 100\% down to 2\% or less). Single- and multi-tree implementations are comparable in terms of computational time, although multi-tree finds high-quality feasible solutions earlier than single tree and accepts our warm-start more consistently, which suggests that it may scale better to high-dimensional settings. Note that “nodes” refers to the number of nodes in the terminal branch-and-bound tree for the multi-tree implementation.

Next, we evaluate the performance of the multi-tree implementation of Algorithm 1 on a more extensive test-set, including instances where $\text{Rank}(M) > 1$, in Table 5. Note that when $r = 1$ we use the same experimental setup (although we impose a time limit of $30t$ seconds, or 7200 seconds if there

\(^9\) We report the gap between the better of Gurobi’s lower bound and the semidefinite lower bound, compare to the objective value which we evaluate directly; this is sometimes $1 - 2\%$ even when Gurobi reports that it has found an optimal solution, due to numerical instability in Gurobi.
Moreover, when \( n > 20 \), and when \( r > 1 \) we increase the time limit per iteration to 300t seconds (or 7200 seconds if there has been no improvement for two consecutive iterations), and allow up to 100 PSD cuts per iteration to be added at the root node via a user cut callback, in order to strengthen the approximation of the PSD constraint \( Y \succeq 0 \). We observe that the problem’s complexity increases with the rank, although not too excessively. Moreover, when \( r > 1 \) the bound gap is actually smaller when \( \gamma = \frac{100}{p} \) than when \( \gamma = \frac{20}{p} \). We believe this is because Gurobi cannot represent the SDO constraint \( Y \succeq 0 \) and its SOC approximation is inexact (even with PSD cuts), and in some cases refining this approximation is actually harder than refining our approximation of \( g(X) \).

Note that the main bottleneck inhibiting solving larger-scale matrix completion problems is Gurobi itself, as the non-convex solver takes increasing amounts of time to process warm-starts (sometimes in the 100s or 1000s of seconds) when \( n \) increases. We believe this may be because of the way Gurobi translates orthogonal projection matrices to a piecewise linear formulation. Encouragingly, this suggests that our approach may successfully scale to 100 \times 100\) matrices as Gurobi improves their solver.

Finally, we compare the solution from the exact formulation (19) solved using Algorithm 1 (multi-tree) with the initial warm-start we proposed and two state-of-the-art heuristics, namely nuclear norm minimization and the Burer-Monterio approach, as in Section 5.2. Here, we take \( n \in \{25, 50\} \), \( r = 1 \), \( p \) ranging from 0 to 0.4, and \( \gamma = \frac{100}{p} \). Figure 6 depicts the average relative MSE over the entire matrix, averaged over 25 random instances per value of \( p \). When \( p \geq 0.2 \), the exact method supplies an out-of-sample relative MSE around 0.6% lower than Burer-Monterio\(^{10}\).

\(^{10}\) Because we ran all methods on the same random instances, this difference is statistically significant, with a p-value of \( 2 \times 10^{-51} \) (resp. \( 2 \times 10^{-129} \)) that the relative MSE is lower for the exact method when \( n = 25 \) (resp. \( n = 50 \)).
Table 5  Scalability of Algorithm 1 (multi-tree) for solving low-rank matrix completion problems to certifiable optimality, averaged over 20 random matrices per row.

<table>
<thead>
<tr>
<th>$n$</th>
<th>$p$</th>
<th>$\gamma$</th>
<th>Rank-1 Time(s)</th>
<th>Rank-1 Nodes</th>
<th>Rank-1 Gap (%)</th>
<th>Rank-1 Cuts</th>
<th>Rank-2 Time(s)</th>
<th>Rank-2 Nodes</th>
<th>Rank-2 Gap (%)</th>
<th>Rank-2 Cuts</th>
<th>Rank-3 Time(s)</th>
<th>Rank-3 Nodes</th>
<th>Rank-3 Gap (%)</th>
<th>Rank-3 Cuts</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.1</td>
<td>20/p</td>
<td>182.1</td>
<td>9,755</td>
<td>0.05%</td>
<td>2.56</td>
<td>24,220</td>
<td>35,670</td>
<td>0.34%</td>
<td>5.78</td>
<td>37,780</td>
<td>39,870</td>
<td>0.71%</td>
<td>9.28</td>
</tr>
<tr>
<td>10</td>
<td>0.2</td>
<td>20/p</td>
<td>3,508</td>
<td>21,060</td>
<td>0.26%</td>
<td>10.8</td>
<td>209,900</td>
<td>108,000</td>
<td>2.52%</td>
<td>35.3</td>
<td>135,260</td>
<td>35,870</td>
<td>3.31%</td>
<td>26.2</td>
</tr>
<tr>
<td>10</td>
<td>0.3</td>
<td>20/p</td>
<td>5,488</td>
<td>30,970</td>
<td>0.39%</td>
<td>13.1</td>
<td>302,200</td>
<td>70,500</td>
<td>8.66%</td>
<td>50.0</td>
<td>302,100</td>
<td>31,870</td>
<td>19.7%</td>
<td>50.0</td>
</tr>
<tr>
<td>10</td>
<td>0.1</td>
<td>100/p</td>
<td>656.5</td>
<td>28,870</td>
<td>0.01%</td>
<td>2.14</td>
<td>676.1</td>
<td>25,493</td>
<td>0.09%</td>
<td>1.83</td>
<td>842.7</td>
<td>20,700</td>
<td>0.24%</td>
<td>1.79</td>
</tr>
<tr>
<td>10</td>
<td>0.2</td>
<td>100/p</td>
<td>1,107</td>
<td>10,010</td>
<td>0.09%</td>
<td>4.29</td>
<td>2,065</td>
<td>42,490</td>
<td>0.19%</td>
<td>5.61</td>
<td>57,530</td>
<td>36,910</td>
<td>1.24%</td>
<td>10.7</td>
</tr>
<tr>
<td>10</td>
<td>0.3</td>
<td>100/p</td>
<td>3,364</td>
<td>48,730</td>
<td>0.22%</td>
<td>6.30</td>
<td>272,300</td>
<td>33,150</td>
<td>1.95%</td>
<td>44.7</td>
<td>249,700</td>
<td>35,530</td>
<td>4.99%</td>
<td>42.2</td>
</tr>
<tr>
<td>20</td>
<td>0.1</td>
<td>20/p</td>
<td>2,017</td>
<td>4,756</td>
<td>0.61%</td>
<td>8.20</td>
<td>253,900</td>
<td>8,030</td>
<td>2.79%</td>
<td>42.7</td>
<td>255,400</td>
<td>3,015</td>
<td>3.09%</td>
<td>43.2</td>
</tr>
<tr>
<td>20</td>
<td>0.2</td>
<td>20/p</td>
<td>6,369</td>
<td>6,636</td>
<td>1.36%</td>
<td>15.0</td>
<td>298,700</td>
<td>3,342</td>
<td>54.9%</td>
<td>50.0</td>
<td>295,500</td>
<td>236.5</td>
<td>87.9%</td>
<td>50.0</td>
</tr>
<tr>
<td>20</td>
<td>0.3</td>
<td>20/p</td>
<td>6,687</td>
<td>4,187</td>
<td>0.82%</td>
<td>18.4</td>
<td>296,500</td>
<td>3,175</td>
<td>112.3%</td>
<td>50.0</td>
<td>291,100</td>
<td>41.35</td>
<td>214.7%</td>
<td>50.0</td>
</tr>
<tr>
<td>20</td>
<td>0.1</td>
<td>100/p</td>
<td>1,266</td>
<td>8,792</td>
<td>0.87%</td>
<td>8.35</td>
<td>211,700</td>
<td>6,860</td>
<td>0.73%</td>
<td>34.24</td>
<td>171,900</td>
<td>2,350</td>
<td>1.31%</td>
<td>29.8</td>
</tr>
<tr>
<td>20</td>
<td>0.2</td>
<td>100/p</td>
<td>1,220</td>
<td>2,710</td>
<td>1.04%</td>
<td>7.80</td>
<td>302,800</td>
<td>2,426</td>
<td>12.3%</td>
<td>50.0</td>
<td>298,800</td>
<td>221.4</td>
<td>12.3%</td>
<td>50.0</td>
</tr>
<tr>
<td>20</td>
<td>0.3</td>
<td>100/p</td>
<td>1,272</td>
<td>1,837</td>
<td>0.64%</td>
<td>3.14</td>
<td>299,000</td>
<td>2,518</td>
<td>26.4%</td>
<td>50.0</td>
<td>293,500</td>
<td>43.0</td>
<td>65.9%</td>
<td>50.0</td>
</tr>
<tr>
<td>30</td>
<td>0.1</td>
<td>20/p</td>
<td>300,300</td>
<td>2,735</td>
<td>9.05%</td>
<td>50.0</td>
<td>304,300</td>
<td>164.0</td>
<td>79.0%</td>
<td>50.0</td>
<td>303,100</td>
<td>1.10</td>
<td>36.5%</td>
<td>50.0</td>
</tr>
<tr>
<td>30</td>
<td>0.2</td>
<td>20/p</td>
<td>298,700</td>
<td>1,511</td>
<td>13.6%</td>
<td>50.0</td>
<td>301,700</td>
<td>9.62</td>
<td>310.5%</td>
<td>50.0</td>
<td>302,600</td>
<td>1.00</td>
<td>558.1%</td>
<td>50.0</td>
</tr>
<tr>
<td>30</td>
<td>0.3</td>
<td>20/p</td>
<td>183,800</td>
<td>1,743</td>
<td>4.76%</td>
<td>36.9</td>
<td>303,000</td>
<td>1,633</td>
<td>523.2%</td>
<td>50.0</td>
<td>305,000</td>
<td>0.70</td>
<td>1,460%</td>
<td>50.0</td>
</tr>
<tr>
<td>30</td>
<td>0.1</td>
<td>100/p</td>
<td>305,600</td>
<td>2,262</td>
<td>2.73%</td>
<td>50.0</td>
<td>302,800</td>
<td>97.40</td>
<td>9.73%</td>
<td>50.0</td>
<td>305,000</td>
<td>1.90</td>
<td>9.67%</td>
<td>50.0</td>
</tr>
<tr>
<td>30</td>
<td>0.2</td>
<td>100/p</td>
<td>246,300</td>
<td>3,285</td>
<td>3.15%</td>
<td>43.6</td>
<td>304,300</td>
<td>6.17</td>
<td>69.7%</td>
<td>50.0</td>
<td>302,600</td>
<td>1.00</td>
<td>141.9%</td>
<td>50.0</td>
</tr>
<tr>
<td>30</td>
<td>0.3</td>
<td>100/p</td>
<td>25,970</td>
<td>11,020</td>
<td>0.89%</td>
<td>17.1</td>
<td>304,000</td>
<td>1.00</td>
<td>92.3%</td>
<td>50.0</td>
<td>304,700</td>
<td>1.00</td>
<td>322.1%</td>
<td>50.0</td>
</tr>
</tbody>
</table>

Figure 6  Average relative MSE for nuclear norm (NN), greedy rounding (GD), Burer-Monterio (BM), and outer-approximation (OA) when imputing a rank-1 $n \times n$ matrix. All results are averaged over 25 matrices.

5.5. Benchmarking Algorithm 1 on Synthetic Coordinate Recovery Problems

We now benchmark the performance of Algorithm 1 on anchor-free synthetic coordinate recovery problems, as previously studied by Biswas and Ye (2004), Luo et al. (2010) among others.

Specifically, we sample $n$ coordinates $x_i$ uniformly over $[-0.5,0.5]^k$ for $k \in \{2,3\}$, and attempt to recover a noisy Gram matrix $G \in S_n^+$ of the $x_i$’s, given a subset of observations of the underlying matrix. Similarly to Biswas and Ye (2004), we supply the distance between the points $D_{i,j} = \|x_i - x_j\|_2^2 + z$, where $z \sim \mathcal{N}(0,0.01)$, if and only if the radio range between the two points is such that $D_{i,j} \leq D_{\text{radio}}$.

Note that we solve these problems in precisely the same fashion as the largest-scale matrix completion problems solved in the previous section (multi-tree, with a limit of 50 cut passes etc.)
Formally, in order to account for noise in the observed entries, we solve the following problem:

$$\min_{Y \in \mathcal{Y}} \min_{G \in S_n^k} \frac{1}{2\gamma} \|G\|_F^2 + \text{tr}(G) + \lambda \cdot \|\xi\|_1 \quad \text{s.t.} \quad G_{i,i} + G_{j,j} - 2G_{i,j} + \xi_{i,j} = D_{i,j} \forall (i,j) \in \Omega, \ G = Y^G,$$

where $\lambda > 0$ is a penalty term which encourages robustness, and the Frobenius norm objective likewise encourages robustness against noise in $G$. The performance of Algorithm 1 (multi-tree) on various synthetic instances is reported in Table 6, for $\gamma, n, d_{\text{radio}}, k$ varying.

### Table 6

Scalability of Algorithm 1 (multi-tree) for solving sensor location problems to certifiable optimality, averaged over 20 random instances per row. A “-” denotes an instance that cannot be solved within the time budget, because Gurobi fails to accept our warm-start and cannot find a feasible solution. We let $\lambda = n^2$ for all instances.

<table>
<thead>
<tr>
<th>$n$</th>
<th>$d_{\text{radio}}$</th>
<th>$\gamma$</th>
<th>Rank-2</th>
<th>Rank-3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Time(s)</td>
<td>Nodes</td>
</tr>
<tr>
<td>10</td>
<td>0.1</td>
<td>1/p</td>
<td>135.3</td>
<td>6,926</td>
</tr>
<tr>
<td>10</td>
<td>0.2</td>
<td>1/p</td>
<td>3,189</td>
<td>5,249</td>
</tr>
<tr>
<td>10</td>
<td>0.1</td>
<td>100/p</td>
<td>76.2</td>
<td>1,155</td>
</tr>
<tr>
<td>10</td>
<td>0.2</td>
<td>100/p</td>
<td>480.6</td>
<td>0.05</td>
</tr>
<tr>
<td>20</td>
<td>0.1</td>
<td>1/p</td>
<td>3,475</td>
<td>4,548</td>
</tr>
<tr>
<td>20</td>
<td>0.2</td>
<td>1/p</td>
<td>73,000</td>
<td>0.50</td>
</tr>
<tr>
<td>20</td>
<td>0.1</td>
<td>100/p</td>
<td>1,878</td>
<td>0.00</td>
</tr>
<tr>
<td>20</td>
<td>0.2</td>
<td>100/p</td>
<td>67,530</td>
<td>0.20</td>
</tr>
</tbody>
</table>

We observe that the problem’s complexity increases with the rank and with the dimensionality of the Gram matrix, although not too excessively. Indeed, Algorithm 1 can solve coordinate recovery problems with tens of data points to certifiable optimality in hours.

### 5.6. Summary of Findings from Numerical Experiments

Our main findings from the numerical experiments in this section are as follows:

- As demonstrated in Section 5.1, Algorithm 2 successfully solves convex relaxations of low-rank problems where $n = 100s$, in a faster and more scalable fashion than state-of-the-art interior point codes such as Mosek.

- As demonstrated in Section 5.2, the following strategy is almost as effective as solving a low-rank problem exactly: solving the convex relaxation using Algorithm 2, greedily rounding the solution to the convex relaxation, and using this greedily rounded solution as a warm-start for a local method such as the method of Burer and Monteiro (2003, 2005). The success of this strategy can be explained because solving a relaxation and rounding is a global strategy which matches the low-order moments of the set of optimal solutions to obtain a solution near an optimal solution, while local methods polish a solution by seeking the best solution within some neighborhood of an initial point.

- As demonstrated in Section 5.3, increasing the amount of regularization in a low-rank problem by decreasing $\gamma$ or $M$ decreases the duality gap between a low-rank problem with Frobenius or spectral...
norm problem, and its convex relaxation. Therefore, increasing the amount of regularization makes the problem easier in a practical sense (although not necessarily in a complexity-theoretic sense).

- As demonstrated in Sections 5.4-5.5, Algorithm 1 scales to solve problems where $n$ is in the tens, i.e., hundreds or thousands of decision variables, in hours. Moreover, the main bottleneck inhibiting solving problems where $n$ is in the hundreds or thousands is that we solve our master problems using Gurobi, a QCQOP solver which translates the orthogonal projection matrix constraint into many piecewise linear constraints. This suggests that a custom branch-and-bound solver which explicitly models orthogonal projection matrices constitutes a promising area for future work.

6. Conclusion

In this paper, we introduced Mixed-Projection Conic Optimization, a new framework for modeling rank constrained optimization problems that, for the first time, solves low-rank problems to certifiable optimality at moderate problem sizes. We also provided a characterization of the complexity of rank constraints, and proposed new convex relaxations and rounding methods that lead to viable and more accurate solutions than those obtained via existing techniques such as the log-det or nuclear norm heuristics. Inspired by the collective successes achieved in mixed-integer optimization, we hope that MPCO constitutes an exciting new research direction for the optimization community. For instance, we believe that custom branch-and-bound solvers that explicitly model orthogonal projection matrices could further enhance the scalability of the MPCO framework.

References


A. Omitted Proofs

In this section, we supply omitted proofs of the results stated in the manuscript, in the order in which the results were stated.

A.1. Proof of Theorem 1

Proof of Theorem 1  By (Blekherman et al. 2012, Theorem 2.49), a set of distances $d_{i,j}$ can be embedded in a Euclidean space of dimension $k$ if and only if there exists some Gram matrix $G$ such that $D = \text{Diag}(G)e^T + e\text{Diag}(G)^T - 2G$ where $G \succeq 0$ and $\text{Rank}(G) \leq k$. Therefore, Proposition 1’s $\exists \mathbb{R}$-hard problem is a special case of:

$$\min_{G \in S_n^+} \text{Rank}(G) \quad \text{s.t.} \quad \text{Diag}(G)e^T + e\text{Diag}(G)^T - 2G = D,$$

where $k = 2$, $D_{i,j} = 1 \forall (i,j) \in E$, and we do not impose the $(i,j)$th equality otherwise. □

A.2. Proof of Theorem 2

Proof of Theorem 2  To establish this result, it suffices to perform a reduction from the following feasibility system to a polynomially sized system of polynomial equalities and inequalities:

$$\exists X : \text{Rank}(X) \leq k, \quad \langle A_i, X \rangle = b_i \quad \forall i \in [m], \quad X \succeq 0,$$

because testing the feasibility of the above system a polynomial number of times for $k \in [n]$ certainly solves Problem (19).

By Proposition 2, this system is feasible if and only if the following system is also feasible:

$$\exists X, Y : \quad Y^2 = Y, \quad X = XY, \quad \text{tr}(Y) \leq k, \quad \langle A_i, X \rangle = b_i \quad \forall i \in [m], \quad X \succeq 0,$$

The result then follows by observing that semidefinite constraints are indeed semialgebraic constraints, by the Tarski-Seidenberg theorem (see Blekherman et al. 2012, Chapter A.4.4), and therefore this system is equivalent to a polynomially sized system of polynomial equalities and inequalities. □

A.3. Complexity of Low-Rank Integer Optimization and Hilbert’s 10th Problem

We now demonstrate that imposing the constraint $X \in \mathbb{Z}^{n \times n}$ changes Problem (1)’s complexity status, by making it undecidable. Formally, we have:

**Theorem 8.** The following problem is undecidable, even when $k = 1$ and its objective is binary:

$$\min_{X \in \mathbb{Z}^{n \times n}} \quad \langle C, X \rangle \quad \text{s.t.} \quad \langle A_i, X \rangle = b_i \quad \forall i \in [m], \quad \text{Rank}(X) \leq k, \quad X \succeq 0.$$
Remark 5. Theorem 8’s reduction does not hold in the presence of a regularizer. Indeed, imposing either a spectral or Frobenius norm regularizer ensures the boundedness of the problem’s level sets, which allows the problem to be solved in finite time via branch-and-bound. Nonetheless, Theorem 8 shows that imposing an integrality constraint makes Problem (1) much harder.

Proof of Theorem 8 We perform a reduction from integer optimization with quadratic constraints, which is undecidable when the objective is binary (Jeroslow 1973), by reduction from Hilbert’s 10th problem. Recall that integer optimization with quadratic constraints is definitionally:

\[ \min_{x \in \mathbb{Z}^n} \langle c, x \rangle \quad \text{s.t.} \quad \langle Q_i, x \rangle + \langle a_i, x \rangle \leq b_i \quad \forall i \in [m]. \] (50)

We now show that this problem is equivalent to:

\[ \min_{x \in \mathbb{Z}^n, X \in \mathbb{Z}^{n \times n}} \langle c, x \rangle \quad \text{s.t.} \quad \langle Q_i, X \rangle + \langle a_i, x \rangle \leq b_i \quad \forall i \in [m], \quad \text{Rank} \left( \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix} \right) \leq 1, \quad \left( \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix} \right) \succeq 0. \]

To establish the result, it suffices to show that \( X = xx^T \) in any feasible solution to the second problem. We now show this, by appealing to the Guttman rank identity (see Lemma B.2). We remind the reader that an equivalent form of the Guttman rank formula is the identity:

\[ \text{Rank} \left( \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix} \right) = \text{Rank}(X) + \text{Rank}(X - xx^T), \]

Since \( X \succeq 0 \) and the left hand side of the above expression is either 0 or 1, this formula implies that either \( X = xx^T \) or \( X = 0 \). However, the latter case can only hold if \( x = 0 \), since \( X \succeq xx^T \) by Schur complements. Therefore, \( X = xx^T \) and the result holds. \( \square \)

A.4. Intermediate Lemmas to Derive Saddle-Point Problem

In this section, we derive the conjugates of the penalties stated in Table 2, in order to complete our proof of Theorem 3. We first derive our results for rectangular matrices under the formulation \( X = YXY' \) for appropriate projection matrices \( Y, Y' \), before specializing our results by setting \( Y' = I \). For completeness, we first prove that this reformulation is indeed a valid reformulation of a rank constraint.

Proposition 8. For any \( X \in \mathbb{R}^{n \times m} \), \( \text{Rank}(X) \leq k \iff \exists Y \in \mathcal{Y}_n, Y' \in \mathcal{Y}_m \text{ s.t. } \text{tr}(Y), \text{tr}(Y') \leq k \) and \( X = YXY' \), where \( \mathcal{Y}_n := \{ P \in S^n : P^2 = P \} \) is the set of projection matrices.

Proof of Proposition 8 We prove the two implications successively.

( \( \implies \) ) Let \( X = U\Sigma V^T \) be a singular value decomposition of \( X \). Since \( \text{Rank}(X) \leq k \), we can let \( U \in \mathbb{R}^{n \times k}, \Sigma \in \mathbb{R}^{k \times k}, V \in \mathbb{R}^{k \times m} \) without loss of generality. Define \( Y = U(U^TU)^{-1}U \) and \( Y' = V(V^TV)^{-1}V \). By construction, \( YXY' = X \). In addition, \( \text{tr}(Y) = \text{rank}(Y) = \text{rank}((U)) \leq k \), and \( \text{tr}(Y') = \text{rank}(Y') = \text{rank}((V)) \leq k \).

( \( \iff \) ) Since \( X = YXY' \), \( \text{rank}(X) \leq \min(\text{rank}(Y), \text{rank}(Y')) \leq \min(\text{tr}(Y), \text{tr}(Y')) \leq k \). \( \square \)
A.4.1. Rectangular Case  In this section, we derive a dual reformulation for the conjugate of the regularization term in (19). More precisely, for all regularizers of interest $\Omega(\cdot)$, we show that for any matrix $\alpha$ of similar dimension as $X$ and any projection matrices $Y \in \mathcal{Y}_n^k$ and $Y' \in \mathcal{Y}_m^k,$

$$\min_{X} [\Omega(YXY') + \langle \alpha, YXY' \rangle] = \max_{\nu_1, \nu_2} -\Omega' (\alpha, Y, Y', V_{11}, V_{22}),$$

where $\Omega'^{\nu}(\cdot)$ is notably linear in $Y, Y'.$

**Lemma A.1** Let $A \in \mathbb{R}^{n \times m}$ be a rectangular matrix, $Y \in \mathcal{Y}_n, Y' \in \mathcal{Y}_m$ be projection matrices and $\gamma > 0$ be a positive scalar. Then

$$\min_{X \in \mathbb{R}^{n \times m}} \frac{1}{2\gamma} \|YXY'\|^2 + \langle A, YXY' \rangle = -\frac{\gamma}{2} \text{tr}(YAY' A^\top).$$

**Proof of Lemma A.1** Any solution to the minimization problem satisfies the first-order condition

$$\frac{1}{\gamma} YXY' + YAY' = 0.$$ Hence, $X^\ast = -\gamma YAY'$ is optimal and the objective is $-\frac{\gamma}{2} \text{tr}(YAY'A^\top).$ \hfill $\Box$

**Lemma A.2** Let $A \in \mathbb{R}^{n \times m}$ be a rectangular matrix, $Y \in \mathcal{Y}_n, Y' \in \mathcal{Y}_m$ be projection matrices and $M > 0$ be a positive scalar. Then

$$\min_{X \in \mathbb{R}^{n \times m} : \|YXY'\|_\sigma \leq M} \langle A, YXY' \rangle = \max_{\nu_1, \nu_2} -\frac{M}{2} \langle V_{11}, Y \rangle - \frac{M}{2} \langle V_{22}, Y' \rangle$$

$$\text{s.t. } \begin{pmatrix} V_{11} & A \\ A^\top & V_{22} \end{pmatrix} \succeq 0,$$

**Proof of Lemma A.2** Since the trace is invariant by cyclic permutation and the matrices $Y$ and $Y'$ are symmetric,

$$\min_{X \in \mathbb{R}^{n \times m} : \|YXY'\|_\sigma \leq M} \langle A, YXY' \rangle = \min_{X \in \mathbb{R}^{n \times m} : \|YXY'\|_\sigma \leq M} \langle YAY', X \rangle.$$ The spectral norm penalty is equivalent to

$$\|YXY'\|_\sigma \leq M \iff YXY'Y'X^\top Y \preceq M^2 I_n$$

$$\iff (YXY')Y'(Y'X^\top Y) \preceq M^2 I_n$$

$$\iff (YXY')Y'(Y'X^\top Y) \preceq M^2 Y',$$

where the last inequality follows from the fact that the matrix on the left-hand side is equal to zero over $\text{Im}(Y)^\top$. By Schur complements, the final semidefinite inequality is equivalent to

$$\begin{pmatrix} M Y & YXY' \\ Y'X^\top Y & M Y' \end{pmatrix} \succeq 0.$$\[ \begin{pmatrix} W_{11} & W_{12} \\ W_{12}^\top & W_{22} \end{pmatrix} \succeq 0, \]

We associate a matrix of dual variables in square brackets. Therefore, the dual problem is

$$\max_{W_{11}, W_{12}, W_{22}} -M \langle W_{11}, Y \rangle - M \langle W_{22}, Y' \rangle$$

$$\text{s.t. } YAY' = 2YW_{12}Y', \begin{pmatrix} W_{11} & W_{12} \\ W_{12}^\top & W_{22} \end{pmatrix} \succeq 0,$$

or equivalently, $\max_{\nu_1, \nu_2} -\frac{M}{2} \langle V_{11}, Y \rangle - \frac{M}{2} \langle V_{22}, Y' \rangle$ \[ \text{s.t. } \begin{pmatrix} V_{11} & A \\ A^\top & V_{22} \end{pmatrix} \succeq 0. \] \hfill $\Box$
We now provide a simplified statement of our result in the special case where \( Y' = I_m \).

**Lemma A.3** Suppose that Assumption 2 holds and \( \Omega(\cdot) \) is either the spectral or Frobenius regularizer. Then, the following two optimization problems are equivalent:

\[
\begin{align*}
\min_{Y \in \mathcal{Y}_n} & \quad \min_{X \in \mathbb{R}^{n \times m}} \lambda \cdot \text{tr}(Y) + g(X) + \Omega(X) \quad \text{s.t.} \quad X = YX, \\
\min_{Y \in \mathcal{Y}_n} & \quad \max_{\alpha, V_{11}, V_{22}} \lambda \cdot \text{tr}(Y) + h(\alpha) - \Omega^*(\alpha, Y, V_{11}, V_{22}),
\end{align*}
\]

where \( h(\alpha) := \inf_V g(V) - \langle V, \alpha \rangle \) is, up to a sign, the Fenchel conjugate of \( g \) (see Boyd and Vandenberghe 2004, Chap. 3.3) and

\[
\Omega^*(\alpha, Y, V_{11}, V_{22}) = \frac{\gamma}{2} \text{tr}(Y \alpha \alpha^\top),
\]

under Frobenius norm regularization.

**A.4.2. Square and Symmetric Case** When the matrix \( X \) is square and symmetric we can take \( Y = Y' \) and obtain

**Proposition 9.** Suppose that Assumption 2 holds and \( \Omega(\cdot) \) is either the spectral or Frobenius regularizer. Then, the following two optimization problems are equivalent:

\[
\begin{align*}
\min_{Y \in \mathcal{Y}_n} & \quad \min_{X \in \mathbb{R}^{n \times m}} \lambda \cdot \text{tr}(Y) + g(X) + \Omega(X) \quad \text{s.t.} \quad X = YXY, \\
\min_{Y \in \mathcal{Y}_n} & \quad \max_{\alpha, V_{11}, V_{22}} \lambda \cdot \text{tr}(Y) + h(\alpha) - \Omega^*(\alpha, Y, V_{11}, V_{22}),
\end{align*}
\]

where \( h(\alpha) := \inf_V g(V) - \langle V, \alpha \rangle \) is, up to a sign, the Fenchel conjugate of \( g \) (see Boyd and Vandenberghe 2004, Chap. 3.3) and

\[
\Omega^*(\alpha, Y, V_{11}, V_{22}) = \frac{\gamma}{2} \text{tr}(Y \alpha \alpha^\top),
\]

under Frobenius norm regularization, where the last inequality holds because \( Y \alpha \) is a symmetric matrix, since from the primal-dual optimality condition we have \( Y \alpha = \frac{-1}{\gamma} X \in S^n \).

For the spectral norm penalty, we can further simplify

**Lemma A.4** Let \( A \in \mathbb{R}^{n \times n} \) be a square symmetric matrix and \( Y \in \mathcal{Y}_n \) be a projection matrix. Then

\[
\begin{align*}
\min_{X \in \mathbb{S}_n: \|YXY\|_\sigma \leq M} \langle A, YXY \rangle = & \max_{V_+, V_- \geq 0} -M\langle V_+ + V_-, Y \rangle \quad \text{s.t.} \quad A = V_+ - V_-.
\end{align*}
\]
**Proof of Lemma A.4** The constraint \( \|YXY\|_2 \leq M \) can be rewritten as \(-MY \preceq YXY \preceq MY\). By strong semidefinite duality (which holds as the minimization problem has non-empty interior; see (Wolkowicz et al. 2012, Chapter 4.1)):

\[
\min_{X \in S_\alpha} \|YXY\|_2 \leq M \quad \exists A = \max_{W_+, W_- \geq 0} -M(Y, W_+ + W_-) \quad \text{s.t.} \quad YAY = YW_+Y - YW_-Y.
\]

The decision variables in the maximization problem decompose the symmetric matrix \(YAY\) into a positive and negative definite parts, \(W_+\) and \(W_-\) respectively. Without loss of optimality we can restrict our attention to \(W_+ = YV_+Y\) and \(W_- = YV_-Y\) where \(A = V_+ - V_-\). \(\square\)

**A.5. Proof of Theorem 4**

**Proof of Theorem 4** Let us fix \(Y \in \text{Conv}(\mathcal{Y}_n)\). Then, we have that:

\[
\max_{\alpha} h(\alpha) - \gamma \frac{1}{2} \langle \alpha, YA \rangle = \max_{\alpha, \beta} h(\alpha) - \gamma \frac{1}{2} \langle \beta, YA \rangle \quad \text{s.t.} \quad \beta = \alpha,
\]

\[
= \max_{\alpha} \min_{X} h(\alpha) - \gamma \frac{1}{2} \langle \beta, YX \rangle - \langle X, \beta - \alpha \rangle,
\]

\[
= \min_{X} \max_{\alpha} \frac{\left[ h(\alpha) + \langle X, \alpha \rangle \right]}{\alpha} \left[ \left( -h(\alpha) \right) = g(X) \right].
\]

Finally, the optimality condition with respect to \(\beta\) is \(Y\beta = \frac{1}{\gamma} X\), which implies the later term is:

\[
\max_{W} \left[ \frac{1}{2\gamma} \langle X, Y^\dagger X \rangle - \frac{1}{2} \langle X, (I - Y^\dagger Y)W \rangle \right] = \max_{W} \left[ \frac{1}{2\gamma} \langle X, Y^\dagger X \rangle - \frac{1}{2} \langle W, (I - Y^\dagger Y)X \rangle \right]
\]

\[
= \begin{cases} 
\frac{1}{2\gamma} \langle X, Y^\dagger X \rangle & \text{if } Y \in \text{Span}(X), \\
+\infty & \text{otherwise}.
\end{cases}
\]

We therefore conclude that the later term is equal to \(\frac{1}{2\gamma} \langle X, Y^\dagger X \rangle\) whenever the constraint \(Y^\dagger YX = X\) holds. By the generalized Schur complement lemma B.1, this expression is equivalent to introducing a new matrix \(\theta\), imposing the term \(\frac{1}{2\gamma} \text{tr}(\theta)\) and requiring that \(\left( \theta X^\dagger Y \right) \succeq 0\). \(\square\)

**A.6. Proof of Theorem 5**

**Proof of Theorem 5** Let us fix \(Y \in \text{Conv}(\mathcal{Y}_n)\). Then, we have that:

\[
\max_{\alpha \in S_\alpha} \min_{W_+, W_- \geq 0} h(\alpha) - M \langle Y, W_+ - W_- \rangle \quad \text{s.t.} \quad \alpha = W_+ - W_- \]

\[
= \max_{\alpha \in S_\alpha} \min_{W_+, W_- \geq 0} \max_{X \in S_\alpha} h(\alpha) - M \langle Y, W_+ - W_- \rangle + \langle X, \alpha - W_+ + W_- \rangle
\]

\[
= \min_{X \in S_\alpha} \max_{\alpha \in S_\alpha} \max_{W_+, W_- \geq 0} h(\alpha) - M \langle Y, W_+ - W_- \rangle + \langle X, \alpha - W_+ + W_- \rangle
\]

\[
= \min_{X \in S_\alpha} \max_{\alpha \in S_\alpha} \frac{\left[ h(\alpha) + \langle X, \alpha \rangle \right]}{\alpha} \left[ \left( -h(\alpha) \right) = g(X) \right] \max_{W_+, W_- \geq 0} \left[ -M \langle Y, W_+ - W_- \rangle + \langle X, -W_+ + W_- \rangle \right].
\]

Finally, the optimality conditions with respect to \(W_+, W_-\) imply that \(-MY \preceq X \preceq MY\). \(\square\)
A.7. Proof of Propositions 4 and 5

Proof of Proposition 4  In Problem (35), it is not too hard to see that for any $X$ an optimal choice of $Y$ is $Y = \frac{1}{M}X_+ + \frac{1}{M}X_-$, where $X_+, X_-$ are orthogonal positive semidefinite matrices such that $X = X_+ - X_-$. Therefore, the result follows by observing that $\text{tr}(X_+ + X_-) = \|X\|_\ast$. □

Proof of Proposition 5  In Problem (37), for any feasible $X$ we have $\|X\|_\ast \leq M$. Under this constraint, it follows that for any $X$ an optimal choice of $Y, Y'$ is $Y = U\Sigma V^\top, Y' = V\Sigma V^\top$, where $X = U\Sigma V^\top$ is an SVD of $X$. The result follows as $\text{tr}(Y) = \text{tr}(\Sigma) = \|X\|_\ast$. □

A.8. Proof of Proposition 3

Proof of Proposition 3  Observe that, by the Generalized Schur Complement Lemma (see, e.g., lemma B.1), an optimal choice of $\theta$ in Problem (33) is $\theta = X Y^\top X^\top$. Therefore, we can eliminate $\theta$ from Problem (33), to obtain the equivalent objective:

$$\min_{Y \in \text{Conv}(Y_n)} \min_{X \in \mathbb{R}^{n \times m}} \lambda \cdot \text{tr}(Y) + g(X) + \frac{1}{2\gamma} \langle XX^\top, Y^\top \rangle.$$  

Moreover, by the rank-nullity theorem (see, e.g., Horn and Johnson 1985, Chapter 0.2.3), we can split the columns of $Y$ into columns in the span of the columns of $X$ and columns orthogonal to the columns of $X$. Since the columns orthogonal to the columns of $X$ do not affect the objective value, it follows that we can write $Y^\top = \sum_{i=1}^n \frac{1}{\theta_i} u_i u_i^\top$ without loss of optimality, where $XX^\top = U\Sigma U^\top$ is an SVD of $XX^\top$, and $0 \leq \theta_i \leq 1$ for each $\theta_i$, because $Y \in \text{Conv}(Y_n)$. Problem (33) then becomes:

$$\min_{X \in \mathbb{R}^{n \times m}, \theta \in \mathbb{R}^n: 0 \leq \theta \leq \lambda} \min_{\theta \leq \epsilon} \left( \lambda \theta + \frac{\|X\|_\ast}{2\gamma \theta_i} \right).$$

The result then follows because, for any $\lambda > 0$, (c.f. Pilanci et al. 2015, Equation (30))

$$\min_{\theta \leq \epsilon} \left[ \lambda \theta + \frac{t^2}{\theta} \right] = \begin{cases} 2\sqrt{\lambda}|t|, & \text{if } |t| \leq \sqrt{\lambda}, \\ t^2 + \lambda, & \text{otherwise}. \end{cases}$$  

A.9. Proof of Theorem 6

Proof of Theorem 6  We only detail the proof of $\epsilon$-optimality; the proof of $\epsilon$-feasibility is almost identical (see Mutapcic and Boyd 2009).

Suppose that at some iteration $k > 1$, Algorithm 1 has not converged. Then,

$$\theta_k - f(Y_k) < -\epsilon, \quad \text{and} \quad \theta_k \geq f(Y_i) + \langle H_i, Y_k - Y_i \rangle \quad \forall i < k.$$  

But $\theta_k \leq f(Y_i)$, since $\theta_k$ and $f(Y_i)$ are respectively valid lower and upper bounds on the optimal objective. Therefore, $\langle H_i, Y_k - Y_i \rangle \geq 0$. Putting the two inequalities together then implies that

$$f(Y_k) - f(Y_i) > \epsilon + \langle H_i, Y_k - Y_i \rangle \geq \epsilon, \quad \text{or equivalently} \quad \epsilon < f(Y_k) - f(Y_i) \leq L\|Y_i - Y_k\|_F,$$

where the second inequality holds by Lipschitz continuity. Rearranging this inequality implies that $\|Y_i - Y_k\|_F > \frac{\epsilon}{L}$, i.e., Algorithm 1 never visits any point within a ball of radius $\frac{\epsilon}{L}$ (with respect to the
Frobenius norm) twice. Moreover, by iteration $k$, Algorithm 1 visits $k$ points within non-overlapping balls with combined volume

$$k \frac{\pi n^2}{\Gamma\left(\frac{n^2}{2} + 1\right)} \left(\frac{\epsilon}{L}\right)^n,$$

and these balls are centered at feasible points, i.e., contained within a ball of radius $K + \frac{\epsilon}{L}$ with volume

$$\frac{\pi n^2}{\Gamma\left(\frac{n^2}{2} + 1\right)} \left(K + \frac{\epsilon}{L}\right)^n.$$

That is, if Algorithm 1 has not converged at iteration $k$, we have:

$$k < \left(\frac{LK}{\epsilon} + 1\right) n^2,$$

which implies that we converge to an $\epsilon$-optimal solution within $k \leq \left(\frac{LK}{\epsilon} + 1\right) n^2$ iterations. \hfill $\square$

A.10. Proof of Lemma 6

Proof of Lemma 6 The equality $\theta^* = X_i^\top (Y^*)^\dagger X_i$ is immediate from the Schur complement lemma B.1. Therefore, we focus on deriving an optimal $Y$ for a fixed $X_i$.

From the second equality in the Schur complement lemma B.1, we must have $X = YY^\dagger X$ for feasibility. Therefore, $\text{span}(X) \subseteq \text{span}(Y)$. Moreover, columns of $Y$ which are in null($X$) do not contribute to the optimal objective and can therefore be omitted without loss of optimality. Therefore, $Y = \sum_{i=1}^n \rho_i u_i u_i^\top$ for some $\rho$, where $X_i = U \Sigma V^\top$ is an SVD of $X_i$. The result follows from observing that $0 \leq \rho \leq e$ and $e^\top \rho \leq k$, since $Y \in \text{Conv}(Y_k^n)$. \hfill $\square$

A.11. Proof of Proposition 7

Proof of Proposition 7 As Assumption 2 holds, we can exchange the minimization and maximization operators in Problem (43). Therefore, Problem (43) has the same optimal objective as:

$$\max Y \in \text{Conv}(Y_k^n) \frac{\gamma}{2} \sum_{i=1}^n \sum_{j=1}^n Y_{i,j} \langle \alpha_i, \alpha_j \rangle.$$ (51)

Therefore, to establish the result, it suffices to show that we obtain Problem (44) after taking the dual of Problem (51)'s inner problem. This is indeed the case, because Conv($Y_k^n$) is a convex compact set with non-empty relative interior, and therefore strong duality holds between the following two problems:

$$\max_{Y \succeq 0} \frac{\gamma}{2} \sum_{i=1}^n \sum_{j=1}^n Y_{i,j} \langle \alpha_i, \alpha_j \rangle \quad \text{s.t.} \quad Y \preceq I, \quad \|U\| \leq k, \quad |t|,$$

$$\min_{U \succeq 0, t \geq 0} \text{tr}(U) + kt \quad \text{s.t.} \quad U + t \succeq \frac{\gamma}{2} \alpha \alpha^\top. \quad \square$$

A.12. Proof of Theorem 7

Proof of Theorem 7 It suffices to establish that

$$f(Y) - f(Y^*) \leq \frac{\gamma}{2} L^2 |R| \max_{\beta \geq 0, \|\beta\|_1 \leq 1} \sum_{i \in R} (\Lambda^*_i - \Lambda_{i,i}) \beta_i,$$

under the Frobenius penalty and

$$f(Y) - f(Y^*) \leq ML |R| \max_{\beta \geq 0, \|\beta\|_1 \leq 1} \sum_{i \in R} (\Lambda^*_i - \Lambda_{i,i}) \beta_i,$$
for the spectral penalty, as the result then follows by minimizing the Lipschitz bound over $\lambda \in \{0, 1\}^n$:  
\[ e^\top \lambda \leq k, \lambda_i = \Lambda_i, \text{ i.e., rounding greedily.} \]

Under a Frobenius penalty, by Lipschitz continuity, we have
\[
f(Y) - f(Y^*) \leq \frac{\gamma}{2} \langle \alpha^*(Y)\alpha^*(Y)^\top, U(\Lambda^* - \Lambda)U^\top \rangle = \frac{\gamma}{2} \langle U^\top \alpha^*(Y)\alpha^*(Y)^\top U, \Lambda^* - \Lambda \rangle.
\]
Moreover, since $\Lambda^* - \Lambda$ is a diagonal matrix we need only include the diagonal terms in the inner product. Therefore, since
\[
(U^\top \alpha^*(Y)\alpha^*(Y)^\top U)_{i,i} = \langle \alpha^*(Y)\alpha^*(Y)^\top U_iU_i^\top \rangle \leq \lambda_{\max}(\alpha^*(Y)\alpha^*(Y)^\top) \leq L^2,
\]
where the second-to-last inequality holds because $\|U_i\|_2 = 1$, the bound on $f(Y) - f(Y^*)$ holds.

Alternatively, under spectral norm regularization, by Lipschitz continuity we have
\[
f(Y) - f(Y^*) \leq M \langle V_{11}^*(Y) + V_{22}^*(Y), U(\Lambda^* - \Lambda)U^\top \rangle = M \langle U^\top (V_{11}^*(Y) + V_{22}^*(Y))U, \Lambda^* - \Lambda \rangle.
\]
Moreover, $\Lambda - \Lambda^*$ is a diagonal matrix and therefore
\[
(U^\top (V_{11}^*(Y) + V_{22}^*(Y))U)_{i,i} = \langle U_iU_i^\top, V_{11}^*(Y) + V_{22}^*(Y) \rangle \leq \lambda_{\max}(\alpha^*(Y)) \leq L,
\]
where the last inequality follows since $V_{11}, V_{22}$ are orthogonal at optimality, meaning $V_{11} + V_{22}$’s leading eigenvalue equals $\alpha^*$’s leading singular value. Therefore, the bound on $f(Y) - f(Y^*)$ holds.  

\[ \square \]

**B. A Collection of Useful Matrix Identities**

In this work, we have repeatedly used some technical matrix identities. In order to keep this paper self contained, we now state these identities formally.

The following result generalizes the well-known Schur complement lemma to the case where neither on-diagonal block matrix is positive definite (see Boyd et al. 1994, Equation 2.41)

**Lemma B.1 (Generalized Schur Complement Lemma)** Let $A, B, C$ be components of  
\[ X := \begin{pmatrix} A & B \\ B^\top & C \end{pmatrix} \]

of appropriate dimension. Then, $X$ is positive semidefinite if and only if the following conditions hold:

(i) $A \succeq 0$.

(ii) $(I - AA^\dagger)B = 0$.

(iii) $C \succeq B^\top A^\dagger B$.

The following result characterizes the rank of a block submatrix in terms of the rank of the entire matrix (see, e.g., Puntanen and Styan 2005, Section 0.9):

**Lemma B.2 (Guttman Rank Identity)** Let $A, B, C$ be components of the matrix  
\[ X := \begin{pmatrix} A & B \\ B^\top & C \end{pmatrix} \]

of appropriate dimension. Then, we have the identity:  
\[ \text{Rank}(X) = \text{Rank}(A) + \text{Rank}(C - BC^\dagger B^\top). \]
In general, a product of positive semidefinite matrices may not be positive semidefinite, indeed, it may not even be symmetric. However, the following result demonstrates that a symmetric product of PSD matrices is indeed PSD (see Meenakshi and Rajian 1999, for a proof):

**Lemma B.3 (A Symmetric Product of PSD Matrices is PSD)** Let \( X, Y \in S^n_+ \) be matrices of appropriate size, and let \( Z := XY \). Suppose that \( Z = Z^T \) is a symmetric matrix. Then, \( Z \succeq 0 \).

The following result demonstrates that evaluating the nuclear norm of a matrix via semidefinite optimization yields a singular value decomposition (see Recht et al. 2010, Proposition 2.1, for a proof):

**Lemma B.4 (Nuclear norm minimization and the SVD of a matrix)** An optimal solution to 
\[
\min_{W_1 \in \mathbb{R}^{n \times n}, W_2 \in \mathbb{R}^{m \times m}} \mathrm{tr}(W_1) + \mathrm{tr}(W_2) \quad \text{s.t.} \quad \begin{pmatrix} W_1 & X \\ X^T & W_2 \end{pmatrix} \succeq 0,
\]

is attained by \( W_1 = U \Sigma U^T \) and \( W_2 = V \Sigma V^T \), where \( X = U \Sigma V^T \) is a singular value decomposition.

### C. Properties of Projection Matrices

In this section, we collect useful properties of orthogonal projection matrices used repeatedly here. We refer the reader to Horn and Johnson (1985) (see also Dattorro 2010, Appendix E) for more properties of projection matrices.

**Lemma C.1** Let \( Y \in \mathcal{Y}_n \) be an orthogonal projection matrix. Then, \( Y = Y^2 = Y^\dagger \), \( \text{Rank}(Y) = \text{tr}(Y) \).

**Proof of Lemma C.1** Suppose that \( Y \) is indeed an orthogonal projection matrix. Then, it admits the singular value decomposition \( Y = U \Sigma U^T \), where \( \Sigma \) is a diagonal matrix containing binary entries. Moreover, \( Y^2 = U \Sigma^2 U^T = U \Sigma U^T = Y \), and \( Y^\dagger = U \Sigma^\dagger U^T = U \Sigma U^T = Y \), as required.

Finally, the equality \( \text{Rank}(Y) = \text{tr}(Y) \) holds because \( Y \)'s singular values are binary. \( \square \)

The next lemma is essentially a restatement of (Overton and Womersley 1992, Theorem 3):

**Lemma C.2** Let \( \mathcal{Y}_n \) denote the set of \( n \times n \) orthogonal projection matrices and \( \mathcal{Y}_k^n \) denote the set of low-rank orthogonal projection matrices. Then, \( \text{Conv}(\mathcal{Y}_n) = \{ P : 0 \preceq P \preceq \mathbb{I} \} \) and \( \text{Conv}(\mathcal{Y}_k^n) = \{ P : 0 \preceq P \preceq \mathbb{I}, \text{tr}(Y) \leq k \} \). Moreover, the set of extreme points of \( \text{Conv}(\mathcal{Y}_n) \) are precisely \( \mathcal{Y}_n \), and the set of extreme points of \( \text{Conv}(\mathcal{Y}_k^n) \) are precisely \( \mathcal{Y}_k^n \).

The following lemma allows us to recover an optimal projection matrix given an optimal low-rank matrix \( X^\star \) via a singular value decomposition:

**Lemma C.3** Let \( X^\star \) be an optimal choice of \( X \). Then, an optimal choice of \( Y \), \( Y^\star \), is given by \( Y^\star = U U^T \), where \( X = U \Sigma V^T \) is a singular value decomposition of \( X \) and we only include the first \( k \) columns in \( U \) where \( \text{Rank}(X) = k \).

**Proof of Lemma C.3** Recall that for any \( X \), there must be some \( Y \in \mathcal{Y}_n \) such that \( X = Y X \). Under the proposed construction, we have \( Y X = U U^T U \Sigma V^T = U \Sigma V^T = X \), where the middle equality holds since \( U^T U = \mathbb{I} \). Therefore, since \( U U^T \) is an orthogonal projection matrix, the result holds. \( \square \)
D. Additional Results

D.1. Pseudocode for the In-out Method

Our main points of difference from Fischetti et al. (2016)’s implementation are twofold. First, we optimize the outer problem over \( \text{Conv}(\mathbf{y}_k^n) \), rather than the Boolean polytope. Second, we recognize that the purpose of the method is to warm-start Algorithm 1’s lower bound, rather than to solve Problem (28). In this spirit, we accelerate the in-out method by first solving Problem (28) in one shot using an interior point method and second using the optimal solution \( \mathbf{Y}^* \) as a stabilization point. Note that a similar method was proposed for sparse portfolio selection problems and MICO by Bertsimas and Cory-Wright (2018).

\begin{algorithm}
\begin{algorithmic}
\Require Stabilization point \( \mathbf{Y}^* \), \( \epsilon \leftarrow 10^{-10}, \lambda \leftarrow 0.1, \delta \leftarrow 2\epsilon, t \leftarrow 1 \\
\Repeat
\State Compute \( \mathbf{Y}_0, \theta_0 \) solution of
\[ \min_{\mathbf{Y} \in \text{Conv}(\mathbf{y}_k^n)} \theta + \lambda \cdot \text{tr}(\mathbf{Y}) \quad \text{s.t.} \quad z_i \theta \geq h_i + \langle \mathbf{H}_i, \mathbf{Y} - \mathbf{Y}_i \rangle \forall i \in [t]. \]
\If \( \mathbf{Y}_0 \) has not improved for 5 consecutive iterations
\State Set \( \lambda = 1 \)
\If \( \mathbf{Y}_0 \) has not improved for 10 consecutive iterations
\State Set \( \delta = 0 \)
\EndIf
\EndIf
\State Set \( \mathbf{Y}_{t+1} \leftarrow \lambda \mathbf{Y}_0 + (1 - \lambda) \mathbf{Y}^* + \delta \mathbf{I} \); project \( \mathbf{Y}_{t+1} \) onto \( \text{Conv}(\mathbf{y}_k^n) \).
\State Compute \( f(\mathbf{Y}_{t+1}, H_{t+1}, z_{t+1}, d_{t+1}) \).
\Until \( f(\mathbf{Y}_0) - \theta_0 \leq \epsilon \)
\State \text{return} \( \mathbf{Y}_t \)
\end{algorithmic}
\end{algorithm}

D.2. On Implementing Algorithm 2

We now discuss some modifications to Algorithm 2 which improve its rate of convergence in practice.

- Imposing a proximal regularization term in the objective, namely \( + \frac{\tau}{2} \| \mathbf{X} - \mathbf{X}_t \|_F^2 \), improves the rate of convergence of the method by stabilizing the iterates; we make use of this in our experiments.

- The method stalls when the eigenvalues of \( \mathbf{Y}_t \) are near zero (a) due to numerical instability and (b) because \( \mathbf{Y}_t \) is near the boundary of \( \text{Conv}(\mathbf{y}_k^n) \). Therefore, to accelerate convergence, we require that \( \lambda_{\min}(\mathbf{Y}) \geq \frac{K}{t} \) at the \( t \)-th iterate, where \( K \approx 10^{-2} \). In practice, this introduces very little error.
• Selecting an optimal $W^{t+1}$ is generally much cheaper than selecting an optimal $V^{t+1}$, since the former problem involves optimizing over $n$ eigenvalues, rather than $n^2$ variables. Therefore, efficient implementations of Algorithm 2 necessarily require efficient methods for obtaining $V^{t+1}$. In the case of matrix completion, $g(X)$ is a quadratic form, which implies that obtaining $V^{t+1}$ is equivalent to solving a linear system, which we do iteratively in our numerical experiments.

• We solve for $V^{t+1}$ by solving the first-order optimality condition using a successive over-relaxation linear technique, or in rare instances where the linear system solver fails to converge we use Ipopt to solve the QP’s first-order optimality condition.