Abstract

This paper investigates an iterative approach to solve the general rank-constrained optimization problems (RCOPs) defined to optimize a convex objective function subject to a set of convex constraints and rank constraints on unknown rectangular matrices. In addition, rank minimization problems (RMPs) are introduced and equivalently transformed into RCOPs by introducing a quadratic matrix equality constraint. The rank function is discontinuous and nonconvex, thus the general RCOPs are classified as NP-hard in most of the cases. An iterative rank minimization (IRM) method, with convex formulation at each iteration, is proposed to gradually approach the constrained rank. The proposed IRM method aims at solving RCOPs with rank inequalities constrained by upper or lower bounds, as well as rank equality constraints. Proof of the convergence to a local minimizer with at least a sublinear convergence rate is provided. Four representative applications of RCOPs and RMPs, including system identification, output feedback stabilization, and structured \( H_2 \) controller design problems, are presented with comparative simulation results to verify the feasibility and improved performance of the proposed IRM method.

Key words: Rank-Constrained Optimization; Matrix Rank Minimization; Convex Relaxation; Semidefinite Programming; Feedback Control.

1 INTRODUCTION

Rank-constrained optimization problems (RCOPs) are to minimize a convex function subject to a convex set of constraints and rank constraints on unknown matrices. They have received extensive attention due to their wide applications in signal processing, model reduction, and system identification, just to name a few [18, 21, 22, 41]. Although some special RCOPs can be solved analytically [14, 25], they are NP-hard in most of the cases. Existing methods for RCOPs mainly focus on alternating projection based methods [4, 5, 15] and combined linearization and factorization algorithms [16, 30] with application to factor analysis, etc. However, these iterative approaches depend on the initial guess and fast convergence cannot be guaranteed. In addition, a Newton-like method [33] has been proposed to search for a feasible solution of RCOPs with application to a feedback stabilization problem. A Riemannian manifold optimization method [43] has been applied to solve large-scale Lyapunov matrix equations by finding a low-rank approximation. Also, [42] proposes to use the toolbox BARON to solve a RCOP. There are alternative approaches for solving specially formulated RCOPs.

For example, a greedy convex smoothing algorithm has been designed to optimize a convex objective function subject to only one rank constraint [37]. When the rank function in constraints of RCOPs appears as the objective of an optimization problem, it turns to be a rank minimization problem (RMP), classified as a category of nonconvex optimization. Applications of RMPs have been found in a variety of areas, such as matrix completion [2, 31, 35], control system analysis and design [10, 11, 13, 28, 29], and machine learning [27, 34]. The wide application of RMPs attracts extensive studies aiming at developing efficient optimization algorithms.

Due to the discontinuous and nonconvex nature of the rank function, most of the existing methods solve relaxed or simplified RMPs by introducing an approximate function, such as log-det or nuclear norm heuristic methods [9, 12]. The heuristic methods minimize a relaxed convex function instead of the exact rank function over a convex set, which is computationally favorable. They generally generate a solution with lower rank, even a minimum rank solution in special cases [35]. The relaxed formulation with convex objective and constraints does not require the initial guess and global optimality is guaranteed for the relaxed formulation. When the unknown matrix is constrained to be positive semidefinite, relaxation of RMPs using a trace function is equivalent to the relaxed formulation using a nuclear norm function.
based on the fact that the trace of a positive semidefinite matrix equals to its nuclear norm [29]. For cases when the unknown matrix is not positive semidefinite, work in [9] introduces a semidefinite embedding lemma to extend the trace heuristic method to general cases.

However, a relaxed function cannot represent the exact rank function and performance of the heuristic method is not guaranteed. Other heuristic methods, e.g., the iterative reweighted least square algorithm [31] which iteratively minimizes the reweighted Frobenius norm of the matrix, cannot guarantee the minimum rank solution either. The uncertainty of the performances in heuristic methods stems from the fact that these methods are minimizing a relaxed function and generally there is a gap between the relaxed objective and the exact one. Other methods for RMPs include the alternating projections and its variations [15,19,33], linearization [7,16], and augmented Lagrangian method [8]. These methods, similar to existing iterative methods for RCOPs, depend on initial guess, which generally leads to slow convergence to just a feasible solution.

After reviewing the literature, we come to a conclusion that more efficient approaches that are applicable for general RCOPs/RMPs with advantages in terms of convergence rate, robustness to initial guess, and performance of cost function, are required to solve RCOPs and RMPs. To our knowledge, there is few literature addresses equivalent conversion from RMPs into RCOPs [6,39]. This paper describes a novel representation of RMPs in the form of RCOPs and proposes a uniform approach to both RCOPs and reformulated RMPs. Therefore, instead of solving two classes of nonconvex optimization problems separately, the uniform formulation and approach significantly reduces the required efforts for solving two types of challenging problems.

An iterative rank minimization (IRM) method, with each sequential problem formulated as a convex optimization problem, is proposed to solve RCOPs. The IRM method was introduced in our previous work to solve quadratically constrained quadratic programming problems which are equivalent to rank-one constrained optimization problems [3,40]. The IRM method proposed in this paper aims to solve general RCOPs, where the constrained rank could be any assigned integer number. Although IRM is primarily designed for RMPs with rank constraints on positive semidefinite matrices, a semidefinite embedding lemma [9] is introduced to extend IRM to RCOPs with rank constraints on general rectangular matrices. Moreover, The proposed IRM method is applicable to RCOPs with rank inequalities constrained by upper or lower bounds, as well as rank equality constraints. Sublinear convergence of IRM is proved via the duality theory and the Karush-Kuhn-Tucker conditions. To verify the effectiveness and improved performance of proposed IRM method, four representative applications, including system identification, output feedback stabilization, and structured $H_2$ controller design problems, are presented with comparative results.

The rest of the paper is organized as follows. In Section II, the problem formulation of RCOP and the conversion of RMP to RCOP are described, including extension to rank constraints on general rectangular matrices. The IRM approach and its local convergence proof are addressed in Section III. Four application examples and their comparative results are presented in Section IV. We conclude the paper with a few remarks in Section V.

1.1 Preliminaries

Some notations used throughout this paper are introduced in this section. The $n$-dimensional Euclidean space is denoted by $\mathbb{R}^n$. The set of $n \times n$ symmetric matrices is denoted by $\mathbb{S}^n$ and the set of $n \times n$ positive semidefinite (definite) matrices is denoted by $\mathbb{S}_+^n (\mathbb{S}_{++}^n)$. The notation $X \succeq 0 (X > 0)$ means that the matrix $X \in \mathbb{S}^n$ is positive semidefinite (definite). The symbol ‘$\leftarrow$’ means if and only if logical connective between statements. The trace of $X$ is denoted by $\text{trace}(X)$ and the rank of $X$ is denoted by $\text{rank}(X)$. The linear span of vectors in $X$ is denoted by $\text{span}(X)$.

2 PROBLEM FORMULATION

2.1 Rank-Constrained Optimization Problems

A general RCOP to optimize a convex objective subject to a set of convex constraints and rank constraints can be formulated as follows

$$\min_X f(X) \quad s.t. \ X \in C, \ \text{rank}(X) \leq r, \quad (2.1)$$

where $f(X)$ is a convex function, $C$ is a convex set, and $X \in \mathbb{R}^{m \times n}$ is a general rectangular matrices set. Without loss of generality, it is assumed that $m \leq n$. The sign $\leq$ include all types of rank constraints, including upper and lower bounded rank equality constraints and rank equality constraints. Although lower bounded rank inequality constraints and rank equality constraints do not have as many practical applications compared to the upper bounded rank inequality constraints, they are included here for completeness. Because the existing and proposed approaches for RCOPs require the to-be-determined matrix to be a positive semidefinite matrix, it is then necessary to convert the rank constraints on rectangular matrices into corresponding ones on positive semidefinite matrices.

Lemma 1 (Lemma 1. in [10]) Let $X \in \mathbb{R}^{m \times n}$ be a given matrix. Then $\text{rank}(X) \leq r$ if and only if there exists matrices $Y = Y^T \in \mathbb{R}^{m \times m}$ and $Z = Z^T \in \mathbb{R}^{n \times n}$ such that

$$\text{rank}(Y) + \text{rank}(Z) \leq 2r, \quad \begin{bmatrix} Y & X \\ X^T & Z \end{bmatrix} \succeq 0.$$

However, Lemma (1) is not applicable to lower bounded rank inequality constraints. As a result, a new lemma is introduced to extend the above semidefinite embedding lemma to all types of rank constraints. Before that, we first describe a proposition which is involved in proof of the new lemma.
Proposition 2: \( Z = X^T X \) is equivalent to
\[
\text{rank} \left( \begin{bmatrix} I_m & X \\ X^T & Z \end{bmatrix} \right) \leq m, \text{ where } Z \in S^n, \ X \in \mathbb{R}^{m \times n}, \text{ and } I_m \in \mathbb{R}^{m \times m} \text{ is an identify matrix.}
\]

Proof: Given that the rank of a symmetric block matrix is equal to the rank of a diagonal block plus the rank of its Schur complement, we have the following relationship, \( \text{rank} \left( \begin{bmatrix} I_m & X \\ X^T & Z \end{bmatrix} \right) \leq m \Leftrightarrow \text{rank}(I_m) + \text{rank}(Z) \leq m \).

Remark: When \( Z = X^T X \), it indicates that \( \begin{bmatrix} I_m & X \\ X^T & Z \end{bmatrix} \succeq 0 \) holds for \( I_m \succ 0 \) and its Schur complement is a zero matrix.

Lemma 3: Let \( X \in \mathbb{R}^{m \times n} \) be a given matrix. Then \( \text{rank}(X) \leq r (= r, \geq r, \text{resp.}) \) if and only if there exists a matrix \( Z \in S^n \) such that
\[
\text{rank}(Z) \leq r \Leftrightarrow \text{rank} \left( \begin{bmatrix} I_m & X \\ X^T & Z \end{bmatrix} \right) \leq m.
\]

Proof: From Proposition (2), \( \text{rank} \left( \begin{bmatrix} I_m & X \\ X^T & Z \end{bmatrix} \right) \leq m \Leftrightarrow Z = X^T X \). It is known that \( \text{rank}(Z) = \text{rank}(X) \) when \( Z = X^T X \). Hence the proof is complete. \( \square \)

Remark: For an upper bounded rank inequality constraint, \( \text{rank}(X) \leq r \), Lemma 3 can be interpreted as a special case of Lemma 1 by setting \( Y \) in Lemma 1 be \( I_m \) and \( \text{rank}(Z) = \text{rank}(X) \) and replacing \( 2r \) by \( m + r \). Lemma 3 for this case transforms the rank inequality on rectangular matrix into two rank inequalities on semidefinite matrices while Lemma 1 equivalently transforms it into one rank inequality and one semidefinite constraint.

The case of the rank equality constraint, \( \text{rank}(X) = r \), since the semidefinite constraint in Lemma 1 indicates that \( \text{rank}(Y) \geq \text{rank}(X) \) and \( \text{rank}(Z) \geq \text{rank}(X) \). Lemma 1 for this case leads to two rank constraints and one semidefinite constraint, expressed as
\[
\text{rank}(Y) = r, \ \text{rank}(Z) = r, \ \text{and} \ \begin{bmatrix} Y & X \\ X^T & Z \end{bmatrix} \succeq 0. \ (2.2)
\]

The general semidefinite Lemma in 3 for this case has the same form of expressions compared to the case with upper bounded rank inequality. Additionally, for a lower bounded rank inequality constraint, \( \text{rank}(X) \geq r \), Lemma 1 is not applicable based on the fact that \( \text{rank}(Y) \geq \text{rank}(X) \) and \( \text{rank}(Z) \geq \text{rank}(X) \) always leads to \( \text{rank}(X) \leq r \). However, the uniform formulation stated in Lemma 3 is applicable to all cases.

Consequently, based on Lemma 3, problem in (2.1) with rank constraints on rectangular matrices can be equivalently transformed into the following formulation with rank constraints on positive semidefinite matrices,
\[
\min_{X,Z} \quad f(X) \quad \text{s.t.} \quad X \in \mathcal{C}, \ \text{rank}(Z) \leq r (= r, \geq r, \text{resp.}) \quad \text{rank} \left( \begin{bmatrix} I_m & X \\ X^T & Z \end{bmatrix} \right) \leq m,
\]

where \( Z \in S^n \) is a newly introduced auxiliary matrix. The above formulation can be summarized as a RCOP with convex objective, a set of convex constraints, and rank constraints on semidefinite matrices. For simplicity, the following discussion of RCOP considers rank constraints on semidefinite matrices only.

2.2 Rank Minimization Problems and Reformulation

A RMP to minimize a rank function within a convex set is formulated as
\[
\min_X \text{rank}(X) \quad \text{s.t.} \quad X \in \mathcal{C}, \quad (2.4)
\]

where \( X \in \mathbb{R}^{m \times n} \) is an unknown matrix and \( \mathcal{C} \) is a convex set. Without loss of generality, we assume \( m \leq n \) in the above formulation provided that \( \text{rank}(X) = \text{rank}(X^T) \). The matrix rank function is discontinuous and highly nonlinear. In the following, we introduce an equivalent conversion to reformulate RMPs as RCOPs.

Based on the fact that the trace of a projection matrix is the dimension of the target space, the mathematical expression of this statement is in the form of
\[
P(X) = X(X^T X)^{-1}X^T, \ \text{trace}(P(X)) = \text{rank}(X),
\]

where \( P(X) \in \mathbb{R}^{m \times m} \) is the projection matrix and its trace is equivalent to the rank of \( X \) if \( X^T X \) is nonsingular. For singular cases, a small regularization parameter, \( \epsilon \), is introduced to reformulate \( P(X) \) as \( P_r(X) = X(X^T X + \epsilon I_n)^{-1}X^T \). It has been verified that \( \text{trace}(P_r(X)) \) can approximately represent \( \text{rank}(X) \) at any prescribed accuracy as long as \( \epsilon \) is properly given [45]. Since \( \text{trace}(P_r(X)) \) is continuous and differentiable with respect to \( X \), the rank function can be replaced by \( \text{trace}(P_r(X)) \) and RMP in (2.4) is rewritten as
\[
\min_{X,Y} \quad \text{trace}(Y) \quad \text{s.t.} \quad X \in \mathcal{C}, \quad Y \succeq X(X^T X + \epsilon I_n)^{-1}X^T, \quad (2.5)
\]

where \( Y \in S^m \) is a slack symmetric matrix. The new formulation in (2.5) is equivalent to (2.4) based on the
fact that if \((Y^*, X^*)\) is an optimal solution pair to (2.5), its matrix inequality constraint will be active such that 
\(Y^* = X^* (X^*)^T + \lambda_{n-r} I\) since we want to minimize \(\text{trace}(Y)\) where \(Y\) is the upper bound of 
\(X (X^T X + \lambda I)^{-1} X^T\). Hence, \(X^*\) is an optimum to (2.4).

In addition, by using Schur complement to convert the nonlinear matrix inequality in (2.5) into a linear matrix equality by introducing a new matrix \(Z\in\mathbb{S}^n\), problem in (2.5) can be equivalently transformed into the following representation,

\[
\min_{X,Y,Z} \quad \text{trace}(Y) \\
\text{s.t.} \quad X \in \mathcal{C}, \begin{bmatrix} Y & X \\ X^T & Z \end{bmatrix} \succeq 0, \quad Z = X^T X. \tag{2.6}
\]

The above problem, with a convex objective function, and dependent on the existence of initial guess.

In fact, the number of nonzero eigenvalues of a matrix is identical to its rank. For an unknown square matrix \(U \in \mathbb{S}^n\), it is not feasible to examine its eigenvalues before it is determined. Therefore, instead of making constraint on the rank, we focus on constraining the eigenvalues of \(U\) such that the \(n - r\) eigenvalues of \(U\) are all zeros. The eigenvalue constraints on matrices have been used for graph design [36] and are applied here for rank-constrained problems. Before addressing the detailed approach for rank-constrained problems, we first provide necessary observations that will be used subsequently in the approach.

**Proposition 4** The \((r+1)\)th largest eigenvalue \(\lambda_{n-r}\) of matrix \(U \in \mathbb{S}^n\) is no greater than \(e\) if and only if \(e I_{n-r} - V^T U V \succeq 0\), where \(I_{n-r}\) is the identity matrix with a dimension of \(n - r\), \(V \in \mathbb{R}^{n \times (n-r)}\) are the eigenvectors corresponding to the \(n-r\) smallest eigenvalues of \(U\).

**Proof:** Assume the eigenvalues of \(U\) is sorted in descending orders in the form of \([\lambda_n, \lambda_{n-1}, \ldots, \lambda_1]\). Since the Rayleigh quotient of an eigenvector is its associated eigenvalue, then \(e I_{n-r} - V^T U V\) is a diagonal matrix with diagonal elements set as \([e\lambda_n, e\lambda_{n-1}, \ldots, e\lambda_1]\). Therefore \(e \geq \lambda_{n-r}\) if and only if \(e I_{n-r} - V^T U V \succeq 0\). □

**Corollary 5** When \(e = 0\) and \(U\) is a positive semidefinite matrix, \(\text{rank}(U) \leq r\) holds if and only if \(e I_{n-r} - V^T U V \succeq 0\). \(\text{rank}(U) = r\) holds if and only if \(e I_{n-r} - V^T U V \succeq 0\) and \(V_p^T U V_p \succ 0\). \(\text{rank}(U) \geq r\) holds if and only if \(V_p^T U V_p \succeq 0\). The matrices \(V \in \mathbb{R}^{n \times (n-r)}\) and \(V_p \in \mathbb{R}^{n \times 1}\) are the eigenvectors corresponding to the \(n-r\) smallest eigenvalues and \(n-r+1\) smallest eigenvalue of \(U\), respectively.

However, for problem (3.8), before we solve \(X\), we cannot obtain the exact \(V\) matrix, i.e. the eigenvectors of \(X\). Therefore an iterative method is proposed to solve (3.8) by gradually approaching the constrained rank. At each step \(k\), we will solve the following semidefinite programming problem formulated as

\[
\min_{X,k,e_k} \quad f(X_k) + w_k e_k \\
\text{s.t.} \quad X_k \in \mathcal{C}, X_k \succeq 0, e_k I_{n-r} - V_{k-1}^T X_k V_{k-1} \succeq 0, e_k \leq e_{k-1}. \tag{3.9}
\]

where \(w_k = w_0 t^k\) is the weighting factor at iteration \(k\). \(w_0\) is increasing with the increment of \(k\) when \(t > 1\) and \(w_0 > 0\) are given parameters. \(V_{k-1} \in \mathbb{R}^{n \times (n-r)}\) are the orthonormal eigenvectors corresponding to the \(n-r\) smallest eigenvalues of \(X_{k-1}\) solved at previous iteration \(k-1\). At the first iteration where \(k = 1\), \(e_0\) is the \((n-r)\)th smallest eigenvalue of \(X_0\) described below. Furthermore, we will prove that each sequential problem is feasible with the additional constraint, \(e_k \leq e_{k-1}\), in the convergence analysis section. It is known that the existing SDP solver based on interior point method has
computational time complexity in order of $O(m(n^2m + n^3))$ for a SDP problem with $m$ linear constraints and a linear matrix inequality of dimension $n$. As a result, the extra linear matrix inequality $e_kI - V_k^TX_kV_k \succeq 0$ in the subproblem leads to additional linear constraints at the order above $O(n^6)$. Therefore, the time complexity of each iteration is lower bounded by $O(n^6)$.

At each step, we are trying to optimize the original objective function and at the same time minimize parameter $e$ such that when $e = 0$, the rank constraint on $X$ is satisfied. The weighting factor, $w_k$, acts as a regularization factor and increasing its values at each step will enforce the $r + 1$th largest eigenvalue to gradually reduce to zero. With this regularization factor, our algorithm not simply switches between searching for the unknown matrix and then its eigenvectors, it also drives the algorithm not simply switches between searching for the unknown matrix and then its eigenvectors, it also drives the slack variable $e_k$ to quickly reduce to zero. Furthermore, the additional constraint, $e_k \leq e_{k-1}$, guarantees that $e_k$ is monotonically decreasing in the iterative algorithm. The above approach is repeated until $e \leq \epsilon$, where $\epsilon$ is a small threshold for the stopping criteria. Each iteration is solved via an existing semidefinite programming (SDP) solver based on an interior point method, which is applicable to small and medium size SDP problems [38]. It is straightforward to extend (3.9) to problems with multiple rank constraints. For brevity, the simplest version with one rank constraint is described here.

In addition, an initial starting point, $V_0$, is required at the first iteration $k = 1$. It is intuitive to use the relaxed solution by dropping the last constraint and the penalty term in the objective function in (3.9) for starting point. Under this assumption, $X_0$ is obtained via

$$\min_{X_0} \quad f(X_0)$$

$$s.t. \quad X_0 \in \mathcal{C}, X_0 \succeq 0,$$  \hspace{1cm} (3.10)

and $V_0$ are the eigenvectors corresponding to $n - r$ smallest eigenvalues of $X_0$. However, under special cases, adding the penalty term in the objective function will not change the solution found from the semidefinite relaxation formulation, e.g., the penalty term is identical to the objective function, which makes the penalty term and the semidefinite constraint become inactive. A new initial value will be used in the IRM algorithm if the one from the semidefinite relaxation problem does not converge to the constrained-rank. The IRM approach is summarized in Algorithm 1 for general RCOPs.

3.2 IRM Approach to RMPs

Combining the algorithm for RCOPs in the above subsection, the RMPs reformulated as RCOPs in (2.7) can be solved iteratively and the sequential problem at iteration $k$ is formulated as

\begin{algorithm}
\caption{Iterative Rank Minimization for Solving (3.8)}
\begin{algorithmic}
\State \textbf{Input:} Problem information $C, w_0, t$, $\epsilon_1$, $\epsilon_2$, $k_{\text{max}}$
\State \textbf{Output:} with local minimum $X^*$
\begin{algorithmic}
\State \textbf{begin}
\State (1) Initialize Set $k = 0$, solve the relaxed problem in (3.10) to obtain $V_0$ from $X_0$ via eigenvalue decomposition. set $k = k + 1$
\State (2) while $k \leq k_{\text{max}}$ & $e_k \geq \epsilon_1$ & $|f(X_k) - f(X_{k-1})|/f(X_{k-1})| \geq \epsilon_2$
\State (3) Solve sequential problem (3.9) and obtain $X_k, e_k$
\State (4) Update $V_k$ from $X_k$ via eigenvalue decomposition and set $k = k + 1$
\State (5) Update $w_k$ via $w_k = w_{k-1} + t$
\State (6) end while
\State end
\end{algorithmic}
\end{algorithm}

\end{algorithm}

$$\min_{X_k, Y_k, Z_k, e_k} \quad \text{trace}(Y_k) + w_k e_k$$

$$s.t. \quad X_k \in \mathcal{C},$$

$$\begin{bmatrix} Y_k & X_k \\ X_k^T & Z_k \end{bmatrix} \succeq 0, \quad \begin{bmatrix} I_m & X_k \\ X_k^T & Z_k \end{bmatrix} \succeq 0, \quad (3.11)$$

$$e_k I_n - V_{k-1}^T \begin{bmatrix} I_m & X_{k-1} \\ X_{k-1}^T & Z_{k-1} \end{bmatrix} V_{k-1} \succeq 0,$$

$$e_k \leq e_{k-1},$$

where $V_{k-1} \in \mathbb{R}^{(m+n) \times n}$ are the eigenvectors corresponding to the $n$ smallest eigenvalues of $\begin{bmatrix} I_m & X_{k-1} \\ X_{k-1}^T & Z_{k-1} \end{bmatrix}$ solved at previous iteration $k - 1$.

Since the heuristic methods, e.g., trace heuristic method, are often considered as a good candidate for approximate solutions of RMPs. They are adopted here to obtain the starting point, $V_0$, by solving the relaxed $X_0$. The trace heuristic method for RMPs formulated in (2.4) is listed below by applying the aforementioned semidefinite embedding lemma for general unknown matrix $X \in \mathbb{R}^{m \times n}$ [11],

$$\min_{X_0, Y_0, Z_0} \quad \text{trace}(Y_0)$$

$$s.t. \quad X_0 \in \mathcal{C}, \begin{bmatrix} Y_0 & X_0 \\ X_0^T & Z_0 \end{bmatrix} \succeq 0, \quad (3.12)$$

Providing this starting point, problem (2.7) can be solved by the proposed IRM algorithm where the sequential problem at each iteration is formulated in (3.11).

3.3 Convergence Analysis of IRM

In the following, we provide the convergence analysis of the proposed IRM method for solving general RCOPs. We first introduce one preparatory lemma.

Lemma 6 (Corollary 4, 3.37 in [17]) For a given matrix $X \in \mathbb{S}^n$ with eigenvalues in increasing order, denoted as $[\lambda_1^X, \lambda_2^X, \ldots, \lambda_n^X]$, when $m \leq n$, one has $\lambda_m^X \leq \lambda_1^X$.
\[ \lambda_{\text{max}}(V^TVV) \quad \text{for any} \quad V \in \{V \mid V \in \mathbb{R}^{n \times m}, V^TV = I_m \}. \] Moreover, equality holds when columns of \( V \) are the eigenvectors of \( X \) associated with the \( m \) smallest eigenvalues.

**Assumption 7** Problem in (3.8) is feasible with interior points \([1]\) and the relaxed problem (3.10) is bounded.

**Proposition 8** (Local Convergence) When Assumption 7 holds, the sequential problem of (3.9) is feasible. Moreover, when \( \|e_k - e^*\| \to 0, \) i.e., \( \lambda_{\text{max}}(V_k^TV_k) \leq \lambda_{\text{max}}(V_k^TV_k), \) where the inequality follows Lemma 6. According to Assumption 7, the relaxed formulation in (3.10) has an optimal solution, denoted by \( X_0. \) For the first iteration \( k = 1, \) the pair \( (X_0, e_0) \) is a feasible solution for the sequential problem at \( k = 1, \) where \( e_0 \) is the \( r \) + 1th largest eigenvalue of \( V_k. \) For any \( k \geq 1, \) the solution pair \( (X_k, e_k) \) obtained at iteration \( k \) is a feasible solution pair for the sequential problem at iteration \( k + 1. \) It is because \( X_k \) satisfies the original constraints \( X_k \in C \) and \( X_k \geq 0 \) at iteration \( k, \) then it will satisfy the same set of constraints at iteration \( k + 1. \) Furthermore, since \( e_k \) satisfies \( e_kI_{n-r} - V_k^TV_k \geq 0 \) at iteration \( k \) and \( e_k \geq \lambda_{\text{max}}(V_k^TV_k), \) the inequality \( e_kI_{n-r} - V_k^TV_k \geq 0 \) at iteration \( k + 1 \) will be satisfied if \( (X_{k+1}, e_{k+1}) \) is set to be \( (X_k, e_k) \). Therefore, the sequential problem of (3.9) has at least one pair of feasible solution and thus it is feasible. Consequently, from the last constraint of (3.9), we get that \( e_{k+1} \leq e_k \) and it can be rearranged as \( e_{k+1} - e_k \leq 1. \) This concludes the proof of at least a sublinear convergence rate of \( \{e_k\}. \)

Next, we prove that when \( \lim_{k \to +\infty} e_k = e^* = 0, \) the corresponding solution \( X_k \) from (3.9) is a local optimum of the original problem (3.8). Some definitions will be introduced here, including \( C = \{X \mid g(X) \leq 0, C = C \cap \{X \mid X \in S^n, X \geq 0 \}, P_k = \{P \mid P \in \mathbb{R}^{n \times m}, X_k^TV_k^TP = 0 \} \) and \( \Gamma_k = \{X \mid X = P^T, P \in P_k \}, \) where \( g(X) : \mathbb{R}^n \to \mathbb{R} \) are convex functions defining \( C. \) Then we have the following assumption.

**Assumption 9** When \( \lim_{k \to +\infty} e_k = 0, \) each sequential problem has only one optimum, that is, \( X_k \in C \cap \Gamma_k \) is the only minimizer of sequential problem at iteration \( k. \)

**Proposition 10** When \( \lim_{k \to +\infty} e_k = 0 \) and Assumption 9 holds, the corresponding solution \( \lim_{k \to +\infty} X_k = X^*, \) where \( X^* \) is a local minimizer of (3.8).

**Proof:** Since \( \lim_{k \to +\infty} e_k = 0, \) it implies \( V_k^TV_kX_kV_k \to 0. \) Then there exists \( P_k \in P_k \) such that \( X_k \to P_kP_k^T, \) which implies the rank of \( X_k \) is no greater than \( r. \) As \( V_k \) are the orthonormal eigenvectors of \( X_k = P_kP_k^T, \) we have \( V_k^TV_k = 0 \), i.e., \( \text{span}(V_k) = (P_k)^{-1}. \) As a result, \( P_{k+1} = (\text{span}(V_k))^T = (P_k)^{-1} = P_k, \) which leads to \( X_{k+1} = X_k. \) It implies the feasible regions for the sequential problem at iterations \( k \) and \( k + 1 \) are the same. From Assumption 9, \( \{X_k\} \) converges to a unique solution.

For the sequential problem (3.9) at iteration \( k, \) the optimality condition of the prime-dual problem pair are written as

\[
\nabla f(X) + \sum_{j=1}^p \mu_j \nabla g_j(X) + V_{k-1}S(1)^{k-1} - S(2) = 0
\]

\[
\lambda_j^*(g_j(X)) = 0, \quad \forall j = 1, \ldots, p
\]

\[
S(1)(e_kI_{n-r} - V_{k-1}X_kV_k) = 0, \quad S(2)X_k = 0
\]

where \( \lambda_j = [\lambda_j^{(1)}, \ldots, \lambda_j^{(m)}] \) are the dual variables. Problem (3.8) is equivalent to

\[
\min_{X \in S^n, P \in \mathbb{R}^{n \times m}} f(X), \quad s.t. \quad X = PP^T,
\]

whose optimality conditions are

\[
\nabla f(X) + \sum_{j=1}^p \mu_j \nabla g_j(X) - S = 0, \ (S + S^T)P = 0
\]

\[
\mu_j^*(g_j(X)) = 0, \quad \mu_j^* \geq 0, \quad \forall j = 1, \ldots, p
\]

where \( \mu = [\mu_1, \ldots, \mu^T] \) and \( S \) are the associated dual variables. When \( \lim_{k \to +\infty} e_k = 0, \) it implies \( V_{k-1}X_kV_k \to 0, \) where columns of \( V_k \) are orthonormal and rank of \( X_k \) is no more than \( r \) due to Lemma 6. As a result, there exists a \( P_k \in \mathbb{R}^{n \times r} \) such that \( X_k = P_kP_k^T, V_{k-1}P_k = 0 \) and \( S(2)P_k = 0 \) from (3.13).

By setting \( \mu = \lambda_k, S = -(S(1)^{k-1} - S(2)^{k-1}) \in \mathbb{R}^n, \) the prime-dual pair \( (\mu, S, X_k, P_k) \) satisfies (3.14), which indicates that \( (X_k, P_k) \) is a local minimizer of the original problem (3.8). If a local optimum of (3.8) is denoted by \( X^*, \) combined with the conclusion that \( \{X_k\} \) converges to a unique solution, we get \( \lim_{k \to +\infty} X_k = X^*. \)

**4 Applications and Simulation Results**

In this section, four representative applications are solved via the proposed IRM method. They are system identification, output feedback stabilization, and structured \( H_2 \) controller design problems. To verify the effectiveness and improved performance of IRM, results obtained from IRM are compared with those obtained from existing approaches. All simulation is run on a desktop computer with a 3.50 GHz processor and a 16.0 RAM.

**4.1 System Identification Problem**

Considering a linear time-invariant system with input \( u \in \mathbb{R}^m \) and output \( y \in \mathbb{R}^p, \) we want to identify the linear system through \( T \) samples of \( u(t) := (u_1(t), y_1(t)) \in (\mathbb{R}^m \times \mathbb{R}^p)^T. \) This identification problem is equivalent to finding a full
row matrix \( R = [R_0, R_1, \ldots, R_l] \) such that
\[
R_0 w(t) + R_1 w(t) + \cdots + R_l w(t + l) = 0, \quad \forall t = 1, 2, \ldots, T - l,
\]
where \( l \) is the lag of the identified model. Expression in (4.15) can be rewritten as \( RH_{l+1,T-l}(w) = 0 \), where
\[
H_{l+1,T-l}(w) = \begin{bmatrix}
w(1) & w(2) & \cdots & w(T - l) \\
w(2) & w(3) & \cdots & w(T - l + 1) \\
\vdots & \vdots & \ddots & \vdots \\
w(l + 1) & w(l + 2) & \cdots & w(T)
\end{bmatrix}.
\]
Since the row dimension of \( R \) is equal to the number of outputs, \( p \), of the model [44], the rank of \( H_{l+1,T-l}(w) \) is constrained by \( p \). As a result, the identification problem is equivalent to the following low-rank approximation problem expressed as,
\[
\min_w \| w - w_d \|_F^2 \quad \text{s.t.} \quad \text{rank}(H_{l+1,T-l}(w)) \leq r,
\]
where \( r = (m + p)(l + 1) - p \).

Using the benchmark system identification problems in database DAISY [32], we solve the above identification problem via IRM and compare with four existing methods, including subspace identification method (SUBID), deterministic balanced subspace algorithm (DETSS) [26], prediction error method (PEM) [20] and the software for Structured Low-Rank Approximation (SLRA) [23], named STLS. All results from the four existing methods can be found in [24]. The comparative results are demonstrated in Table (1) in terms of the relative misfit percentage, \( M_{rel} = 100\|w - w_d\|_F/\|w_d\|_F \).

The comparison verifies that identification results from IRM lead to a significant reduction on fitting error at the cost of more computational time.

### 4.2 Output Feedback Stabilization Problem

Consider the following linear time-invariant (LTI) continuous dynamical system
\[
\dot{x} = Ax + Bu, \quad y = Cx,
\]
where \( x \in \mathbb{R}^n \), \( u \in \mathbb{R}^m \), \( y \in \mathbb{R}^p \), \( A \in \mathbb{R}^{n \times n} \), \( B \in \mathbb{R}^{n \times m} \) and \( C \in \mathbb{R}^{p \times n} \). For system (4.16) and a given number \( k \leq n \), the stabilizing control law is defined as
\[
\dot{z} = Akz + B Ku, \quad u = Cz + D Ky,
\]
where \( z \in \mathbb{R}^k \), \( Ak \in \mathbb{R}^{k \times k} \), \( Bk \in \mathbb{R}^{k \times m} \), \( Ck \in \mathbb{R}^{m \times k} \), and \( Dk \in \mathbb{R}^{m \times p} \). Existence of the \( k \)th order control law can be determined by solving a RMP with linear matrix constraints [13].

**Lemma 1** (Corollary 1. in [13]) There exists a stabilizing output feedback law of order \( k \) if and only if the global minimum objective of the program
\[
\min_{R,S,\gamma} \text{rank}(\begin{bmatrix} \gamma R & I_n \\ I_n & \gamma S \end{bmatrix})
\]
\[
s.t. \quad AR + RA^T < BB^T \\
A^T S + SA < C^TC
\]
\[
\gamma > 0, \quad \begin{bmatrix} \gamma R & I_n \\ I_n & \gamma S \end{bmatrix} \succeq 0,
\]
is less than or equal to \( n + k \), where \( R \in \mathbb{S}^n \) and \( S \in \mathbb{S}^m \). By multiplying the first two linear matrix inequalities by \( \gamma \) and letting \( W_1 = \gamma R \), \( W_2 = \gamma S \), problem in (4.18) is converted to
\[
\min_{W_1, W_2, \gamma} \text{rank}(\begin{bmatrix} W_1 & I_n \\ I_n & W_2 \end{bmatrix})
\]
\[
s.t. \quad AW_1 + W_1 A^T < \gamma BB^T \\
A^T W_2 + W_2 A < C^TC
\]
\[
\gamma > 0, \quad \begin{bmatrix} W_1 & I_n \\ I_n & W_2 \end{bmatrix} \succeq 0,
\]
where constraints are linear functions with respect to \( W_1, W_2 \), and \( \gamma \). Using the above converted formulation, the stabilizing output feedback law can be determined via the proposed IRM approach. In the simulation, we assume \( n = 10 \), \( m = 4 \), \( p = 5 \) and randomly generate matrices \( A \), \( B \), and \( C \) for 100 simulation cases. We compare the results obtained from five methods, including IRM, nuclear norm heuristic method (denoted as Trace), log-det heuristic method (denoted as log-det), the iterative reweighted least square (IRLS) method, and trace penalty method (TPM). Among the 100 cases, IRM obtains lower rank solution than the other methods for 79 cases. In addition, IRM generates the same lowest rank solution compared to the best solution of the other method for 18 cases. However, IRM cannot find the best solution for 3 cases. Although the IRM method converges to a local optimum, the results indicate that it yields significantly improved performance in solving most of the RMPs. Figure 1 provides comparison of the computed rank obtained from the five methods for the 100 cases. Moreover, for one of the 100 cases, Table 2 shows the corresponding computation time, number of iteration, and optimal rank obtained from the five methods. It is obvious that IRM consumes more computation time to achieve a lower rank solution compared to the four existing methods. The trade-off between computation time and optimal value can be illustrated therein. Figure 2 demonstrates the value of \( \epsilon_k \) at each iteration of IRM for the same case, which verifies that convergence to the constrained rank is achieved within a few iterations under a stopping threshold \( \epsilon = 5\epsilon - 5 \).
Table 1
Comparative results for the system identification problem, where \( \Delta M_{rel}(\%) \) represents the percentage decrement of \( M_{rel} \) for IRM compared to SLRA and ‘dist. col.’ represents ‘distillation column’.

<table>
<thead>
<tr>
<th>NO.</th>
<th>Dataset name in [32]</th>
<th>( T )</th>
<th>( m )</th>
<th>( p )</th>
<th>( l )</th>
<th>( M_{rel} )</th>
<th>( M_{rel} )</th>
<th>( \Delta M_{rel} ) (%)</th>
<th>STLS</th>
<th>IRM</th>
<th>Time(s)</th>
<th>IRM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>dist. col.</td>
<td>90</td>
<td>5</td>
<td>3</td>
<td>1</td>
<td>0.0089</td>
<td>0.0306</td>
<td>0.0505</td>
<td>0.0029</td>
<td>0.021</td>
<td>27.6</td>
<td>0.6010</td>
</tr>
<tr>
<td>2</td>
<td>dist. col. n20</td>
<td>90</td>
<td>5</td>
<td>3</td>
<td>1</td>
<td>0.4309</td>
<td>0.1187</td>
<td>1.8574</td>
<td>0.0448</td>
<td>0.0257</td>
<td>42.6</td>
<td>0.1841</td>
</tr>
<tr>
<td>3</td>
<td>dist. col. n30</td>
<td>90</td>
<td>5</td>
<td>3</td>
<td>1</td>
<td>0.4357</td>
<td>0.1848</td>
<td>7.3600</td>
<td>0.0522</td>
<td>0.0322</td>
<td>38.3</td>
<td>0.6761</td>
</tr>
</tbody>
</table>

Fig. 1. Comparison of rank value, where \( \text{rank}_{\text{best}} \) represents the lowest rank value obtained from the five methods.

Table 2
Comparison of computation time, number of iteration, and computed rank for one case of the output feedback stabilization problem.

<table>
<thead>
<tr>
<th>Method</th>
<th>Time (s)</th>
<th>No Iter.</th>
<th>Time per Iter.</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trace</td>
<td>0.429</td>
<td>1</td>
<td>0.429</td>
<td>14</td>
</tr>
<tr>
<td>log-det</td>
<td>3.196</td>
<td>10</td>
<td>0.319</td>
<td>13</td>
</tr>
<tr>
<td>TPM</td>
<td>1.789</td>
<td>1</td>
<td>1.789</td>
<td>18</td>
</tr>
<tr>
<td>IRLS</td>
<td>7.163</td>
<td>20</td>
<td>0.358</td>
<td>17</td>
</tr>
<tr>
<td>IRM</td>
<td>57.862</td>
<td>16</td>
<td>3.616</td>
<td>12</td>
</tr>
</tbody>
</table>

Fig. 2. IRM convergence history for one case of the output feedback stabilization problem.

4.3 Structured \( H_2 \) Controller Design

Consider the following LTI system,

\[
\begin{align*}
\dot{x} &= Ax + B_1w + B_2u \\
z &= C_1x + D_{12}u, \quad y = C_2x,
\end{align*}
\]

(4.20)

where \( x \in \mathbb{R}^n \) is the state variable, \( w \in \mathbb{R}^{n_w} \) is the input noise, \( u \in \mathbb{R}^m \) is control input, \( z \in \mathbb{R}^{n_z} \) is the output to be regulated, and \( y \in \mathbb{R}^p \) is the measured output. The goal here is to design a static output feedback controller, \( u = Ky \), to minimize the \( H_2 \) norm while placing all eigenvalues of the closed-loop system in a stability region described by \( D(p, q, r) = \{ s \in \mathbb{C} | p + qs + q^*s + r \} < 0 \), where \( q^* \) and \( s^* \) are conjugate variables of \( q \) and \( s \), respectively. Additionally, structure constraints on \( K \) are considered. For example, if control input \( u_i \) does not depend on the measured output \( y_j \), element \( K_{ij} \) is set as 0. Thus, \( K \in \mathbb{C} \) is used to denote the structure constraint.

In the simulation example, the stability region is denoted by \( D(-1, 0, 1) \). Work in [18] formulates the structured \( H_2 \) controller design problem as a RCOP in the form of

\[
\begin{align*}
\min_{P, \mu, W, K} & \quad \text{trace}(B_1^T PB_1) \\
\text{s.t.} & \quad P > 0, W > 0, \mu > 0, K \in \mathbb{K} \\
& \quad \begin{bmatrix}
    pP & qP & 0 & A_{cl}^T & C_{cl}^T \\
    q^*P & rP & 0 & -I_n & 0 \\
    0 & 0 & I_{n_z} & 0 & -I_{n_z} \\
    A_{cl} & -I_n & 0 & \mu I_n & 0 \\
    C_{cl} & 0 & -I_{n_z} & 0 & \mu I_{n_z}
\end{bmatrix} < W
\end{align*}
\]

(4.21)

where \( A_{cl} = A + B_2KC_2, C_{cl} = C_1 + D_{12}KC_2 \). To verify the improved performance of IRM in solving this type of problem, the penalty function method (PFM) introduced in [18] is used to solve the same problem. The simulation example assumes that \( m = 5, n = 8, n_w = 5, n_z = 2, \) and \( p = 5 \). In addition, elements of matrices \( A, B_1, B_2, C_1, C_2 \) and \( D_{12} \) are randomly generated within a uniform distribution in the range of \([-1, 1]\). Under these assumptions, 50 cases are generated among which 43 have a feasible solution, as the random matrix generation does not guarantee a feasible solution for this type of problem. We claim that for any case if neither IRM or PFM generates a feasible solution, then this case is infeasible. Moreover, we apply a commercial solver LMIRANK based on [33] to further verify the feasibility. It is verified that LMIRANK also fails for all the 7 cases when IRM or PFM cannot generate a feasible solution.

Among the 43 feasible cases, IRM converges for all of the cases while PFM converges for 19 cases. In addi-
tion, IRM requires 51 iterations on average to achieve convergence while PFM requires 135 iterations on average. Moreover, among the 19 cases where both methods converge, IRM outperforms PFM in terms of objective value for 15 cases. For the remaining 4 cases, IRM yields slightly larger objective value. Comparison of objective values for the 19 cases is shown in Fig. 3. And comparison of penalty term, objective value, and weighting factors for results of one selected case generated from both methods are demonstrated in Fig. 4, where penalty term for IRM is \( e_k \) with \( w_k \) assigned as its corresponding weighting factor, as described in (3.9). For PFM, the penalty term is \( \text{trace}(V_{k-1}^T X_k V_{k-1}) \) with \( \mu_k = \tau \mu_{k-1} \) assigned as its corresponding weighting factor and more details can be found from [18]. Moreover, to verify robustness of IRM in terms of local convergence under random initial values, 50 simulation results are generated using random initial values for the same selected case in Fig. 4. All results from IRM converge while none of them converges using random initial values for PFM. It indicates that random initial guess will not affect the convergence property of IRM. However using random initial guess, instead of initial from relaxation solution, will not improve convergence of PFM. Figure 5 demonstrates the objective value distribution of the 50 results with a covariance of 0.29. In summary, IRM demonstrates improved performance in terms of convergence rate, robustness, and cost function values for most of the cases.

5 CONCLUSIONS

This paper established a uniform formulation for rank-constrained optimization problems (RCOPs) and rank minimization problems. An iterative rank minimization (IRM) method is proposed to solve general RCOPs. The sublinear convergence of IRM to a local optimum is proved through the duality theory and the Karush-Kuhn-Tucker conditions. We apply IRM to several representative applications, including system identification, output feedback stabilization problem, and structured \( H_2 \) controller design problems. Comparative results are provided to verify the effectiveness and improved performance of IRM in terms of convergence rate, robustness to initial guess, and cost function. Future work will focus on seeking the global optimal solution for RCOPs.

Acknowledgements

The authors would like to thank the anonymous reviewers for their helpful and constructive comments and NSF (grant #ECCS-1453637) for sponsoring the research work.

References


