Proceedings of the American Control Conference
Arlington, VA June 25-27, 2001

Rank Minimization Approach for Solving BMI Problems with Random Search

Soichi Ibaraki
Department of Precision Engineering,
Kyoto University, Kyoto 606-8501, Japan
ibaraki@prec.kyoto-u.ac.jp

Masayoshi Tomizuka
Department of Mechanical Engineering,
University of California at Berkeley, Berkeley, CA 94720-1740
tomizuka@me.berkeley.edu

Abstract

This paper presents the rank minimization approach to solve general bilinear matrix inequality (BMI) problems. Due to the *NP*-hardness of BMI problems, no proposed algorithm that globally solves general BMI problems is a polynomial-time algorithm. We present a local search algorithm based on the semidefinite programming (SDP) relaxation approach to indefinite quadratic programming, which is analogous to the well-known relaxation method for a certain class of combinatorial problems. Instead of applying the branch and bound (BB) method for global search, a linearization-based local search algorithm is employed to reduce the relaxation gap. Furthermore, a random search approach is introduced along with the deterministic approach. Four numerical experiments are presented to show the search performance of the proposed approach.

1 Introduction

This paper considers an algorithm to solve BMI (bilinear matrix inequality) problems of the following form (Safonov et al. [1]):

$$\text{Find} \quad x = \{x_i\}_{i=1,\cdots,N} \in \mathbb{R}^N$$
$$\text{such that} \quad F_0 + \sum_{i=1}^{N} x_i F_i + \sum_{i=1}^{N}\sum_{j=1}^{N} x_i x_j F_{ij} \prec 0 \quad (1)$$

where F_0, F_i $(i = 1,\cdots,N)$, and F_{ij} $(i,j = 1,\cdots,N)$ are $m \times m$ real, constant symmetric matrices. $A \prec 0$ denotes that a matrix A is symmetric and negative definite. Similarly, $A \preceq 0$, $A \succ 0$ and $A \succeq 0$ denote that a symmetric matrix A is negative semidefinite, positive definite and positive semidefinite, respectively.

General BMI problems are not convex optimization problems due to the bilinear terms in the constraint (1) and, therefore, can have multiple local solutions. BMI problems are proven to be *NP*-hard (Toker and Özbay [2]), which means that any algorithms that globally solve general BMI problems are quite likely non-polynomial time algorithms.

In recent years, considerable research efforts have been devoted to the development of algorithms to solve general BMI problems. Most of the algorithms found in the literature that claim the applicability to control-related problems of practical size are local search algorithms. One of the simplest approaches is an iterative algorithm solving alternating LMIs at each step, making use of the bilinear property of the problem. Another simple approach is based on the linearization; under an assumption of small search steps, one can approximate a BMI problem by an SDP (semidefinite programming) problem by using the first-order perturbation approximation [3]. The SDP problem can be written in the following canonical form:

$$\text{Find} \quad x = \{x_i\}_{i=1,\cdots,N} \in \mathbb{R}^N$$
$$\text{such that} \quad F_0 + \sum_{i=1}^{N} x_i F_i \prec 0 \qquad (2)$$

where F_0, F_i $(i = 1,\cdots,N) \in \mathbb{R}^{m\times m}$ are constant symmetric matrices. SDP problems are convex optimization problems that can be solved in a polynomial time by using well-developed interior point algorithms (e.g. [4]).

It is, however, highly likely for such local search approaches to fail to reach the global optimum due to the nonconvex nature of BMI problems. Goh et al. [5] showed this aspect by using a small BMI problem as an example (see Section 3.1).

Most of global search algorithms found in the literature are variations of the Branch and Bound (BB) method based on different formulations of BMI problems [5, 6, 7, 8, 9, 10]. Although the computational efficiency is a major focus for all of those works, none of global search algorithms are polynomial-time algorithms due to the *NP*-hardness of BMI problems. Therefore, their applicability to problems of practical size is questionable.

The approach proposed in this paper is outlined as follows. First, it is shown that general BMI constraints can be reformulated as a combination of LMI constraints and a rank constraint. If the rank constraint is dropped, the problem becomes a convex optimization problem, whose optimal objective value gives a lower bound for the original BMI problem. This approach is analogous to the well-known SDP relaxation approach to a certain class of combinatorial problems. Although the approximate solution of the relaxed problem can be used in BB methods, this paper employs a linearization-based local search algorithm to reduce the relaxation gap. The linearization-based rank minimization approach is analogous to the algorithm presented by El Ghaoui et al. [11] to solve reduced-order H_∞ controller synthesis problems.

Furthermore, a random search approach can be straightforwardly applied along with the deterministic approach. In recent years, random searches have attracted more attention in the field of nonconvex optimization as a tool that has a potential to substantially enhance the computational efficiency [12, 13]. The proposed formulation of BMI problems is propitious to the introduction of random search approaches.

The proposed approach is a local search algorithm and, therefore, there is no guarantee that it finds the global solution. It is, however, based on a completely different formulation of BMI problems from conventional, simpler local search algorithms, and it is claimed that the proposed approach can more likely find the global solution than conventional local search approaches in practice. Considering that any global search algorithm is a non-polynomial time algorithm due to the *NP*-hardness of BMI problems, the proposed approach is more practical than any existing global search algorithm from the viewpoint of the computational efficiency. It is more reliable than conventional, simpler local search algorithms from the viewpoint of the likelihood of finding the global solution.

The practical importance of solving BMI problems has been particularly recognized in the field of optimal controller design. For example, it is well known that the H_∞ optimization problem of full-order controllers can be formulated as an SDP problem (Gahinet and Apkarian [14]). A critical limitation of the LMI-based H_∞ controller synthesis is that it allows no additional constraint to be imposed on the problem; the closed-loop H_∞ norm constraint must be the only constraint imposed to the problem in order for it to be globally solvable by convex optimization. The BMI framework offers a unified approach to formulate "generalized" H_∞ optimization problems with arbitrary constraints. See e.g. [5] for further details.

The remainder of this paper is organized as follows. The proposed algorithm is presented in Section 2. Section 3 presents four numerical experiments to show the search performance of the proposed approach.

2 Rank Minimization Approach for Solving BMI Problems with Random Search

2.1 Rank Minimization Approach for Solving BMI Problems

This paper considers the problem given in Eq. (1). The problem in fact belongs to the class of indefinite quadratic programming, which includes BMI problems. First, the following lemma shows that the problem (1) can be equivalently rewritten as a rank minimization problem subject to LMI constraints.

Lemma 1 *The indefinite quadratic programming problem* (1) *is equivalent to the following problem:*

$$\text{Find} \quad x = \{x_i\}_{i=1\cdots N} \in \mathbb{R}^N \quad \text{and} \quad X = \{X_{ij}\}_{i,j=1\cdots N} \in \mathbb{R}^{N\times N}$$

$$\text{such that} \quad F_0 + \sum_{i=1}^{N} x_i F_i + \sum_{i=1}^{N}\sum_{j=1}^{N} X_{ij}F_{ij} \prec 0 \qquad (3)$$

$$\begin{bmatrix} X & x \\ x^T & 1 \end{bmatrix} \succeq 0 \qquad (4)$$

$$\text{rank}(X) = 1\,. \qquad (5)$$

The above problem can be seen as a minimization problem of the objective function, $rank(X)$, under LMI constraints (3) and (4). The rank minimization problems are generally quite hard to solve (a special case which can be equivalently transformed into an SDP problem was discussed by Mesbahi [15]). The rank minimization problems appear, however, in many control-related optimization problems such as reduced-order H_∞ controller synthesis and scaled-H_∞ optimization with constant scaling matrices [16]. Considerable research efforts have been devoted to develop algorithms to solve them. Most of the algorithms are local search algorithms, either by coordinate descent approaches (e.g. [17]) or the linearization approach [11]. We employ an analogous linearization-based local search approach to solve the problem (3)~(5).

Since the constraint (4) assures that $tr(X) - x^T x \geq 0$ and $tr(X) - x^T x = 0$ if and only if $X = xx^T$, there exists a solution, (x, X), that satisfies the constraints (3)~(5) if and only if the optimal value of the following problem is zero:

$$\min_{x,X} \ \text{tr}(X) - x^T x \qquad \text{subject to (3) and (4)} \qquad (6)$$

where $tr(X)$ denotes the trace of a square matrix X. By linearizing the objective function, the following descent method to find a local optimum of the problem (6) is obtained.

Algorithm 1

1. *Find a feasible set* $(x^{(0)}, X^{(0)})$ *that satisfies the constraints* (3) *and* (4). *Set* $k = 1$. *If there is no feasible solution, then the problem is infeasible.*
2. *Solve the following convex optimization problem for* $x^{(k)} \in \mathbb{R}^N$ *and* $X^{(k)} \in \mathbb{R}^{N\times N}$:

$$\min_{x^{(k)},X^{(k)}} \ \text{tr}(X^{(k)}) - 2x^{(k-1)\,T}x^{(k)} \quad \text{subject to (3) and (4)}\,. \qquad (7)$$

3. *Set* $k = k + 1$ *and repeat Step 2 until convergence.*

It is easy to show that this algorithm converges.

Lemma 2 *The sequence:*

$$t_k := \text{tr}\left(X^{(k)}\right) - 2x^{(k-1)\,T}x^{(k)} + x^{(k-1)\,T}x^{(k-1)}\,, \qquad (8)$$

where $k = 1, 2, \cdots$, *is bounded below by zero and non-increasing. Thus, the sequence* $\{t_k\}$ $(k = 1, 2, \cdots)$ *converges to some value,* $t_{opt} \geq 0$. *Equality holds if and only if* $X^{(k)} = x^{(k)}x^{(k)\,T}$ *as* $k \to \infty$.

The proof is straightforward and thus omitted for the brevity.

2.2 SDP Relaxation Approach to BMI Problems and Combinatorial Problems

Consider the following minimization problem subject to BMI constraints:

$$\min_{x\in\mathbf{R}^N} \gamma \quad \text{such that} \quad F_0 + \sum_{i=1}^{N} x_i F_i + \sum_{i=1}^{N}\sum_{j=1}^{N} x_i x_j F_{ij} - \gamma I \prec 0 \qquad (9)$$

where I denotes the identity matrix of the same size as F_0. Then, compare with the following problem:

$$\min_{x\in\mathbf{R}^N, X\in\mathbf{R}^{N\times N}} \gamma \quad \text{such that}$$

$$F_0 + \sum_{i=1}^{N} x_i F_i + \sum_{i=1}^{N}\sum_{j=1}^{N} X_{ij}F_{ij} - \gamma I \prec 0$$

$$\begin{bmatrix} X & x \\ x^T & 1 \end{bmatrix} \succeq 0 \qquad (10)$$

Notice that the above problem is an SDP problem and, therefore, can be globally solved by convex optimization. From Lemma 1, it is clear that the optimal solution for the problem (10) gives a lower bound for the original BMI problem (9) (Notice that if the rank constraint, $rank(X) = 1$, is added to the problem (10), then the problem (10) becomes equivalent to the problem (9)).

An analogous relaxation approach is well known to find an approximate solution for a certain class of combinatorial optimization problems. The Max-Cut problem is to find a cut of maximum total weight in an edge-weighted undirected graph (see e.g. [18]). This problem is one of the original *NP*-complete problems and thus hard to solve. It can be formulated as an indefinite quadratic problem in binary variables as follows:

$$\max_x x^TQx+2b^Tx+d \qquad \text{subject to} \quad x\in\{-1,1\}^N \quad (11)$$

where $Q = Q^T \in \mathbb{R}^{N\times N}$ and $b, d \in \mathbb{R}^N$ are given. Using the fact that $x \in \{-1,1\}^N$ can be written as $x_i^2 = 1$ $(i = 1,\ldots,N)$, the above problem can be equivalently transformed to:

$$\max_{x\in\mathbf{R}^N, X\in\mathbf{R}^{N\times N}} \mathrm{tr}(QX) + 2b^Tx + d \quad \text{subject to}$$

$$\begin{bmatrix} X & x \\ x^T & 1 \end{bmatrix} \succeq 0, \quad \mathrm{diag}(X) = e, \quad \mathrm{rank}(X) = 1 \quad (12)$$

where $e \in \mathbb{R}^N$ is the vector of all ones. If the rank constraint in the problem (12) is dropped, then the problem can be solved by convex optimization, and it gives an upper bound for the original problem (11). This SDP relaxation approach method proposed by Goemans and Williamson [19] is widely accepted as the best current approximation approach to the Max-Cut problem. Goemans and Williamson [19] proposed a randomized algorithm to recover the optimal solution from the approximate solution (x, X). It is guaranteed to produce a solution with the expected value at worst 14% smaller than the true optimum.

The clear similarity of the formulation (12) of the Max-Cut problem and the formulation (3)$\sim$(5) of the BMI problem implies that the analogous SDP relaxation approach also offers a good approximation for BMI problems. The mathematical justification of the SDP relaxation approach to BMI problems can be found in Konishi and Shin [20], where three relaxation approaches (the Lagrange relaxation, the relaxation using nonconvex quadratic inequalities, and the SDP relaxation) are applied to the problem (1), and it is shown that the SDP relaxation gives the best lower bound among them. This strongly justifies the approach presented in this paper.

Notice that the application of the SDP relaxation approach (10) to BB methods is straightforward; the SDP relaxation approach gives a tight lower bound at each region of the branching space (see [9]). However, this paper employs the local search algorithm presented in Algorithm 1 to reduce the relaxation gap, considering that any variations of the BB method are non-polynomial time algorithms due to the *NP*-hardness of BMI problems.

2.3 Random Search Algorithm

The algorithm proposed by Goemans and Williamson [19] to solve the Max-Cut problem includes random searches. Other nontrivial examples of randomized algorithms based on the SDP relaxation include the application to the Graph Coloring problem [21]. In both cases, the SDP relaxation is followed by an algorithm examining several random draws from the distribution defined by the solution of the relaxed problem.

A simple random search approach, which is analogous to the one used by Frazzoli et al. [13], can be applied in Algorithm 1 along with the deterministic approach. After each step of solving the problem (7), random samples are drawn from the Gaussian distribution with the mean $x^{(k)}$ and covariance $\alpha\left(X^{(k)} - x^{(k)}x^{(k)\,T}\right)$, where $(x^{(k)}, X^{(k)})$ is the solution for the problem (7) at the k-th step, and $\alpha > 0$ is a scalar constant. Notice that if $X = xx^T$, then the distribution consists of a unique point, which is the optimal solution of the original BMI problem (1). If a solution of the original problem (1) is found by this search, then the algorithm is terminated. Otherwise, Step 2 is repeated.

A major disadvantage of the formulation (3) is that it introduces a slack variable matrix X, which has $\frac{1}{2}N(N+1)$ parameters. When N is large, the increase of the number of variables may significantly slow down the algorithm. On the other hand, the direct applicability of random searches is a strong advantage of the proposed formulation, and so is the fact that the SDP relaxation approach generally gives a very good lower bound for the original problem. Although the algorithm proposed in the previous section is a local search, the combination of the linearization-based descent method and random searches more likely finds the global solution in many practical applications.

3 Numerical Experiments

In this section, four numerical experiments are presented to show the search performance of the proposed algorithm for solving BMI problems. All experiments were carried out by using MATLAB on a PC with CPU Pentium 450 *MHz*. The SDP solver engine *SP* [22] and its MATLAB interface *LMITOOL* [23] were used for SDP problems.

3.1 A Simple BMI Problem over Two Variables

The first problem is a very simple BMI problem over two variables, which was shown by Goh et al. [5] as an example of BMI problems that had multiple local minimums. The problem is given as follows:

$$\min_{x=[x_1,\, x_2]^T} \gamma$$

$$\text{subject to} \quad F(x) - \gamma I \prec 0$$

$$x_1 \in [-0.5,\, 2], \quad x_2 \in [-3,\, 7] \quad (13)$$

where $F(x) = F_0 + x_1F_1 + x_2F_2 + x_1x_2F_{12}$ and F_0, F_1, F_2, and F_{12} are constant 3×3 symmetric matrices. See [5] for their values.

There are three local minimums in the domain, as can be observed from the contour plot of the greatest eigenvalue of $F(x_1, x_2)$ (denoted by $\bar{\lambda}\{F(x_1, x_2)\}$) shown in Figure 1. Therefore, conventional local search algorithms may not be able to find the global optimum, depending on the initial point.

First, the SDP relaxation approach presented in Section 2.2 was applied to compute a lower bound for this problem. The relaxed problem (10) gave the optimal objective value $\gamma^{(0)} = -1.000$ with the solution $x^{(0)} = (1.00, 0.00)$ (shown in Figure 1 by ◯ #0). The global optimum of the problem (13) is known to be $x^* = (1.0488, 1.4179)$ with the corresponding optimal objective value $\gamma^* = \bar{\lambda}\{F(x^*)\} = -0.9565$ [5]. Notice that the SDP relaxation approach gave quite a tight lower bound. However, its solution, $x^{(0)}$, achieved only $\bar{\lambda}\{F(x^{(0)})\} = 5.919$, which is not sufficiently close to the global optimum.

Then, Algorithm 1 was applied starting from this initial point, $x^{(0)}$, to reduce $tr(X^{(k)}) - x^{(k)T}x^{(k)}$ to zero. Notice that the proposed approach can be only applied to the feasibility problem in the form (1). For this problem, the objective index, γ, in the problem (13) is fixed to -0.9565, which is equal to the global optimum. The objective is to find the global solution, x^*, that satisfies the constraint (13) with γ fixed to this value.

◯ #1 ~ ◯ #3 in Figure 1 indicate the optimal solutions, $x^{(k)}$, at the k-th step of the rank minimization approach (7). After three iterations, $tr(X^{(k)}) - x^{(k)T}x^{(k)}$ became approximately zero and $x^{(3)}$ achieved $\bar{\lambda}\{F(x^{(3)})\} = -0.9565$. The total computational time for the initial search and three iterations was only 0.14 *sec.*

Since the problem is small and thus each step of solving the problem (7) did not computationally cost much, a random search was not used. However, to show the effectiveness of the random search presented in Section 2.3, 50 random samples were drawn from the Gaussian distribution with the mean $x^{(1)}$ and the covariance $X^{(1)} - x^{(1)}x^{(1)T}$ after the first iteration (see Figure 2). The best objective value among 50 random samples was $\bar{\lambda}\{F(x)\} = -0.9555$ (the mean value, $x^{(1)}$, gave $\bar{\lambda}\{F(x^{(1)})\} = 2.113$). Although it did not reach the global optimum, this result shows the effectiveness of random search to some extent. The computational time to compute $\bar{\lambda}\{F(x)\}$ for 50 random samples was only 0.14 *sec.*

Finally, it should be noted that it is not necessarily because the initial point, $x^{(0)}$, was already close to the global optimum that the sequence, $\{x^{(k)}\}$, converged to the global optimum. Notice that the objective function in the rank minimization approach (7) is not $\bar{\lambda}\{F(x)\}$ and, therefore, the existence of multiple local minimums for $\bar{\lambda}\{F(x)\}$ does not necessarily mean that the problem (7) also has multiple local minimums. For this particular problem, we verified that Algorithm 1 found the global minimum by at most four iterations for any of 36 different initial points in the domain.

3.2 Randomly Generated BMI Problems

The proposed algorithm was then tested to solve randomly generated BMI problems of the following form:

$$\min_{x \in \mathbf{R}^{N_1},\, y \in \mathbf{R}^{N_2}} \gamma \qquad \text{such that}$$

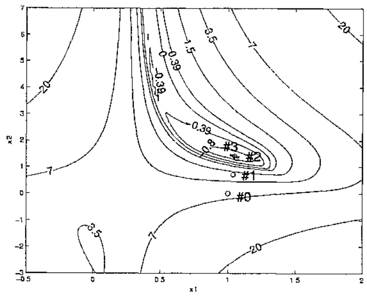

Figure 1: Contour plot of the greatest eigenvalue of $F(x_1, x_2)$ with the trajectory of the rank minimization approach (◯ #0 ~ ◯ #3)

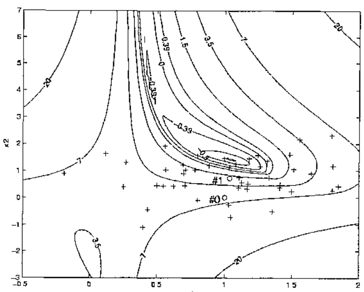

Figure 2: 50 random samples ("+") after one step of the rank minimization approach

$$F_0 + \sum_{i=1}^{N_1} x_i F_{i0} + \sum_{j=1}^{N_2} y_j F_{0j} + \sum_{i=1}^{N_1}\sum_{j=1}^{N_2} x_i y_j F_{ij} - \gamma I \prec 0 \quad (14)$$

where F_0, F_{i0}, F_{0j}, and F_{ij} ($i = 1, \cdots, N_1$; $j = 1, \cdots, N_2$) $\in \mathbb{R}^{m \times m}$ are randomly generated constant symmetric matrices. Each entry of the coefficient matrices are randomly chosen from -10 to 10. All variables, x_i ($i = 1, \cdots, N_1$) and y_j ($j = 1, \cdots, N_2$), are restricted to [0.01, 100]. Similar tests were conducted by Tuan et al. [9]. Notice that there is no quadratic term in the constraint (14), i.e. the above problem is strictly a BMI problem and less general than the problem (1).

First, for each given problem, the global minimum was computed by using the BB method proposed by Tuan et al. [9]. A major advantage of this method is that it only performs the branching operations on either x-space of N_1 dimension or y-space of N_2 dimension that has less dimension than the other. By making use of the bilinear property of BMI problems, it reduces the dimension of the branching space significantly than conventional general-purpose BB methods such as the one presented by Goh et al. [5]. The BB operations were iterated until the global minimum is found with 1% tolerance.

Then, Algorithm 1 was applied to solve the same problem (14) with γ fixed to the global optimum. This test was conducted for three different problem settings: 1) F_0, $F_{ij} \in \mathbb{R}^{3\times 3}$ and x, $y \in \mathbb{R}^3$ (100 problems), 2) F_0, $F_{ij} \in \mathbb{R}^{6\times 6}$ and x, $y \in \mathbb{R}^3$ (100 problems), and 3) F_0, $F_{ij} \in \mathbb{R}^{3\times 3}$ and x, $y \in \mathbb{R}^5$ (35 problems).

Table 1 shows the average number of iterations and computational time that the BB method and Algorithm 1 respectively had to perform to reach the global optimum

Table 1: The average number of iterations and computational time for the BB method and the proposed algorithm to solve randomly generated BMI problems (*: does not include cases that failed)

Tests		BB Method			Algorithm 1		
Size	Number of problems	Number of successes	Average iterations	Average time (sec)	Number of successes	Average iterations	Average time (sec)
$F_{ij} \in \mathbb{R}^{3\times 3}$, $x, y \in \mathbb{R}^3$	100	100(100%)	32.20	3.103×10^2	84(84%)	14.95*	8.900*
$F_{ij} \in \mathbb{R}^{6\times 6}$, $x, y \in \mathbb{R}^3$	100	100(100%)	100.85	1.138×10^3	81(81%)	54.69*	80.97*
$F_{ij} \in \mathbb{R}^{3\times 3}$, $x, y \in \mathbb{R}^5$	35	35(100%)	40.24	2.874×10^4	30(85.7%)	8.100*	18.52*

in total 235 problems. In total 40 problems (17.0% of all problems), Algorithm 1 failed to reach the global minimum. The approach proposed in this paper is a local search algorithm, and thus there is no guarantee that it finds the global minimum. For practical control-related problems, however, it can be used at least to improve the control performance, as will be shown in the next experiment. Although the BB method proposed by Tuan et al. [9] is more efficient for BMI problems than conventional general-purpose BB methods, it still requires excessively heavy computation. Computational loads increase exponentially as the size of problem becomes larger.

3.3 Mixed H_2/H_∞ Controller Design

The next example presents the application of the proposed algorithm to more practical control-related optimization problems. The problem is taken from [3].

Consider the following plant model:

$$\dot{x} = Ax + Bu + B_1 w, \quad z_1 = C_1 x + D_1 u, \quad z_2 = C_2 x + D_2 u. \tag{15}$$

The system dimensions are $A \in \mathbb{R}^{3\times 3}$, $B \in \mathbb{R}^{3\times 1}$, $B_1 \in \mathbb{R}^{3\times 2}$ and C_1,$C_2 \in \mathbb{R}^{1\times 3}$. See [3] for their values.

The objective is to design a state feedback control, $u = Kx$, such that the closed-loop H_2 norm from w to z_2 is minimized, while the H_∞ norm from w to z_1 is kept less than the given level, $\gamma > 0$. This problem can be formulated as a BMI problem as follows.

$$\min \eta^2 \quad \text{such that}$$

$$\begin{bmatrix} (A+BK)^T P_1 + P_1(A+BK) & P_1 B_1 & (C_1 + D_1 K)^T \\ B_1^T P_1 & -\gamma I & 0 \\ C_1 + D_1 K & 0 & -\gamma I \end{bmatrix} \prec 0$$

$$\begin{bmatrix} (A+BK)^T P_2 + P_2(A+BK) & P_2 B_1 \\ B_1^T P_2 & -I \end{bmatrix} \prec 0$$

$$\begin{bmatrix} P_2 & (C_2 + D_2 K)^T \\ C_2 + D_2 K & Z \end{bmatrix} \succ 0$$

$$\mathrm{tr}(Z) < \eta^2, \quad P_1 \succ 0, \quad P_2 \succ 0. \tag{16}$$

First, the above constraints were transformed into the form given in Eq. (1). Then, the initial point, $x^{(0)}$, was computed. For this problem, instead of using the rank relaxation approach given in Section 2.2, the following heuristic method was used to find a "better" initial point.

Most of practical control-related optimization problems can be solved by convex optimization if the design objective is relaxed. For this problem, if a common Lyapunov matrix for the H_2 and H_∞ problems is assumed (i.e. $P_1 = P_2$), then the problem can be equivalently transformed into an LMI problem (see e.g. [24]). This LMI approach gave the suboptimal solution that achieved $\eta^2 = 2.545$. We used this solution as the initial point for Algorithm 1.

Recall that the proposed approach can be applied only to the feasibility problem to find a solution set, (P_1, P_2, K, Z), with both η and γ fixed. Therefore, iterations over η are required to solve the minimization problem (16). After three iterations over η from $\eta^2 = 2.545$ (totally 11 iterations of solving the problem (7)), the solution set was found that achieved $\eta^2 = 1.875$. The total computational time was 228.23 sec.

This solution may or may not be the global optimum. Although there is no guarantee that the proposed approach finds the global optimum, it can be at least used to improve the controller performance, as shown in this experiment. This "path-following" approach is commonly used with most of local search algorithms to solve control-related BMI problems [3].

3.4 Simultaneous State-Feedback Stabilization

The next problem is also taken from [3]. Consider the following three linear time-invariant plants:

$$\dot{x} = A_k x + B_k u, \quad k = 1, 2, 3 \tag{17}$$

The system dimensions are $A_k \in \mathbb{R}^{3\times 3}$, $B_k \in \mathbb{R}^{3\times 2}$ ($k = 1, \cdots, 3$). See [3] for their values.

The goal is to find $K = \{K_{ij}\}_{i=1,2;\, j=1,2,3} \in \mathbb{R}^{2\times 3}$ satisfying $|K_{ij}| \leq K_{ij,max}$ ($i = 1, 2$; $j = 1, 2, 3$) such that the state feedback control, $u = Kx$, stabilizes all three plants and that the decay rate of each closed-loop system is maximized, where $K_{ij,max}$ ($i = 1, 2$; $j = 1, 2, 3$)> 0 are constant. This problem can be written as the following BMI problem:

$$\max_{P_k \in \mathbf{R}^{3\times 3}\ (k=1,2,3),\ K \in \mathbf{R}^{2\times 3}} \alpha \quad \text{such that}$$

$$|K_{ij}| \leq K_{ij,max}$$

$$(A_k + B_k K)^T P_k + P_k(A_k + B_k K) \prec -2\alpha P_k \tag{18}$$

$$P_k \succ 0, \quad k = 1, 2, 3$$

The similar "path-following" method as presented in the previous section was used to reduce α. Starting from $\alpha = -2.05$ (if $K = 0_{2\times 3}$, then the problem (18) has trivially a feasible solution for $\alpha \geq -2.05$), α is reduced at each step. The feasible solution set, (K, P_1, P_2, P_3), is computed for each given α by using Algorithm 1. The optimal solution set is used as an initial point for the next step of reducing α.

Table 2: Number of iterations that Algorithm 1 requires to reach the solution at each step of reducing α ("+" denotes that the solution was obtained by random searches)

α	-2.05	-1.85	-1.50	-1.00	-0.50	-0.10
iterations	-	1	1+	2+	4	10

0	0.10	0.30	0.50	0.60	0.70	0.75
3+	2+	8+	12+	7+	9+	5

0.80	0.85	0.90	0.95	1.00	1.05	total
6	5	5+	5+	4	5+	94

Table 2 shows the number of iterations required at each step. For this problem, random searches proposed in Section 2.3 were performed after each iteration (100 points were sampled after each iteration of Step 2 in Algorithm 1). "+" in Table 2 indicates that the solution was obtained by random searches. Unlike the previous problem, the proposed algorithm required too many iterations (94 iterations) to reach the optimum, $\alpha = 1.05$. This result is the same as the one obtained in [3].

4 Concluding Remarks

In this paper, the rank minimization approach to solve BMI problems was presented. The proposed algorithm is based on the SDP relaxation approach to indefinite quadratic programming problems, which is analogous to the well-known relaxation method for a certain class of combinatorial problems. We employ the linearization-based local search algorithm to reduce the relaxation gap. A direct applicability of a random search is another strong advantage of the formulation presented in this paper.

Considering the *NP*-hardness of BMI problems, the proposed algorithm is more practical than any existing global search approaches from the viewpoint of the computational efficiency. Four numerical experiments were presented to show the search performance of the proposed approach. Although it shows better performances for the problems presented in Section 3.1 and 3.3, it is not always guaranteed to find the global minimum, as can be observed by the results in Section 3.2. And also, the result in Section 3.4 was not satisfactory from the viewpoint of the convergence speed. Although its performance to find the global solution may not be satisfactory in some cases, it can be at least used to improve the controller performance by applying the "path-following" approach presented in Section 3.3 and 3.4. Finally, we note that the application of the proposed approach to BB methods for global search is straightforward.

References

[1] M. G. Safanov, K. C. Goh, and J. H. Ly, "Control System Synthesis via Bilinear Matrix Inequalities, " *Proc. of the American Control Conf.*, Baltimore, MD, pp.45-9, June 1994

[2] O. Toker and H. Özbay, "On the NP-hardness of solving bilinear matrix inequalities and simultaneous stabilization with static output feedback," *Proc. of the American Control Conference*, Seattle, WA, p.2525-6, 1995

[3] A. Hassibi, J. How, and S. Boyd, "A path-following method for solving BMI problems in control," *Proc. of the American Control Conference*, San Diego, CA, pp.1385-9, 1999

[4] L. Vandenberghe and S. Boyd, "Semidefinite programming," *SIAM Review*, vol. 38, No. 1, pp. 49-95, Mar. 1996

[5] K.-C. Goh, M. G. Safonov, G. P. Papavassilopoulos, "A global optimization approach for the BMI problem," *Proc. of the 33rd Conf. on Decision and Control*, Lake Buena Vista, FL, pp.2009-14, Dec. 1994

[6] R. Horst and H. Tuy, *Global optimization: Deterministic approaches*, Springer-Verlag, Berlin, second edition, 1993

[7] E. Beran, L. Vandenberghe, and S. Boyd, "A global BMI algorithm based on the generalized Benders decomposition, " *Proc. of the European Control Conference*, July 1997

[8] S. -M. Liu and G. P. Papavassilopoulos, "Numerical experience with parallel algorithms for solving the BMI problem," *Proc. of the 13th World Congress of IFAC*, San Francisco, CA, Vol. D, pp.387-92, 1996

[9] H. D. Tuan, P. Apkarian, S. Hosoe, and H. Tuy, "D. C. optimization approach to robust control: feasibility problems," *Int. J. of Control*, Vol. 73, No. 2, pp.89-104, Jan. 2000

[10] Y. Wakasa, "Global optimization algorithms for BMI problems," *Trans. of the Inst. of Systems, Control and Information Engineers*, Vol. 43, No. 5, pp.258-265, 1999 (in Japanese)

[11] L. El Ghaoui, F. Oustry and M. AitRami, "A cone complementarity linearization algorithm for static output-feedback and related problem," *IEEE Trans. on Automatic Control*, vol. 42, No. 8, pp. 1171-1176, 1997

[12] R. Motwani and P. Raghavan, "Randomized algorithms," Cambridge University Press, Cambridge, 1995

[13] E. Frazzoli, Z.-H. Mao, J.-H. Oh, and E. Feron, "Aircraft conflict resolution via semidefinite programming," To appear in *AIAA Journal of Guidance, Control and Dynamics.*

[14] P. Gahinet and P. Apkarian, "A linear matrix inequality approach to H_∞ control," *Int. J. Robust Nonlinear Control*, No. 4, pp.421-448, 1994

[15] M. Mesbahi, "On the semi-definite programming solution of the least order dynamic output feedback synthesis," *Proc. of the American Control Conference*, San Diego, CA, pp.2355-2359, 1999

[16] L. El Ghaoui and P. Gahinet, "Rank minimization under LMI constraints: a framework for output feedback problems," *Proc. of the European Cont. Conf.*, July 1993.

[17] T. Iwasaki and M. A. Rotea, "Fixed-order scaled H_∞ synthesis," *Optimal Control Applications and Methods*, vol. 18, pp. 381-398, 1997

[18] S. Poljak and Z. Tuza, "Maximum cuts and largest bipartite networks," In W. Cook, L. Lovász, and P. Seymour, editors, *Combinatorial Optimizations*, DIMACS series in Discrete Mathematics and Theoretical Computer Science, AMS, 1995

[19] M. X. Goemans and D. P. Williamson, "Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming," *J. of the Assoc. for Computing Machinery*, Vol. 42, No. 6, pp.1115-45, 1995

[20] K. Konishi and S. Shin, "Matrix-based SDP relaxation for QMI problems," Submitted to Trans. of SICE

[21] D. Karger, R. Motwani, and M Sudan, "Approximate graph coloring by semidefinite programming," *Proc. of the 35th Annual Sympo. on Foundations of Computer Science*, pg.2-13, 1994

[22] L. Vandenberghe, S. Boyd, and B. Alkire, *SP: Software for Semidefinite Programming*, Available at `http://www.stanford.edu/~boyd/`

[23] L. El Ghaoui, R. Nikoukhah, and F. Delebecque, "LMI-TOOL: A Package for LMI Optimization," *Proc. of the 34th Conf. on Decision and Control*, New Orleans, LA, U.S.A., pp. 3096-3101, 1995

[24] P. P. Khargonekar and M. A. Rotea, "Mixed H_2/H_∞ control: A convex optimization approach," *IEEE Trans. on Automatic Control*, Vol. 36, No. 7, pp.824-37, July 1991