A Sequential Parametric Convex Approximation Method for Solving Bilinear Matrix Inequalities

Donghwan Lee and Jianghai Hu

***Abstract*— The goal of this paper is to develop an algorithm for solving optimization problems subject to bilinear matrix inequalities (BMIs). We propose a sequential semidefinite programming (SDP) algorithm mainly motivated from the recently developed convex-concave programming approach, the path-following approach, and the general non-convex programming approaches. New convex over approximations of the BMI constraints are used. Finally, numerical experiments are provided for comparative analysis.**

I. Introduction

This paper is devoted to a class of bilinear matrix inequality (BMI) optimization problems, minimizing a convex objective function subject to BMI constraints. This problem arises in many engineering applications, for instance, the design of a reduced-order controller [1], the switched control design [2] in control theories, and the linear consensus protocol design [3]. Nowadays, there is an immense literature addressing the BMI optimization problems and related applications, such as the cone complementarity linearization method [4] for the reduce-order control design, the path-following method for general BMI problems [5], a concave programming approach [6], a Newton-like method for solving rank constrained linear matrix inequalities (LMIs) [7], general nonlinear programming methods [8], and [9].

Recently, the idea of DC (difference of two convex functions) programming [10] has been extended to the optimizations with convex-concave matrix inequality constraints in [11], [12]. The DC programming method is known as an effective technique for solving a class of non-convex optimizations, where the non-convex inequality constraints are expressed as a difference of two convex functions (or convex-concave decomposition) constraints. Then, the concave function is linearized to obtain a convex constraint whose feasible set is an inner convex approximation of the non-convex feasible set of the original inequality constraint.

In this paper, we propose an approach which does not exploit the DC decomposition and the linearization procedures. Instead, an elementary algebraic matrix inequality is used to obtain different approximations of the BMI constraints. The proposed method can be viewed as an extension of the path-following algorithm [5], where the bilinear term is replaced by a convex over approximation over the cone of positive semidefinite matrices. We also note that both the DC programming and the proposed method can be categorized into the inner approximation algorithm [13], the sequential parametric convex approximation method [14], and the sequential semidefinite programming (SDP) algorithm. Through numerical experiments, it is demonstrated that, in some cases, the proposed method outperforms the DC programming approach in [10] at the expense of increased computational efforts. The proposed method preserves several useful properties of the DC programming approach in [10]. One of them is that the proposed sequential SDP algorithm is relatively easy to implement by using standard SDP solvers and is intuitively easy to understand. Besides, at every iteration, a solution to the subproblem is guaranteed to be feasible, and no step size rule is required to ensure the feasibility.

*This material is based upon work supported by the National Science Foundation under Grant No. 1329875

D. Lee and J. Hu are with the Department of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47906, USA lee1923@purdue.edu, jianghai@purdue.edu.

II. Preliminaries

In this paper, we follow the notation used in [12]. Let $\mathbb{S}^p$ be the set of symmetric matrices of size $p \times p$, $\mathbb{S}^p_+$ and $\mathbb{S}^p_{++}$ be the set of symmetric positive semidefinite and positive definite matrices, respectively. For given matrices X and Y in $\mathbb{S}^p$, the relation $X \succeq Y$ (respectively, $X \preceq Y$) means $X - Y \in \mathbb{S}^p_+$ (respectively, $Y - X \in \mathbb{S}^p_+$) and $X \succ Y$ (respectively, $X \prec Y$) is $X - Y \in \mathbb{S}^p_{++}$ (respectively, $Y - X \in \mathbb{S}^p_{++}$). The quantity $X \circ Y := \mathrm{trace}(X^T Y)$ is an inner product of two matrices X and Y defined on $\mathbb{S}^p$, where $\mathrm{trace}(Z)$ is the trace of matrix Z. In addition, the following standard notation will be used: $\mathrm{He}\{A\} := A^T + A$; I_n and I: $n \times n$ identity matrix and identity matrix of appropriate dimensions; $||\cdot||$: Euclidean norm of a vector or spectral norm of a matrix; $\mathrm{s.t.}$: abbreviation of "subject to."

A function $f : \mathbb{R}^n \to \mathbb{R}$ is said to be strongly convex [15, chapter 9.1.2] with parameter $\rho > 0$ if $f(\cdot) - (1/2)\rho \left\|\cdot\right\|^2$ is convex. We define the derivative of a matrix-valued mapping $\mathcal{F}$ at $z_0 \in \mathbb{R}^n$ as a linear mapping from $\mathbb{R}^n$ to $\mathbb{R}^{p\times p}$ defined by

$$D_z\mathcal{F}[z]|_{z=z_0}[d] := \sum_{i=1}^n d_i \left.\frac{\partial \mathcal{F}[z]}{\partial z_i}\right|_{z=z_0}, \quad \forall d \in \mathbb{R}^n.$$

Let $\mathcal{A} : \mathbb{R}^n \to \mathbb{S}^p$ be a linear mapping defined as $\mathcal{A}[x] := \sum_{i=}^n x_i \mathcal{A}_i$, where $\mathcal{A}_i \in \mathbb{S}^p$ for $i \in \{1, \ldots, n\}$. The adjoint operator of $\mathcal{A}$, $\mathcal{A}^*$, is defined as $\mathcal{A}^* Z := [\mathcal{A}_1 \circ Z, \mathcal{A}_2 \circ Z, \ldots, \mathcal{A}_n \circ Z]^T$ for any $Z \in \mathbb{S}^p$. The concept of the convexity for the matrix-valued mapping is defined as follows.

Definition 1 ([16]): The matrix-valued mapping $\mathcal{A} : \mathbb{R}^n \to \mathbb{S}^p$ is said to be positive semidefinite convex (psd-convex) on a convex subset $C \subseteq \mathbb{R}^n$ if for all $t \in [0, 1]$, and $x, y \in C$, one has $\mathcal{A}[tx+(1-t)y] \preceq t\mathcal{A}[x]+(1-t)\mathcal{A}[y]$. The matrix-valued mapping $\mathcal{A}$ is said to be positive semidefinite concave (psd-concave) on $C \subseteq \mathbb{R}^n$ if $-\mathcal{A}$ is psd-convex.

Consider the matrix-valued mapping $\mathcal{F} : \mathbb{R}^n \to \mathbb{S}^p$

$$\mathcal{F}(z) = C + \mathcal{L}[z] + \text{He}\{\mathcal{A}[x]\mathcal{B}[y]\}, \tag{1}$$

where $z = [x^T, y^T]^T \in \mathbb{R}^n$, $x \in \mathbb{R}^{n_x}$, $y \in \mathbb{R}^{n_y}$ are variables, $\mathcal{L} : \mathbb{R}^n \to \mathbb{S}^p$, $\mathcal{A} : \mathbb{R}^{n_x} \to \mathbb{R}^{p\times q}$, and $\mathcal{B} : \mathbb{R}^{n_y} \to \mathbb{R}^{q\times p}$ are linear mappings. In this paper, we consider a class of optimization problems with a BMI constraint (BMIP) of the form

$$\min_z \quad f(z) \quad \text{s.t.} \quad \mathcal{F}[z] \preceq 0, \quad z \in \Omega, \tag{2}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is convex, and $\Omega \subset \mathbb{R}^n$ is a nonempty, closed, and convex set. The algorithm proposed in this paper can be directly extended to the optimization problems with multiple BMI constraints. Thus, the general case will not be considered here for notational simplicity and to save space.

To proceed, denote by $\mathcal{D} := \{z \in \Omega : \mathcal{F}[z] \preceq 0\}$ the feasible set of (2), and define $\mathcal{D}_0 := \{z \in \text{ri}(\Omega) : \mathcal{F}[z] \prec 0\}$, where $\text{ri}(\Omega)$ is the set of relative interior points of the convex set Ω [15, chapter 2.1.3]. Throughout the paper, the following assumptions apply.

Assumption 1: $\mathcal{D}_0$ is nonempty.

Assumption 2: f is bounded from below on $\mathcal{D}$, and is differentiable.

III. A Sequential Parametric Convex Approximation Method

In this section, we propose an algorithm for solving the BMIP (2). The proposed method can be also viewed as an extension of the sequential parametric convex approximation method [14] to the BMIP (2).

To explain the proposed approach, we reformulate the matrix $\mathcal{F}[z]$ in (2) by

$$\begin{aligned}
&\mathcal{F}[z] \\
&= C + \mathcal{L}[z] + \text{He}\{\mathcal{A}[x]\mathcal{B}[y]\} \\
&= C + \mathcal{L}[z - z_k + z_k] \\
&+ \text{He}\{\mathcal{A}[x - x_k + x_k]\mathcal{B}[y - y_k + y_k]\} \\
&= \mathcal{F}[z_k] + D_z\mathcal{F}[z]|_{z=z_k}[z - z_k] \\
&+ \text{He}\{\mathcal{A}[x - x_k]\mathcal{B}[y - y_k]\},
\end{aligned} \tag{3}$$

where $z_k := [x_k^T, y_k^T]^T$, and

$$\begin{aligned}
&D_z\mathcal{F}[z]|_{z=z_k}[z - z_k] \\
&\quad = \mathcal{L}[z - z_k] + \text{He}\{\mathcal{A}[x_k]\mathcal{B}[y - y_k]\} \\
&\quad + \text{He}\{\mathcal{A}[x - x_k]\mathcal{B}[y_k]\}.
\end{aligned}$$

If the last term in (3) is ignored, then the linearization $\mathcal{F}[z_k] + D_z\mathcal{F}[z]|_{z=z_k}[z - z_k]$ of $\mathcal{F}[z]$ around z_k is obtained. The path-following algorithm in [5] for the BMIP (2) is to solve (2) with the BMI constraint replaced with the linearized constraint. Instead of dropping the bilinear term $\text{He}\{\mathcal{A}[x - x_k]\mathcal{B}[y - y_k]\}$ in (3), we can obtain an over estimation of $\mathcal{F}[z]$ over the cone of positive semidefinite matrices by using a matrix inequality, which is often used in the control theory literature, e.g., [17, Proposition 2.1 and Proposition 2.2].

Lemma 1: Let A and E be real matrices of appropriate dimensions. Then, for any $S \in \mathbb{S}^n_{++}$,

$$\text{He}\{DE\} \preceq DSD^T + E^T S^{-1} E.$$

Proof: For any $S \in \mathbb{S}^n_{++}$, the inequality is obtained by expanding $(D^T - S^{-1}E)^T S (D^T - S^{-1}E) \succeq 0$. ■

Using Lemma 1, an upper bound on $\text{He}\{\mathcal{A}[x - x_k]\mathcal{B}[y - y_k]\}$ in (3) is obtained as

$$\begin{aligned}
&\text{He}\{\mathcal{A}[x - x_k]\mathcal{B}[y - y_k]\} \\
&\quad \preceq \mathcal{A}[x - x_k] S \mathcal{A}[x - x_k]^T \\
&\quad + \mathcal{B}[y - y_k]^T S^{-1} \mathcal{B}[y - y_k],
\end{aligned}$$

where $S \in \mathbb{S}^n_{++}$. Thus, for any given $z_k \in \Omega$, an over estimation of $\mathcal{F}[z]$ is given by

$$\begin{aligned}
\mathcal{F}[z] &\preceq \mathcal{F}[z_k] + D_z\mathcal{F}[z]|_{z=z_k}[z - z_k] \\
&+ \mathcal{A}[x - x_k] S \mathcal{A}[x - x_k]^T \\
&+ \mathcal{B}[y - y_k]^T S^{-1} \mathcal{B}[y - y_k] \\
&=: \mathcal{K}[z;\, z_k,\, S].
\end{aligned} \tag{4}$$

The mapping $\mathcal{K}[z;\, z_k,\, S]$ in (4) has the following properties.

Proposition 1: For any given $z_k \in \Omega$, $S \in \mathbb{S}^n_{++}$, the matrix-valued mapping $\mathcal{K}[\cdot;\, z_k,\, S]$ in (4) satisfies the following properties:

1) $\mathcal{F}[z] \preceq \mathcal{K}[z;\, z_k,\, S], \quad \forall z \in \mathbb{R}^n$;
2) $\mathcal{F}[z_k] = \mathcal{K}[z_k;\, z_k,\, S]$;
3) $D_z\mathcal{K}[z;\, z_k,\, S]|_{z=z_k}[u] = D_z\mathcal{F}[z]|_{z=z_k}[u], \quad \forall u \in \mathbb{R}^n$.

Proof: The statement 1) was proved in (4), and the statement 2) can be proved by setting $z = z_k$ in (4). To prove 3), note that $D_z\mathcal{F}[z]|_{z=z_k}[u]$ is given by

$$\begin{aligned}
D_z\mathcal{F}[z]|_{z=z_k}[u] &= \mathcal{L}[u] + \text{He}\{\mathcal{A}[x_k]\mathcal{B}[w]\} \\
&+ \text{He}\{\mathcal{A}[v]\mathcal{B}[y_k]\}.
\end{aligned}$$

where $v \in \mathbb{R}^{n_x}$, $w \in \mathbb{R}^{n_y}$ are appropriately partitioned vectors such that $u = [v,\, w]^T$. In addition, $D_\xi\mathcal{K}(\xi;\, z_k,\, S)|_{\xi=z}[u]$ is obtained as

$$\begin{aligned}
&D_\xi\mathcal{K}(\xi;\, z_k,\, S)|_{\xi=z}[u] \\
&\quad = D_\xi\mathcal{F}(\xi)|_{\xi=z}[u] \\
&\quad + \mathcal{A}[x - x_k] S \mathcal{A}[v]^T + \mathcal{A}[v] S \mathcal{A}[x - x_k]^T \\
&\quad + \mathcal{B}[w]^T S^{-1} \mathcal{B}[y - y_k] + \mathcal{B}[y - y_k]^T S^{-1} \mathcal{B}[w].
\end{aligned}$$

Plugging $z = z_k = [x_k^T, y_k^T]^T$ into the above equality yields $D_\xi\mathcal{K}(\xi;\, z_k,\, S)|_{\xi=z_k}[u] = D_\xi F(z)|_{\xi=z_k}[u]$, concluding the proof. ■

It is worth mentioning that the above properties of the approximation $\mathcal{K}[z;\, z_k,\, S]$ are equivalent to that given in [13, page 682]. Therefore, the set of $z \in \Omega$ such that $\mathcal{K}[z;\, z_k,\, S] \preceq 0$ is an inner approximation of the feasible set of (2) around z_k. Instead of solving (2), we can solve the following approximate problem for a fixed $S \in \mathbb{S}^n_{++}$:

$$\begin{cases} \min\limits_z \; f(z) \quad \text{s.t.} \\ \mathcal{K}[z;\, z_k,\, S] \preceq 0, \quad z \in \Omega \end{cases} \tag{5}$$

It is easy to prove that $\mathcal{K}[z;\, z_k,\, S]$ is psd-convex in $z \in \mathbb{R}^n$. Therefore, the problem (5) is a convex optimization. The path-following approach in [5] replaces the whole BMI constraints with their linearizations, and the DC programming approach in [12] replaces only the concave terms in the BMIs with their linearizations while preserving the affine and convex terms. In the proposed approach, we replace the bilinear terms with their convex quadratic over approximations while preserving the affine terms. If an optimal solution z^* to the above problem is close to z_k, then the constraint (5) approximates the constraint of (2) because of the statement (2) of Proposition 1. Using the Schur complement, (5) is converted to the equivalent form

$$\min_{z\in\Omega} f(z) \quad \text{s.t.}$$
$$\begin{bmatrix} \mathcal{F}[z_k] + D_z\mathcal{F}[z]|_{z=z_k}[z-z_k] & * & * \\ \mathcal{A}[z-z_k]^T & -S^{-1} & * \\ \mathcal{B}[z-z_k] & 0 & -S \end{bmatrix} \preceq 0. \quad (6)$$

The inequality (6) is an LMI constraint, and the above convex optimization can be solved using standard convex optimization techniques [15]. If the set Ω consists of LMI constraints, then it can be solved by using SDP solvers. By repeatedly solving the problem and using the current optimal value z^* for the next point z_{k+1}, we can obtain an iterative convex optimization algorithm for the BMIP.

Although it is difficult to compare the tightness of the proposed approximation with that of the DC programming, the quality of the approximation can be improved since the gap between the over approximation and the original bilinear term in Lemma 1 can be reduced by an appropriate choice of S. However, the determination of S in (6) is a nonconvex problem due to the inverse of S in (6). One possible way to determine $S \succ 0$ through the convex program is to linearize the inverse $-S^{-1}$. A key observation is that $f(S) = S^{-1}$ is psd-convex on $\mathbb{S}^n_{++}$ by [12, Lemma 3.1], and hence, $g(S) = -S^{-1}$ is psd-concave. By linearizing $-S^{-1}$ around S_k, we have the over approximation $-S^{-1} \preceq -2S_k^{-1} + S_k^{-1}SS_k^{-1}$ around S_k. For completeness and easy reference, it is formally stated in the following lemma.

Lemma 2: Suppose that $\mathcal{S} : \mathbb{R}^n \to \mathbb{S}^p$ is a linear mapping defined as $\mathcal{S}[x] = \sum_{i=1}^n x_i\mathcal{S}_i$, where $\mathcal{S}_i \in \mathbb{S}^p$ for $i \in \{1, \ldots, n\}$. If $\mathcal{S}[x] \succ 0$ and $\mathcal{S}[y] \succ 0$, then $-\mathcal{S}[y]^{-1} \preceq -2\mathcal{S}[x]^{-1} + \mathcal{S}[x]^{-1}\mathcal{S}[y]\mathcal{S}[x]^{-1}$ holds.

Proof: From the formulation of the derivative of the matrix inverse [18], we have

$$\begin{aligned} D_z(-\mathcal{S}[z]^{-1})\Big|_{z=x}[d] &= -\sum_{i=1}^n d_i \frac{\partial(\mathcal{S}[z]^{-1})}{\partial z_i}\Bigg|_{z=x} \\ &= -\sum_{i=1}^n d_i \left\{ -\mathcal{S}[z]^{-1}\frac{\partial\mathcal{S}[z]}{\partial z_i}\mathcal{S}[z]^{-1} \right\}\Bigg|_{z=x} \\ &= \mathcal{S}[x]^{-1}(D_z\mathcal{S}[z]|_{z=x}[d])\mathcal{S}[x]^{-1}. \end{aligned}$$

Using this result, if $\mathcal{S}[x]$ and $\mathcal{S}[y]$ are invertible, then the linearization of $-\mathcal{S}[y]^{-1}$ around x is

$$-\mathcal{S}[x]^{-1} + \mathcal{S}[x]^{-1}(D_z\mathcal{S}[z]|_{z=x}[y-x])\mathcal{S}[x]^{-1}.$$

Since the mapping $g(S) = -S^{-1}$ is psd-concave on $\mathbb{S}^n_{++}$, for $\mathcal{S}[x] \succ 0$ and $\mathcal{S}[y] \succ 0$, we have

$$\begin{aligned} -\mathcal{S}[y]^{-1} &\preceq -\mathcal{S}[x]^{-1} + \mathcal{S}[x]^{-1}(D_z\mathcal{S}[z]|_{z=x}[y-x])\mathcal{S}[x]^{-1} \\ &= -2\mathcal{S}[x]^{-1} + \mathcal{S}[x]^{-1}\mathcal{S}[y]\mathcal{S}[x]^{-1}. \end{aligned}$$

This completes the proof. ■

By replacing $-S^{-1}$ in (6) with $-2S_k^{-1} + S_k^{-1}SS_k^{-1}$ and multiplying (6) from the left and right by the block diagonal matrix $\mathrm{diag}(I,\, S_k,\, I)$, we obtain the following convex optimization:

$$\min_{z\in\Omega,\, S\in\mathbb{S}^n} f(z) \quad \text{s.t.} \quad (7)$$
$$\begin{bmatrix} \mathcal{F}[z_k] + D_z\mathcal{F}[z]|_{z=z_k}[z-z_k] & * & * \\ S_k\mathcal{A}[z-z_k]^T & -2S_k + S & * \\ \mathcal{B}[z-z_k] & 0 & -S \end{bmatrix} \preceq 0,$$
$$S \succ 0, \quad 2S_k - S \succ 0.$$

By sequentially solving the above convex program and using the current optimal point for the next point z_{k+1} and S_{k+1}, Algorithm 1 shown at the top of the next page is obtained.

Algorithm 1 A sequential parametric convex approximation algorithm for BMIP

1: Initialize $z_0 \in \mathcal{D}_0$ and set $k \leftarrow 0$, $S_k = I_n$;
2: **repeat**
3: Solve

$$(z_{k+1},\, S_{k+1}) = \operatorname*{arg\,min}_{z\in\mathbb{R}^n,\, S\in S^n} f_\rho(z;\, z_k) \quad \text{s.t.} \quad (8)$$
$$\begin{bmatrix} \mathcal{F}[z_k] + D_z\mathcal{F}[z]|_{z=z_k}[z-z_k] & * & * \\ S_k\mathcal{A}[z-z_k]^T & -2S_k + S & * \\ \mathcal{B}[z-z_k] & 0 & -S \end{bmatrix} \preceq 0,$$
$$c_1 I \preceq S \preceq c_2 I, \quad -2S_k + S \preceq -c_3 I \quad z \in \Omega, \quad (9)$$

where $\rho > 0$, $c_2 > c_1 > 0$, $c_3 > 0$, $f_\rho(z;\, z_k) := f(z) + \frac{\rho}{2}\|z - z_k\|^2$.
4: $k \leftarrow k+1$;
5: **until** a certain stopping criterion is satisfied.

Remark 1: Due to some technical reasons, the optimization (7) is modified in the subproblem of Algorithm 1. In particular, the constraint $c_1 I \preceq S \preceq c_2 I$ in (9) ensures that each S_k is nonsingular and the sequence $\{S_k\}_{k\geq 0}$ is bounded. Moreover, $-2S_k + S \preceq -c_3 I$ is included in the algorithm to guarantee that $-2S_k + S_{k+1}$ is nonsingular for all $k \geq 0$. The term $\frac{\rho}{2}\|z - z_k\|^2$ in the objective function in (8) is a regularization term to guarantee that the value of the function f is strictly descent at each iteration.

Similarly to the DC programming in [12], a favorable feature of Algorithm 1 is that the optimal solution of the subproblem at each iteration is a feasible point of the original problem (2).

Proposition 2: Let $\{(z_k,\, S_k)\}_{k\geq 0}$ be a sequence of optimal solutions generated by Algorithm 1. For every $k \geq 0$, z_k is a feasible solution to (2), i.e., $z_k \in \Omega$, $\mathcal{F}[z_k] \preceq 0$.

Proof: The proof will be completed by the induction argument. From Algorithm 1, the feasibility of the initial point is guaranteed, i.e., $z_0 \in \Omega$, $\mathcal{F}[z_0] \preceq 0$. Suppose that z_k is a feasible point of (2). Then, it is guaranteed that the subproblem in Algorithm 1 is always feasible for any $S_k \in \mathbb{S}^n_{++}$ because of the trivial feasible point $(z, S) = (z_k, S_k)$. Let (z_{k+1}, S_{k+1}) be an optimal solution to the subproblem in Algorithm 1. By plugging $(z, S) = (z_{k+1}, S_{k+1})$ into the constraint of (8), and applying the Schur complement, we get

$$\begin{bmatrix} \begin{pmatrix} \mathcal{F}[z_k] + D_z\mathcal{F}[z]|_{z=z_k}[z_{k+1} - z_k] \\ +\mathcal{B}[z_{k+1} - z_k]^T S_{k+1}^{-1}\mathcal{B}[z_{k+1} - z_k] \end{pmatrix} & * \\ S_k\mathcal{A}[z_{k+1} - z_k]^T & -2S_k + S_{k+1} \end{bmatrix}$$
$$\preceq 0.$$

Multiplying the above matrix by $[I, \mathcal{A}[z_{k+1} - z_k]]$ from the left and by its transpose from the right, we have

$$\begin{aligned} &\mathcal{F}[z_k] + D_z\mathcal{F}[z]|_{z=z_k}[z_{k+1} - z_k] \\ &\quad + \mathcal{A}[z_{k+1} - z_k]^T S_{k+1}\mathcal{A}[z_{k+1} - z_k]^T \\ &\quad + \mathcal{B}[z_{k+1} - z_k]^T S_{k+1}^{-1}\mathcal{B}[z_{k+1} - z_k] \\ &\quad = \mathcal{K}[z_{k+1}; z_k, S_{k+1}] \\ &\quad \preceq 0. \end{aligned}$$

By the statement (1) of Proposition 1, this implies that $z_{k+1} \in \Omega$ satisfies $\mathcal{F}[z_{k+1}] \preceq 0$. Therefore, z_{k+1} is a feasible point of (2). This completes the proof. ■

IV. Example: Application to Control Design

All numerical examples were solved by MATLAB R2008a running on a Windows 10 PC with Intel Core i5-4210 2.6G Hz CPU, 4 GB RAM. The convex optimization problems were solved with SeDuMi [19] and Yalmip [20].

In this section, the static output feedback (SOF) controller design problem [1] will be considered to illustrate the proposed method. The spectral abscissa optimization problem for the SOF controller design, addressed also in [12, page 1383], is a well-known BMIP of the form (2) in the systems and control literature, e.g., [4], [5], [19]. Consider the continuous-time linear time-invariant system $\dot{x}(t) = Ax(t) + Bu(t)$, $y(t) = Cx(t)$, $x(0) = \mathbb{R}^{n_x}$, where $(A, B, C) \in \mathbb{R}^{n_x \times n_x} \times \mathbb{R}^{n_x \times n_u} \times \mathbb{R}^{n_y \times n_x}$, $t \geq 0$, $x(t) \in \mathbb{R}^{n_x}$ is the state, $u(t) \in \mathbb{R}^{n_u}$ is the control input, and $y(t) \in \mathbb{R}^{n_y}$ is the measured output. The goal is to find the gain matrix $F \in \mathbb{R}^{n_u \times n_y}$ so that the system can be stabilized by the SOF controller $u(t) = Fy(t)$. Specifically, the design problem can be formulated as the non-convex optimization problem

$$\inf_{F \in \mathbb{R}^{n_u \times n_y}} f(F),$$

where $f(F) := \max\{\mathrm{Re}\{\lambda\} : \lambda \in \Lambda(A + BFC)\}$, $\mathrm{Re}\{\lambda\}$ denotes the real part of $\lambda \in \mathbb{C}$, and $\Lambda(A + BFC)$ is the spectrum of $A + BFC$. The problem can be expressed as the BMIP (2)

$$\inf_{P,\,F,\,\alpha} \alpha \quad \text{s.t.} \tag{10}$$
$$(A + BFC)^T P + P(A + BFC) - 2\alpha P \preceq 0,$$
$$(P, F, \alpha) \in \Omega,$$

where

$$\begin{aligned} \Omega = \{&(P, F, \alpha) \in \mathbb{R}^{n_x \times n_x} \times \mathbb{R}^{n_x \times n_u} \times \mathbb{R} : \\ &P = P^T, P - 10^{-6} I \succeq 0\}. \end{aligned} \tag{11}$$

To apply Algorithm 1, we replace (F, P, α) in (10) with $(F_k + \Delta F, P_k + \Delta P, \alpha_k + \Delta\alpha)$ to have

$$\begin{aligned} &\mathrm{He}\{(P_k + \Delta P)(A + B(F_k + \Delta F)C)\} \\ &\quad - 2(\alpha_k + \Delta\alpha)(P_k + \Delta P) \preceq 0, \end{aligned}$$

where $\Delta F = F - F_k$, $\Delta P = P - P_k$, and $\Delta\alpha = \alpha - \alpha_k$. By expanding and rearranging the terms in the last matrix inequality, we obtain

$$\begin{aligned} &\mathrm{He}\{PA + PBF_kC + P_kB\Delta FC\} \\ &\quad - 2(\alpha P_k + \alpha_k\Delta P) \\ &\quad + \mathrm{He}\{\Delta P(B\Delta FC - \Delta\alpha I_n)\} \preceq 0. \end{aligned} \tag{12}$$

Noting that the last term in the left-hand side of the above inequality is bilinear, the LMI constraint (9) is obtained as (13) at the top of page 5. Then, the subproblem (8) of Algorithm 1 is obtained as follows:

$$\begin{aligned} &(P_{k+1}, F_{k+1}, \alpha_{k+1}, S_{k+1}) \\ &:= \operatorname*{arg\,inf}_{(P,F,\alpha)\in\Omega,\, S\in\mathbb{S}^n} \Big\{\alpha + 0.005\,\|\Delta P\|_F^2 \\ &\qquad\qquad + 0.005\,\|\Delta F\|_F^2 + 0.005\,\|\Delta\alpha\|_F^2\Big\} \\ &\text{s.t.}\quad \text{LMI (13) holds, and} \\ &10^{-6} I \preceq S \preceq 10^4 I, \quad -2S_k + S \preceq -10^{-6} I, \end{aligned}$$

where $\|\cdot\|_F$ is the Frobenius norm.

There are several remarks on the implementation of Algorithm 1.

1) We use the system data extracted from the publicly available database COMPleib library [21];
2) Since comprehensive comparison results among the DC programming approach and other BMI solvers, such as HIFOO [8], PENBMI [9], and LMIRank [7], have been reported in [12], we focus only on the comparison between the proposed method and the DC programming approach in [12];
3) We use the same method of [12] to determine an initial feasible point of (10), i.e., $F_0 = 0$, $\alpha_0 = -0.5f(0)$, and P_0 is chosen as a solution to the LMIs $P_0 \succeq 10^{-6}I$, $A^TP_0 + P_0A + 2\alpha_0P_0 \preceq 0$;
4) The maximum number of iterations is set to 50. However, no stopping criterion is used.
5) Ω is convex and closed;

The comparison of the values of $\alpha_0(A_F)$ computed using the proposed method and the DC programming are shown in Figs. 1-6, using the system data AC1, AC2, AC7, AC11, HE1, DIS1, DIS4, NN1, NN13, and REA1 of the COMPleib library [21]. From the experiments, one can observe that the proposed method is computationally more demanding but

$$\begin{bmatrix} \mathrm{He}\{PA + PBF_kC + P_kB\Delta FC\} - 2(\alpha P_k + \alpha_k \Delta P) & * & * \\ S_k \Delta P & S - 2\mathrm{S}_k & * \\ B\Delta FC - \Delta\alpha I_n & 0 & -S \end{bmatrix} \preceq 0, \quad (13)$$

shows faster convergence for AC1, AC2, AC11, HE1, DIS4, REA1, and NN1. However, for NN13, DIS1, AC7, the DC programming gives similar or better results.

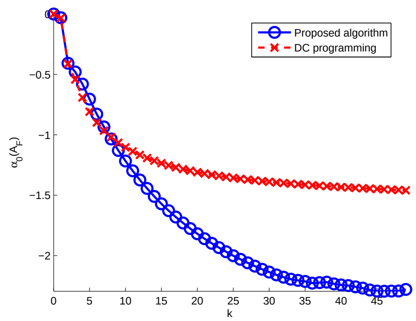

Fig. 1. Comparison using AC1. Computational time in seconds: 25.5 for the proposed method and 16.5 for the DC programming.

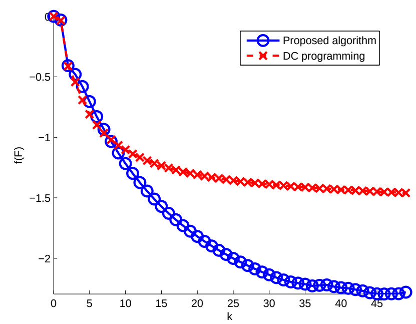

Fig. 2. Comparison using the AC2. Computational time in seconds: 35.2 for the proposed method and 23.6 for the DC programming.

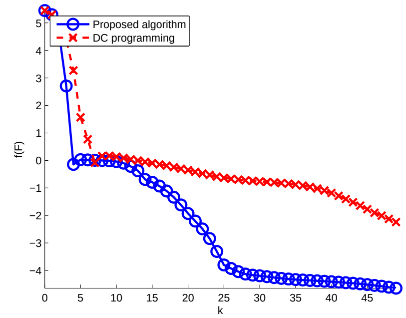

Fig. 3. Comparison using the AC11. Computational time in seconds: 35.3 for the proposed method and 25.1 for the DC programming.

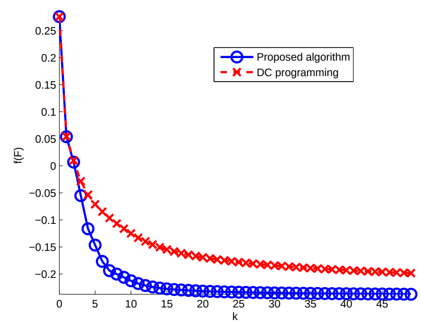

Fig. 4. Comparison using the HE1. Computational time in seconds: 33.4 for the proposed method and 18.1 for the DC programming.

V. Conclusion

In this paper, we have proposed a sequential parametric convex approximation method for solving optimization problems subject to BMIs. Numerical experiments have suggested that, for some examples, the proposed method shows faster convergence compared to the DC programming approach at the expense of higher computational efforts. It is expected that both approaches can complement each other.

References

[1] V. L. Syrmos, C. T. Abdallah, P. Dorato, and K. Grigoriadis, "Static output feedback–a survey," *Automatica*, vol. 33, no. 2, pp. 125–137, 1997.

[2] T. Hu, L. Ma, and Z. Lin, "Stabilization of switched systems via composite quadratic functions," *IEEE Trans. Autom. Control*, vol. 53, no. 11, pp. 2571–2585, 2008.

[3] S. You, "A fast linear consensus protocol on an asymmetric directed graph," in *Proc. Amer. Control Conf.*, 2014, pp. 3281–3286.

[4] L. El Ghaoui, F. Oustry, and M. AitRami, "A cone complementarity linearization algorithm for static output-feedback and related problems," *IEEE Trans. Autom. Control*, vol. 42, no. 8, pp. 1171–1176, 1997.

[5] A. Hassibi, J. How, and S. Boyd, "A path-following method for solving BMI problems in control," in *Proc. Amer. Control Conf.* IEEE, 1999, pp. 1385 – 1389.

[6] P. Apkarian and H. D. Tuan, "Robust control via concave minimization local and global algorithms," *IEEE Trans. Autom. Control*, vol. 45, no. 2, pp. 299–305, 2000.

[7] R. Orsi, U. Helmke, and J. B. Moore, "A Newton-like method for solving rank constrained linear matrix inequalities," *Automatica*, vol. 42, no. 11, pp. 1875–1882, 2006.

[8] S. Gumussoy, D. Henrion, M. Millstone, and M. L. Overton, "Multiobjective robust control with HIFOO 2.0," *arXiv preprint arXiv:0905.3229*, 2009.

[9] D. Henrion, J. Löfberg, M. Kočvara, and M. Stingl, "Solving polynomial static output feedback problems with PENBMI," in *Proc. 44th IEEE Conf. Decision Control, and Eur. Control Conf.*, 2005, pp. 7581–7586.

[10] P. D. Tao and L. T. H. An, "A DC optimization algorithm for solving the trust-region subproblem," *SIAM J. Optim.*, vol. 8, no. 2, pp. 476–505, 1998.

[11] Q. T. Dinh, W. Michiels, S. Gros, and M. Diehl, "An inner convex approximation algorithm for bmi optimization and applications in control," in *Proc. 51st IEEE Conf. Decision Control*, 2012, pp. 3576–3581.

[12] Q. T. Dinh, S. Gumussoy, W. Michiels, and M. Diehl, "Combining convex–concave decompositions and linearization approaches for solv-

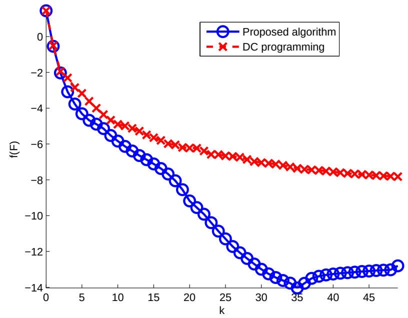

Fig. 5. Comparison using the DIS4. Computational time in seconds: 33.4 for the proposed method and 24.1 for the DC programming.

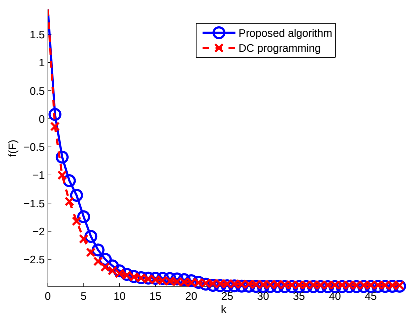

Fig. 6. Comparison using the NN13. Computational time in seconds: 30.2 for the proposed method and 19.3 for the DC programming.

ing BMIs, with application to static output feedback," *IEEE Trans. Autom. Control*, vol. 57, no. 6, pp. 1377–1390, 2012.

[13] B. R. Marks and G. P. Wright, "A general inner approximation algorithm for nonconvex mathematical programs," *Oper. Res.*, vol. 26, no. 4, pp. 681–683, 1978.

[14] A. Beck, A. Ben-Tal, and L. Tetruashvili, "A sequential parametric convex approximation method with applications to nonconvex truss topology design problems," *J. Global Optim.*, vol. 47, no. 1, pp. 29–51, 2010.

[15] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[16] A. Shapiro, "First and second order analysis of nonlinear semidefinite programs," *Math. Program.*, vol. 77, no. 1, pp. 301–320, 1997.

[17] X. Li and C. E. De Souza, "Criteria for robust stability and stabilization of uncertain linear systems with state delay," *Automatica*, vol. 33, no. 9, pp. 1657–1662, 1997.

[18] K. B. Petersen, M. S. Pedersen *et al.*, *The Matrix Cookbook*. Technical University of Denmark, 2008.

[19] J. F. Sturm, "Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones," *Optim. Methods Softw.*, vol. 11, no. 1-4, pp. 625–653, 1999.

[20] J. Löfberg, "YALMIP: A toolbox for modeling and optimization in MATLAB," in *Proc. CACSD Conf.*, 2004, pp. 284–289.

[21] F. Leibfritz and W. Lipinski, "Description of the benchmark examples in COMPleib 1.0," *Dept. Math., Univ. Trier, Trier, Germany, Tech. Rep*, 2003.

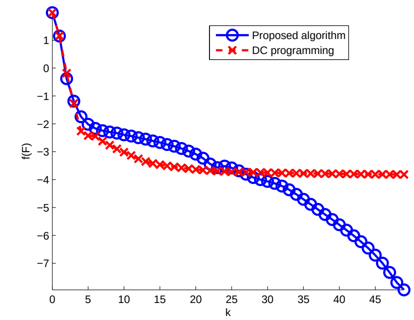

Fig. 7. Comparison using the REA1. Computational time in seconds: 20.9 for the proposed method and 14.9 for the DC programming.

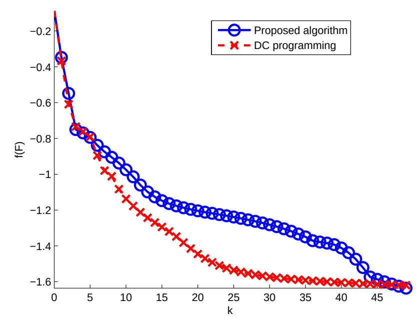

Fig. 8. Comparison using the DIS1. Computational time in seconds: 36.1 for the proposed method and 22.4 for the DC programming.

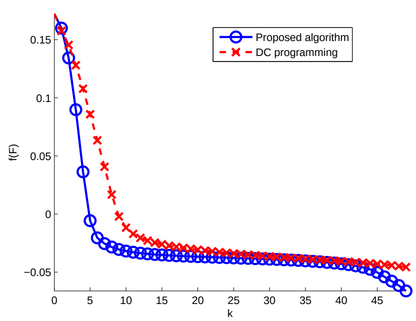

Fig. 9. Comparison using the AC7. Computational time in seconds: 29.1 for the proposed method and 17.1 for the DC programming.

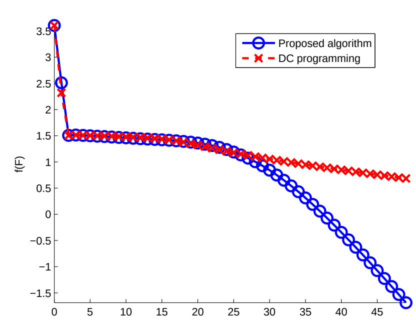

Fig. 10. Comparison using the NN1. Computational time in seconds: 16.3 for the proposed method and 13.2 for the DC programming.