A GLOBAL BMI ALGORITHM BASED ON THE GENERALIZED BENDERS DECOMPOSITION

Eric Beran [†*] **Lieven Vandenberghe**[‡] **Stephen Boyd**[‡]

[†] Department of Automation, Technical University of Denmark, DK-2800 Lyngby
Email: `E.Beran@mat.dtu.dk`, WWW: `http://www.iau.dtu.dk/~ebb`, FAX: +45 45 88 12 95

[‡] Durand Building, Information Systems Lab.,
Department of Electrical Engineering, Stanford, CA 94305-9510
Email: `vandenbe/boyd@isl.stanford.edu`

Appeared in the Proceedings of the
European Control Conference, Brussels, Belgium, 1997.
Paper no. 934.

Keywords: Linear matrix inequalities, Robust control, Numerical Methods

Abstract

We present a new algorithm for the global solution of optimization problems involving bilinear matrix inequalities (BMIs). The method is based on a technique known in large-scale and global optimization as the generalized Benders decomposition. It extends the GOP algorithm of Visweswaran and Floudas for bilinear and biconvex programming to problems with BMI constraints.

1 Problem statement

A *bilinear matrix inequality* (BMI) is an inequality of the form

$$F(x,y) \triangleq F_0 + \sum_{i=1}^{m_x} x_i F_i + \sum_{j=1}^{m_y} y_j G_j + \sum_{j=1}^{m_y}\sum_{i=1}^{m_x} y_j x_i H_{ij} \geq 0. \qquad (1)$$

The variables are $x \in \mathbf{R}^{m_x}$ and $y \in \mathbf{R}^{m_y}$. The matrices $F_i = F_i^T \in \mathbf{R}^{n\times n}, i = 1,\ldots,m_x$, $G_j = G_j^T \in \mathbf{R}^{n\times n}, j = 1,\ldots,m_y$, and $H_{ij} = H_{ij}^T \in \mathbf{R}^{n\times n}, i = 1,\ldots,m_x, j = 1,\ldots,m_y$, are given data, and the inequality sign '$\geq$' denotes positive semidefiniteness. For fixed y the BMI (1) reduces to a linear matrix inequality (LMI) in the variable x; for fixed x it reduces to an LMI in the variable y. BMIs were introduced by Safonov, Goh, Mesbahi, and others as a unified description of a wide variety of control problems [GLTS94, SGL94, MPS95].

We will consider the following *BMI optimization problem*:

$$\begin{array}{ll} \text{minimize} & f(x,y) = c^T x + d^T y \\ \text{subject to} & F(x,y) \geq 0 \\ & A(y) = A_0 + \sum_{j=1}^{m_y} y_j A_j \geq 0 \\ & -l_i \leq x_i \leq u_i,\ i = 1,\ldots,m_x, \end{array} \qquad (2)$$

where the variables are x and y. $A_i = A_i^T \in \mathbf{R}^{p\times p}$, $i = 1,\ldots,m_y$. The set $\{y \mid A(y) \geq 0\} \equiv \mathcal{Y}$ is assumed to be bounded.

Note that it has been shown that the BMI problem is NP-hard, see [TO95]. Existing BMI methods are either local methods that alternate between optimization over y and over x (see also [EB94]), or global (branch-and-bound type) methods [GSP94, GSL95]. In the branch and bound method, lower bounds can be found by relaxing the BMI to an LMI by substituting the bilinear terms $x_i y_j$ with new variables w_{ij}, and adding bounds on w_{ij}. This method has two drawbacks. First the LMI to be solved for the lower bound grows exponentially in size with the number of variables, and second the relaxation is only tight for very small rectangular bounds. For some specific classes of BMI problems, for example the low-order controller design problem, heuristic specialized methods with local convergence have been developed [BG96].

The method presented in this paper uses a technique called generalized Benders decomposition [Ben62, Geo72]. It can be interpreted as an extension of the GOP algorithm of Visweswaran and Floudas[FV93, VFIP96], which

*Research sponsored by the Danish Research Council under grant no. 26-1830 and 95-00765.

is a global optimization method for bilinear and biconvex optimization problems.

The outline of the paper is as follows. In the next section we present the duality theory, which provides computable upper and lower bounds. for the BMI optimization problem. In the following section the generalized Bender's decomposition is presented. In section 4 we show the use and effectiveness of the algorithm on two simple control problems. We conclude in section 5.

2 Duality theory

In this section we show how to compute upper and lower bounds for (2), based on Lagrange duality. We define the set

$$\mathcal{R}^x = \{x \mid -l_i \le x_i \le u_i,\ i=1,\dots,m_x\} \subset \mathbf{R}^{m_x}. \quad (3)$$

Since $\mathcal{Y}$ and $\mathcal{R}^x$ are bounded and closed, and $f(x,y)$ is linear it follows that there exists an optimum. We will denote this by f^*.

We will assume that all $y \in \mathcal{Y}$ are feasible. By introducing a slack variable $t \ge 0$, we can transform the possibly infeasible problem (2) to

$$\begin{array}{ll} \text{minimize} & c^T x + d^T y + \rho t \\ \text{subject to} & F(x,y) + tI \ge 0, \\ & A(y) \ge 0 \\ & -l_i \le x_i \le u_i,\ i=1,\dots,m_x, \\ & 0 \le t \le M, \end{array} \quad (4)$$

where ρ, M are fixed. M should be chosen such that for all $y \in \mathcal{Y}$ it is possible to find x, t implying $F(x,y) + tI \ge 0$. ρ should be chosen in a way that ρt does not change the global minimum. By including t in the x variables we get a problem of the form (2). We can therefore assume that $F(x,y) \ge 0$ is feasible for all $y \in \mathcal{Y}$.

2.1 Upper bound

If we fix $y = \hat{y} \in \mathcal{Y}$ in (2) we obtain a semidefinite programming (SDP) problem

$$\begin{array}{ll} \text{minimize} & c^T x + d^T \hat{y} \\ \text{subject to} & F(x,\hat{y}) \ge 0, \\ & -l_i \le x_i \le u_i,\ i=1,\dots,m_x, \end{array} \quad (5)$$

We know that the problem is feasible. We will denote the optimum by $f(\hat{y})$, and the optimal x by $\hat{x}$. Since we have restricted the variable y in the BMI problem to $y = \hat{y}$, we call (5) the *restricted problem*.

Solving the SP (5) yields an upper bound on the optimal value of (2): $f(y) \ge f^*$ for all $y \in \mathcal{Y}$. Solving the BMI problem (2) is equivalent to minimizing the (non-convex) function f over $\mathcal{Y}$.

2.2 Lower bound

We introduce Lagrange variables $Z = Z^T \ge 0$, $\mu \ge 0$, $\nu \ge 0$ associated with (5). For future use we will introduce the Lagrangian:

$$\begin{aligned} L(x,y,Z,\mu,\nu) &\\ = \ & c^T x + d^T y - \mathbf{Tr} Z F(x,y) \\ & + \mu^T(-l - x) + \nu^T(-u + x) \\ = \ & d^T y - \mathbf{Tr} Z \left(F_0 + \sum_{j=1}^{m_y} y_j G_j\right) - \mu^T l - \nu^T u \\ & + \sum_{i=1}^{m} x_i \left(c_i - \mathbf{Tr} Z F_i - \sum_{j=1}^{m_y} y_j \mathbf{Tr} Z H_{ij} - \mu_i + \nu_i\right). \end{aligned}$$

This Lagrangian can be used to provide lower bounds valid over $\mathcal{Y}$. First observe that for all Z, μ, ν,

$$\left.\begin{array}{c} F(x,y) \ge 0 \\ x \in \mathcal{R}^x \end{array}\right\} \implies c^T x + d^T y \ge L(x,y,Z,\mu,\nu),$$

and therefore, for all $y \in \mathcal{Y}$,

$$\begin{aligned} f(y) \ge & \inf_{x \in \mathcal{R}^x} L(x,y,Z,\mu,\nu) \\ & = d^T y - \mathbf{Tr} Z \left(F_0 + \sum_{j=1}^{m_y} y_j G_j\right) - \mu^T l - \nu^T u \\ & + \inf_{x \in \mathcal{R}^x} \sum_{i=1}^{m} x_i \left(c_i - \mathbf{Tr} Z F_i - \sum_{j=1}^{m_y} y_j \mathbf{Tr} Z H_{ij} - \mu_i + \nu_i\right) \\ & = d^T y - \mathbf{Tr} Z \left(F_0 + \sum_{j=1}^{m_y} y_j G_j\right) - \mu^T l - \nu^T u \\ & + \sum_{i=1}^{m_x} \inf_{l_i \le x_i \le u_i} x_i \left(c_i - \mathbf{Tr} Z F_i - \sum_{j=1}^{m_y} y_j \mathbf{Tr} Z H_{ij} - \mu_i + \nu_i\right). \end{aligned} \quad (6)$$

Define

$$g_i(y) \triangleq c_i - \mathbf{Tr} Z F_i - \sum_{j=1}^{m_y} y_j \mathbf{Tr} Z H_{ij} - \mu_i + \nu_i.$$

The infimum of $L(x,y,Z,\mu,\nu)$ over $x \in \mathcal{R}^x$ can be found by computing $\inf_{l_i \le x_i \le u_i} x_i g(y)$ separately. For $g_i(y) > 0$ the optimal x_i is to choose l_i, and for $g_i(y) < 0$ the optimal x_i is u_i. For $g(y) = 0$ any choice will do. By introducing a sign variable s_i we can write the infimum in closed form

$$\begin{aligned} \inf_{x_i} x_i g_i(y) = & \frac{(u_i + l_i) - s_i(u_i - l_i)}{2} g_i(y) \\ & \text{for } s_i g_i(y) \ge 0,\ s_i \in \{-1, 1\}. \end{aligned} \quad (7)$$

Choosing a specific s_i for all $i = 1 \dots m_x$ the combination of halfspaces $\{y \mid s_i g_i(y) \ge 0\}$ defines a polyhedron in $\mathbf{R}^{m_y}$. On this polyhedron we can provide the following lower bound from (6):

$$f(y) \geq d^T y - \mathbf{Tr} Z \left(F_0 + \sum_{j=1}^{m_y} y_j G_j \right) - \mu^T l - \nu^T u + \sum_{i=1}^{m_x} \frac{(u_i + l_i) - s_i(u_i - l_i)}{2} g_i(y), \quad \text{for } s_i g_i(y) \geq 0, i = 1, \ldots, m_x \quad (8)$$

Considering all the possible sign vectors s, which corresponds to all the corners in $\mathcal{R}^x$, the right hand side of (8) defines a set of lower bounding functions (linear in y) that completely covers $\mathcal{Y}$. Combining these lower bounding functions we get piecewise linear, concave and continous function lying as a pyramid under $f(y)$. Therefore, to obtain a lower bound on f^*, we need to examine all 2^{m_x} vectors $s \in \mathbf{R}^{m_x}$ with $s_i \in \{-1, +1\}$, by solving the 2^{m_x} SDPs

$$\begin{array}{ll} \text{minimize} & d^T y - \mathbf{Tr} Z \,(F_0 + \sum_{j=1}^{m_y} y_j \; G_j) - \mu^T l - \nu^T u \\ & \quad + \sum_{i=1}^{m_x} \frac{(u_i + l_i) - s_i(u_i - l_i)}{2} g_i(y) \\ \text{subject to} & A(y) \geq 0 \\ & s_i g_i(y) \geq 0, \quad i = 1, \ldots, m_x \end{array}$$

in the variable y. We call this problem for the *region problem* associated with the sign vector s, and denote the optimal solution by ϕ_s^*. We then have

$$f^* \geq \min_s \phi_s^*.$$

The lower bound (8) is valid for any choice of $Z \geq 0$, $\mu, \nu \geq 0$. In the next section we discuss a special choice which follows from the dual of the SDP (5).

2.3 Star-shaped partition

The dual problem of (5) follows from the Lagrangian $L(x, \hat{y}, Z, \mu, \nu)$, and is an SDP

$$\begin{array}{l} \text{maximize } L(0, \hat{y}, Z, \mu, \nu) \\ \text{subject to } \frac{\partial}{\partial x} L(x, \hat{y}, Z, \mu, \nu) = 0 \\ \qquad Z \geq 0, \mu \geq 0, \nu \geq 0 \end{array} \quad (9)$$

with variables $Z = Z^T \in \mathbf{R}^{n \times n}$, $\mu, \nu \in \mathbf{R}^m$, or, more explicitly,

$$\begin{array}{ll} \text{maximize} & d^T \hat{y} - \mathbf{Tr} Z \left(F_0 + \sum_{j=1}^{m_y} \hat{y}_j G_j \right) - \mu^T l - \nu^T u \\ \text{subject to} & c_i - \mathbf{Tr} Z F_i - \sum_{j=1}^{m_y} \hat{y}_j \mathbf{Tr} Z H_{ij} - \mu_i + \nu_i = 0 \\ & Z \geq 0, \mu \geq 0, \nu \geq 0. \end{array} \quad (10)$$

This dual SDP is always strictly feasible and therefore its optimal value is equal to $f(y)$, *i.e.*, the optimal value of (5) (see [VB96]).

We denote the primal optimal solution of (5) by $\hat{x}$, and the dual optimal solution of (10) by $\hat{Z}$, $\hat{\mu}$, $\hat{\nu}$. Due to the equality in (10) and the zero duality gap the objective of the dual problem equals $f(\hat{y})$, i.e.

$$f(\hat{y}) = d^T \hat{y} - \mathbf{Tr} Z \left(F_0 + \sum_{j=1}^{m_y} \hat{y}_j G_j \right) - \mu^T l - \nu^T u.$$

If we choose $Z = \hat{Z}$, $\mu = \hat{\mu}$, $\nu = \hat{\nu}$ in the lower bound (8), then the expression simplifies to

$$f(y) \geq f(\hat{y}) + \sum_{i=1}^{m_x} \frac{(u_i + l_i) - s_i(u_i - l_i)}{2} g_i(y), \quad (11)$$

where

$$\begin{aligned} g_i(y) &= c_i - \mathbf{Tr} \hat{Z} F_i - \sum_{j=1}^{m_y} y_j \mathbf{Tr} \hat{Z} H_{ij} - \hat{\mu}_i + \hat{\nu}_i \\ &= -\sum_{j=1}^{m_y} (y_j - \hat{y}_j) \mathbf{Tr} \hat{Z} H_{ij}. \end{aligned} \quad (12)$$

The last equality follows by subtracting $0 = c_i - \mathbf{Tr} \hat{Z} F_i - \sum_{j=1}^{m_y} \hat{y}_j \mathbf{Tr} \hat{Z} H_{ij} - \hat{\mu}_i + \hat{\nu}_i$.

From this we conclude that $\hat{Z}$, $\hat{\mu}$, $\hat{\nu}$ define a partition with hyperplanes that pass through $\hat{y}$, and that the piecewise-linear lower bound is equal to $f(\hat{y})$ at $\hat{y}$. We call this a *star-shaped* partition.

As a function of a sign vector we can characterize each region $\mathcal{Q}_{\hat{y}}(s)$ as

$$\mathcal{Q}_{\hat{y}}(s) \triangleq \left\{ y \;\middle|\; \begin{array}{c} s_i g_i(y) \geq 0, \\ i = 1, \ldots, m. \end{array} \right\}. \quad (13)$$

Each inequality $s_i g_i(y) \geq 0$ defines a half space in $\mathcal{Y}$ with the dividing hyperplane going through $\hat{y}$. Combining the m_x inequalities we get a star-shaped partition of $\mathcal{Y}$.

On each such region we get an affine lower bound

$$h_{\hat{y}}(y, s) \triangleq f(\hat{y}) - \sum_{i=1}^{m} \frac{(u_i + l_i) - s_i(u_i - l_i)}{2} g_i(y). \quad (14)$$

That is for a given s we have

$$f(y) \geq h_{\hat{y}}(y, s), \quad \forall y \in \mathcal{Q}_{\hat{y}}(s).$$

2.4 Refinements

Suppose we know that $\underline{f} \leq f^* \leq \overline{f}$, then the following simple SDP provides a lower bound on the variable x_i

$$\begin{array}{ll} \text{minimize} & x_i \\ \text{subject to} & c^T x + d^T y \leq \overline{f} \\ & c^T x + d^T y \geq \underline{f} \\ & A(y) \geq 0 \\ & l_i \leq x_i \leq u_i, \; i = 1, \ldots, m_x. \end{array}$$

Similarly upper and lower bounds can be found for all x_i, y_j. Exploiting other features of the algorithm even tighter bounds can be obtained.

The computation of the lower bounds requires that we solve 2^{m_x} SDP problems. This number of SDP's can often be reduced. For instance, if $g_i(y)$ is zero for all y then x_i has no influence on the lower bound, and the number of SDP's to be solved is cut down to the half.

3 The algorithm

The algorithm presented here is an extension of the generalized Benders decomposition procedure [Geo72], and is based on the more recent work developed in [FV93, VFIP96].

The algorithm builds an approximation to $f(y)$ over $\mathcal{Y}$ from below. At iteration k, we solve (5) and its dual (10) for some y^k. We denote the optimal x by x^k and the optimal dual solutions by Z^k, μ^k, ν^k. By solving (5) at y^k we do not only obtain $f(y^k)$, but also, as explained above, a set of the linear lower bounds (14) on a star-shaped partition (13). By taking the maximum of this lower bound and the lower bounds from previous iterations we get a piecewise linear approximation to the function f.

At the first iteration there is only one region, namely $\mathcal{P}$. In each iteration that follows, one region is selected and further partitioned.

Tree structure

The progress of the algorithm can be represented as a tree where each node corresponds to a region. All nodes in the tree that are not leaves correspond to regions that have been considered in a previous iteration. Associated with each such node is a vector y^k and the primal and dual solution to (5), (10). From these variables the star-shaped partition of the region and the lower-bounding function can be reconstructed.

The leaves of the tree correspond to the smallest regions in the current partition. Associated with each leaf is the computed lower bound on that region. At the next iteration the leaf with the lowest lower bound is selected for further partitioning. The solution y where the lower bound is attained becomes the new y^k. An upper bound is computed, and the star-shaped partition of that region along with lower bounding function are computed. The computed optimal values and dual variables are saved for each iteration.

Inheritance

Tracing back in the tree it is possible to obtain additional information from all the parents, *i.e.*, all regions that contain the region represented by the current leaf node under examination. Each parent supplies a lower bounding function valid for all its children. This gives a list $\mathcal{L}$ of iteration numbers from which a region and lower bound function can be inherited. For iteration l in the list $\mathcal{L}$ a unique sign vector s^l exists such $y^k \in \mathcal{Q}_{y^l}(s^l)$. The polyhedron $\mathcal{Q}_{y^l}(s^l)$ and the associated lower function $h_{y^l}(y, s^l)$ provides additional information to the lower bounding functions computed around y^k.

Region problems

For each region in the star-shaped partition around y^k a set of lower bounding functions is available and by solving the next region problem, a new (and better) lower bound is computed.

Let s be the sign vector associated with a given region, then the solution to the following *region problem* provides a lower bound for $f(y)$ over that region:

$$\begin{array}{ll} \text{Min.} & \phi_L \\ \text{s.t.} & \phi_L \geq h_{y^k}(y, s) \\ & y \in \mathcal{Q}_{y^k}(s) \\ & \left.\begin{array}{l} \phi_L \geq h_{y^l}(y, s^l) \\ y \in \mathcal{Q}_{y^l}(s^l) \end{array}\right\} l \in \mathcal{L} \\ & y \in \mathcal{Y}. \end{array} \tag{15}$$

Termination and progress

The lowest of the upper bounds among the nodes that are not leaves in the tree provides an upper bound for the solution to (2) and the lowest of the lower bounds among the leaves provides a lower bound for the problem (2). When the difference between the upper and lower bound are below a prespecified value ϵ the optimal solution is picked out, and the algorithm is terminated. The region to be split next is the leaf with the lowest lower bound.

4 Simple control problem

The algorithm presented in last section has been implemented in Matlab. The SDP problems are solved by use of the package SP [VB94]. In this section we will show the result of the algorithm on simple control problem. These problems could have been solved as a generalized eigenvalue problem, but their simplicity allows us to find the global optimum using other methods. By doing this we can show that the algorithm does indeed find a global optimum.

Consider the following plant

$$\begin{bmatrix} \dot{x} \\ y \end{bmatrix} = \begin{bmatrix} A & B_u \\ C_y & 0 \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix}$$

where $x \in \mathbf{R}^n$, $x \in \mathbf{R}^{n_u}$,and $y \in \mathbf{R}^{n_y}$. We are looking for controllers of the form:

$$\begin{bmatrix} u \\ \dot{x}_c \end{bmatrix} = \underbrace{\begin{bmatrix} D_c & C_c \\ B_c & A_c \end{bmatrix}}_{K} \begin{bmatrix} y \\ x_c \end{bmatrix}$$

where $K \in \mathbf{R}^{(n_u+n)\times(n+n_y)}$. The closed-loop can then be written as

$$\begin{bmatrix} \dot{x} \\ \dot{x}_c \end{bmatrix} = \underbrace{\left(\begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} B_u & 0 \\ 0 & I_{n_c} \end{bmatrix} K \begin{bmatrix} C_y & 0 \\ 0 & I_{n_c} \end{bmatrix} \right)}_{A_{cl}(K)} \begin{bmatrix} x \\ x_c \end{bmatrix}.$$

We now consider the control problem of maximizing the decay rate α, which can be cast as:

$$\begin{array}{ll} \text{minimize} & -\alpha \\ \text{subject to} & A_{cl}^T(K)\tilde{P} + \tilde{P}A_{cl}(K) + 2\alpha\tilde{P} \leq 0 \\ & P \geq 1/\kappa I, \quad \mathbf{Tr}\, P = n \\ & -l_k \leq K_{ij} \leq u_k, \quad i = 1 \cdots (n_u + n) \\ & \qquad\qquad\qquad\quad j = 1 \cdots (n_y + n) \\ & 0 \leq \alpha \leq \overline{\alpha} \end{array} \tag{16}$$

where the variables are K, α, and the matrix $\tilde{P}$. κ is a simple constant that defines how positive P should be. The inequality constraint $\mathbf{Tr}\, P = n$ can be removed by a simple variable transformation. Here we will put K and α in the x-variables and the transformed variables of the symmetric matrix $\tilde{P}$ in y. Since the BMI in (16) is (in general) not feasible for all P, we will need to add t as showned in (4).

Suppose we know that the optimal α is above $\underline{\alpha}$, we can add the following constraint

$$C_y^{T\perp}\left(A^T P + PA + 2\underline{\alpha}P\right) C_y^{T\perp T} < 0$$

where $^\perp$ denotes the left annihilator, i.e., $A^\perp$ satisfies Im A = Ker $A^\perp$ and $A^\perp$ has full row rank$P$ is the upper left part of $\tilde{P}$ of size $n \times n$. This is in fact one of the LMI's known from the free order LMI formulation, see for instance [SIG93, BGFB94]. By introducing more variables tighter bound can be added.

The form of K above is only appropriate for low controller orders, because the number of variables required to represent K grows with the square of n_c.

We consider two small examples, for which the optimum can be determined by root-locus techniques or by gridding the domain of one of the variables.

Example 1. Consider the simple example:

$$\begin{bmatrix} A & B_u \\ C_y & 0 \end{bmatrix} = \left[\begin{array}{cc|c} 0 & 1 & 1 \\ 1 & -1 & 0 \\ \hline 1 & 1 & 0 \end{array}\right]. \tag{17}$$

We will look for a static controller $-6 \leq K \leq -1$. By root-locus methods it is easy to find the global optimum (with the lower bound on P replaced by $P \geq 0$): $K = -5$ and $\alpha = -3$. Since $A_{cl}(-5) - (-3)I$ is singular we cannot find a positive definite P s.t. the Lyapunov inequality in (16) holds. We will choose $\kappa = 50$, and we will get a suboptimal solution. The extra parameter t lies between 0 and 100. It is weighted with $\rho = 10^2$. We know that the optimum will be less than -2, so we set $\underline{\alpha} = 2$.

We require the precision to be $\epsilon = 10^{\perp 2}$. The algorithm provides $K = -4.7637$ with closed loop poles at $-2.8818 \pm 0.4716j$, where as the optimal value from the solver is 2.8775. The optimal Lyapunov matrix is $P = \begin{bmatrix} 0.4187 & 0.7890 \\ 0.7890 & 1.5813 \end{bmatrix}$, which has the condition number 98.99. The number of iteration is 36. Since $x \in \mathbf{R}^2$ the number of region problems per iteration is 2^2. This means that $36(1+2^2) = 180$ SDP problems have been solved.

In figure 1 the upper and lower bound available at each iteration are shown.

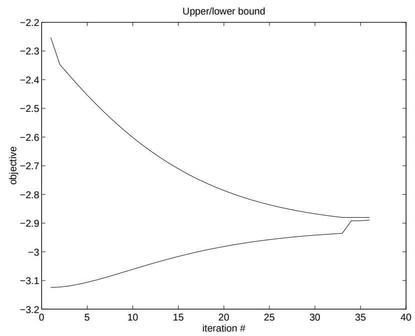

Figure 1: The evolution of the upper and lower bound for each iteration in the algorithm.

Example 2. Consider the slightly more complex example:

$$\begin{bmatrix} A & B_u \\ C_y & 0 \end{bmatrix} = \left[\begin{array}{ccc|c} 0 & 1 & 0 & 1 \\ 1 & -1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ \hline 1 & 0 & 2 & 0 \end{array}\right]. \tag{18}$$

we will search for a static controller between -1 and -10, that optimizes the decay rate between $-.1$ and 1.. We take $\kappa = 10$. Root-locus methods show the optimum to be between 1 and 1.1. Therefore we say $\underline{\alpha} = 1$, and we have again $0 \leq t \leq 100$ and $\rho = 1e2$

The algorithm finds $K = -9.4277$ with closed loop poles at $-7.31955, -1.0541 \pm 0.5600j$. The optimal Lyapunov matrix is

$$P = \begin{bmatrix} 0.2164 & -0.0411 & 0.5469 \\ -0.0411 & 0.1145 & -0.1929 \\ 0.5469 & -0.1929 & 2.6691 \end{bmatrix},$$

which has the condition number 28.0000. The number of iterations was 24. Trying the same with $\kappa = 20$ increases the number of iterations heavily. The difference is that in this case the original BMI in (16) becomes infeasible for some of the restricted problems.

5 Conclusion

In this paper we have presented a global BMI algorithm that extends a practically efficient method for bilinear optimization (the GOP method of [FV93]) to problems with matrix inequalities. As an example we solved two very small problems (which could have been solved more efficiently by other methods). However the algorithm is very general, and as such the results seems very promising. More numerical experimentation will be necessary to see if the same practical efficiencies achieved by GOP will be achieved for bilinear matrix inequalities as well.

References

[Ben62] J. F. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4:238–252, 1962.

[BG96] E. Beran and K. Grigoriadis. A combined alternating and semidefinite programming algorithm for low-order control design. In *Proceedings of the 13th World Congress of IFAC*, volume C, pages 85–90, 1996.

[BGFB94] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*. SIAM, 1994.

[EB94] L. El Ghaoui and V. Balakrishnan. Synthesis of fixed-structure controllers via numerical optimization. In *Proceedings 1994 IEEE Conf. on Decision and Control*, pages 2678–2683, 1994.

[FV93] C. A. Floudas and V. Visweswaran. A primal-relaxed dual global optimization approach. *Journal of Optimization Theory and Applications*, 78(2):178–225, 1993.

[Geo72] A. M. Geoffrion. Generalized Benders decomposition. *Journal of Optimization Theory and Applications*, 10(4):237–260, 1972.

[GLTS94] K. C. Goh, J. H. Ly, L. Turand, and M. G. Safonov. μ/k_m-synthesis via bilinear matrix inequalities. In *Proceedings of the 33rd IEEE Conference on Decision and Control*, December 1994.

[GSL95] K. C. Goh, M. G. Safonov, and J. H. Ly. Robust synthesis via bilinear matrix inequalities. 1995. Submitted to the *International Journal of Robust and Nonlinear Control.*

[GSP94] K. C. Goh, M. G. Safonov, and G. P. Papavassilopoulos. A global optimization approach for the BMI problem. In *Proceedings of the 33rd IEEE Conference on Decision and Control*, December 1994.

[MPS95] M. Mesbahi, G. P. Papavassilopoulos, and M. G. Safonov. Matrix cones, complementarity problems and the bilinear matrix inequality. In *Proc. IEEE Conf. on Decision and Control*, pages 3102–3107, 1995.

[Sch95] C. W. Scherer. From single-channel lmi analysis to multi-channel mixed lmi synthesis: A general procedure. In O. H. Bosgra, P. M. J. Van den Hof, and C. W. Scherer, editors, *Selected Topics in Identification, Modelling and Control*, volume 8, pages 1–8. Delft University Press, December 1995.

[SGL94] M. G. Safonov, K. C. Goh, and J. H. Ly. Control system synthesis via bilinear matrix inequalities. In *Proc. American Control Conf.*, pages 45–49, 1994.

[SIG93] R. E. Skelton, T. Iwasaki, and K. M. Grigoriadis. A unified approach to linear control design, 1993. Manuscript.

[TO95] O. Toker and H. Ozbay. On the np-hardness of solving bilinear matrix inequalities and simultaneous stabilization with static output feedback. In *Proc. American Control Conf.*, pages 2525–2526, 1995.

[VB94] L. Vandenberghe and S. Boyd. SP*: Software for Semidefinite Programming. User's Guide, Beta Version.* K.U. Leuven and Stanford University, October 1994.

[VB96] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM Review*, 38(1):49–95, March 1996.

[VFIP96] V. Visweswaran, C.A. Floudas, M.G. Ierapetritou, and E.N. Pistikopoulos. *A Decomposition-Based Global Optimization Approach for Solving Bilevel Linear and Quadratic Programs.* Kluwer Academic Publishers, 1996.