[11] M. L. Overton, *Large-scale optimization of eigenvalues*, SIAM J. Optimization 2 (1992), pp. 88-120.

[12] M.L. Overton and R.S. Womersley *Second derivatives for optimizing eigenvalues of symmetric matrices*, in preparation.

[13] X. Ye, New York University. Private communication.

References

[1] S. Boyd and L. El Ghaoui, *Method of centers for minimizing generalized eigenvalues*. Submitted to the special issue of Linear Algebra Appl. on Numerical Linear Algebra Methods in Control, Signals and Systems, 1992. (Revised February 10, 1993.)

[2] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear matrix inequalities in system and control theory*, in preparation.

[3] F.H. Clarke, *Optimization and Nonsmooth Analysis*, John Wiley, New York, 1983. Reprinted by SIAM, Philadelphia, 1990.

[4] R. Fletcher, *Second order corrections for non-differentiable optimization*, Lecture Notes in Mathematics 912, Springer-Verlag, 1982.

[5] M.K.H. Fan and B. Nekooie, *On minimizing the largest eigenvalue of a symmetric matrix*, Proceedings of 31st IEEE Conference on Decision and Control, Tuscon, AZ (Dec 1992).

[6] S. Friedland, J. Nocedal, and M. L. Overton, *The formulation and analysis of numerical methods for inverse eigenvalue problems*, SIAM J. Numer. Anal. 24 (1987), pp. 634-667.

[7] J.-P. Haeberly, *On shape optimizing the ratio of the first two eigenvalues of the Laplacian*, Computer Science Dept. Technical Report 586, New York University, 1991.

[8] F. Jarre. *An interior point method for minimizing the maximum eigenvalue of a linear combination of matrices*, SIAM J. Control and Opt., to appear.

[9] Y. Nesterov and A. Nemirovsky *An interior point method for generalized linear-fractional programming*, Technical report, USSR Acad. Sci. Centr. Econ. & Math. Inst., 32 Krasikova St., Moscow 117418, 1991.

[10] M. L. Overton, *On minimizing the maximum eigenvalue of a symmetric matrix*, SIAM J. Matrix Anal. Appl., 9 (1988), pp. 256-268.

each of dimension 10. The matrix B_0 was set to the identity matrix and the matrices $A_0, \ldots, A_{15}$ and $B_1, \ldots, B_{15}$ were generated using the Matlab random number generator, and symmetrizing. The matrices $B_1, \ldots, B_{15}$ were then scaled by a factor 0.05 in order to ensure the existence of a reasonable-sized domain with $B(x) > 0$. All parameters have the same values as in the previous example. The vector x is initialized with a random vector. The results are given in Tables 3 and 4. The hybrid algorithm attempted to switch to the local scheme at the thirteenth step since $\|d^{12}\| < \kappa$. The step was rejected, however, because it failed to achieve a reduction of $\lambda_{\max}$. The operation count for $\nu = 13$ is the sum of the cost of one local step and one iteration of the method of centers. The same phenomenon occurred at the fourteenth step. The algorithm switched permanently to the local scheme at the fifteenth step and quadratic convergence was established. The last line indicates the limits of double precision computation. (The data for this example is available from the authors.)

The multiplicity of $\lambda_{\max}$ at the computed solution x^* is 6, with block multiplicities 1, 2 and 3. The hybrid algorithm reduces the gap between λ_1 and λ_6 to 3.3×10^{-16}, while the method of centers reduces the gap to 1.4×10^{-5}. The final dual matrix U has block dimensions $(1, 2, 3)$, satisfies (7) and (9) to machine precision, and has eigenvalues 0.195 (first block), 0.074, 0.126 (second block), and 0.002, 0.244, and 0.359 (third block), demonstrating that (8) is also satisfied.

7 Conclusion

We have presented an algorithm for the optimization of the maximum eigenvalue of a symmetric definite pencil depending affinely on a vector of parameters. The algorithm combines a scheme based on the method of centers developed by Boyd and El Ghaoui [1] and a new local scheme exhibiting quadratically convergent behavior. The local scheme is an extension of the methods introduced in [10,11,12] to the case of matrix pencils.

ν	$\lambda_{\max}$	$\|d\|$	Mflops	N
1	0.7989467570479351	2.889803	41.0	9
⋮	⋮	⋮	⋮	⋮
13	0.3073787506822123	6.222999e-03	40.9	9
14	0.3073365132846426	4.171214e-03	40.9	9
15	0.3073153048002088	2.824038e-03	40.9	9
16	0.3073045769895835	1.916467e-03	40.9	9
17	0.3072991033588865	1.290257e-03	40.9	9
18	0.3072962829954534	8.516237e-04	40.9	9
19	0.3072948149058472	5.446555e-04	40.9	9
20	0.3072940435712944	3.348214e-04	40.9	9
21	0.3072936352938653	1.976833e-04	40.9	9
22	0.3072934180681651	1.127953e-04	40.9	9
23	0.3072933021081565	6.278650e-05	40.9	9
24	0.3072932400776711	3.439867e-05	40.9	9

Table 4: Method of Centers

the global algorithm finds the neighborhood of the solution very reliably, while the use of the local algorithm, once in the neighborhood of the solution, rapidly locates the solution to full precision. Note, specifically, the quadratic convergence of $\|d\|$ to zero once the local scheme is in effect. The eigenvalue $\lambda_{\max}$ has block multiplicities 1, 0, 0, 1 at the computed solution x^*. The hybrid algorithm reduces the gap between $\lambda_1(x^*)$ and $\lambda_2(x^*)$ to 1.4×10^{-15}, while the method of centers reduces it only to 5.4×10^{-7}. The dual matrix U has block dimensions $(1,0,0,1)$, corresponding to the multiplicity of $\lambda_{\max}$, and is computed by the hybrid algorithm in Step 3 of the local scheme. Its final value is found to be Diag(.583, .417) (to 3 digits), verifying that the optimality condition (8) is satisfied. Optimality conditions (7), (9) hold to machine precision.

The second example provides a better test of the new algorithm, since $\lambda_{\max}(x^*)$ has block multiplicities greater than one. There are fifteen variables and the matrices $A(x)$ and $B(x)$ are block diagonal with three blocks,

ν	$\lambda_{\max}$	$\|d\|$	Mflops	N
1	0.7989467570479351	2.889803	41.0	9
2	0.5279030947496347	1.161417	41.0	9
3	0.4096247700327728	5.185782e-01	41.0	9
4	0.3564700009737637	2.556177e-01	41.0	9
5	0.3313368545063169	1.409038e-01	40.9	9
6	0.3190742531948093	8.384906e-02	40.9	9
7	0.3130620662624993	5.438315e-02	40.9	9
8	0.3101212297584327	3.939520e-02	40.9	9
9	0.3086835410941962	2.939005e-02	40.9	9
10	0.3079797136097306	2.094047e-02	40.9	9
11	0.3076340074134891	1.419033e-02	40.9	9
12	0.3074633984606852	9.392075e-03	40.9	9
13	0.3073787506822123	6.222999e-03	46.3	9
14	0.3073365132846426	4.171214e-03	46.3	9
15	0.3072932213520559	8.185445e-03	5.7	0
16	0.3072931684710006	3.461869e-05	5.7	0
17	0.3072931684689409	1.382263e-09	5.7	0
18	0.3072931684689404	2.081514e-15	5.7	0

Table 3: Hybrid

method of centers alone. In both cases the results use our implementation of the method of centers, as described in [1]. The number N denotes the number of Newton steps required by the inner iteration which computes the analytic center. This is zero once the hybrid algorithm has switched to the local scheme. The expression $\|d\|$ refers to the norm of $x^{\nu+1} - x^{\nu}$. The hybrid algorithm terminates when $\|d\| < 10^{-10}$. In the case of Table 2, the stopping criterion is $\bar{\rho} - \lambda_{\max}(\bar{x}) < 10^{-10}$. The *Mflops* column displays the number of floating point operations required, in millions. The large number of operations reflects the fact that the inner iteration required by the method of centers is being performed accurately; this number could undoubtedly be reduced. The significant point, however, is the following:

6 Numerical Results

We now present some numerical results. The algorithms were implemented in Matlab.

ν	$\lambda_{\max}$	$\|d\|$	Mflops	N
1	2.56696348918400	2.937e-01	1.20	7
2	1.65996563727620	1.597e-01	1.54	9
3	1.08305393696925	1.002e-01	1.54	9
4	0.75949101184420	8.061e-02	1.54	9
5	0.66424777849618	6.591e-02	1.37	8
6	0.66109151988020	3.149e-02	1.18	7
7	0.66063603170038	2.594e-03	1.19	7
8	0.66055966458506	1.785e-03	0.14	0
9	0.66055960982024	1.358e-06	0.14	0
10	0.66055960981957	5.684e-11	0.14	0

Table 1: Hybrid

ν	$\lambda_{\max}$	$\|d\|$	Mflops	N
1	2.56696348918400	2.937e-01	1.20	7
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
7	0.66063603170038	2.594e-03	1.19	7
8	0.66057058873535	4.167e-04	1.19	7
9	0.66056118766552	5.915e-05	1.19	7
10	0.66055983654089	1.960e-05	1.36	8

Table 2: Method of Centers

We first consider the example presented in [1]. The matrices $A(x)$ and $B(x)$ are block diagonal with four blocks of dimension 4. There are nine variables. We set $\mu = 0.0001$, $x_{\text{sup}} = 50$, $\sigma = 0.001$. The threshold value κ, determining when to attempt to switch to the local algorithm, is set to 0.01. The vector x is initialized to $(1, 1, 1, 0, 0, 0, 0, 0, 0)$. The results of the hybrid algorithm are given in Table 1 while Table 2 contains the output from the

Observe that no line search is performed. If the new iterate fails to produce a reduction of $\lambda_{\max}$, the step $\hat{d}$ is rejected altogether, and a new step is computed using the method of centers. This is discussed further in the next section.

5 The Global Algorithm

The global algorithm is a two-stage process. In the first stage, a sequence of iterates x^ν is computed using the method of centers (Boyd and El Ghaoui [1]) until the norm of the step $x^{\nu+1} - x^\nu$ is reduced below a certain threshold. In the second stage, we proceed as follows. First, we compute a step d^ν by solving the quadratic problem described in the previous section. If the point $x^\nu + d^\nu$ is feasible, i.e. $B(x^\nu + d^\nu) > 0$, and if d^ν is such that $\lambda_{\max}(x^\nu + d^\nu) < \lambda_{\max}(x^\nu)$, then we set $x^{\nu+1} = x^\nu + d^\nu$. Otherwise, $x^{\nu+1}$ is computed via the method of centers.

The algorithm of Boyd and El Ghaoui is thoroughly discussed in [1], so we sketch only the basic step. Consider the matrix inequality $C(x) > 0$ where $C(x)$ is the block diagonal matrix

$$C(x) = (\rho B(x) - A(x)) \oplus (B(x) - \mu I) \oplus (x_{\sup} I - \mathrm{Diag}(x)) \oplus (x_{\sup} I + \mathrm{Diag}(x)).$$

Here μ is a small constant used to ensure that $B(x)$ remains in the interior of the positive definite cone, while $x_{\sup}$ is a large constant used to ensure that x remains bounded. For a fixed ρ let $x^*(\rho)$ denote the analytic center of the inequality $C(x) > 0$. Now for ρ^ν and x^ν with $C(x^\nu) > 0$ let

$$x^{\nu+1} = x^*(\rho^\nu)$$

and

$$\rho^{\nu+1} = (1 - \sigma)\lambda_{\max}(x^\nu) + \sigma\rho^\nu,$$

where $0 < \sigma < 1$ is a parameter kept fixed throughout the algorithm. The analytic center is computed via Newton's method applied to the logarithmic barrier function $\log\det C^{-1}(x)$, using an exact line search for the computation of the step length.

Step 0. Initialize $\hat{x}$.

Step 1. Compute $A(\hat{x})$, $B(\hat{x})$, the Choleski factor $G(\hat{x})$, the multiplicity estimate t, the eigenvalues $\Lambda_{\max}(\hat{x})$ and the eigenvectors $Q_{\max}(\hat{x})$.

Step 2. Compute the matrix K and the vector h from (18) and compute the QR-factorization of K^T, which is used in the implementation of the next two steps.

Step 3. Compute an estimate $U = U^1 \oplus \ldots \oplus U^l$ of the dual matrix. This is done by computing the least square solution v to the equation

$$K^T v = e_1,$$

where $e_1 = (1, 0, \ldots, 0) \in \Re^{m+1}$. Note that the estimate from the previous iterate is useless because the basis of eigenvectors may have rotated an arbitrary amount. The vector v has dimension

$$\sum_{j=1}^{l} \frac{t_j(t_j+1)}{2}$$

and can be assembled into the matrix U. Compute the eigenvalues of U. If U has a negative eigenvalue, split the maximum eigenvalue as explained in theorem 2 and go to step 5.

Step 4. Use the dual matrix estimate U to compute the matrix W and solve the quadratic program (17), (18) to obtain a step $\hat{d}$. If $\|\hat{d}\| < 1$, a second order correction is computed to avoid the Maratos effect (see [4]). This is unnecessary in most cases but the cost is negligible.

Step 5. If $\|\hat{d}\|$ is less than a certain convergence tolerance, stop; otherwise, set $\hat{x} = \hat{x} + \hat{d}$, and go to step 1.

Given a nonsingularity condition, the iteration just described is locally quadratically convergent to a minimizer of $\lambda_{\max}(x)$. This property is demonstrated by the numerical results in Section 6. Proving this assertion requires extension of the results in [12] and is beyond the scope of this paper.

where:

1. $H^j_{pq,1} = M_1 + M_1^T$ with
$$M_1 = -\hat{Q}^{j\ T}_{\max}\hat{L}^j_{pq}(\hat{G}^j)^T\hat{Q}^j_{\max}\hat{\Lambda}^j_{\max}$$

2. $H^j_{pq,2} = M_2 + M_2^T$ with
$$M_2 = -\hat{Q}^{j\ T}_{\max}\hat{L}^j_{p}(\hat{G}^j)^{-1}\hat{A}^j(\hat{G}^j)^{-T}(\hat{L}^j_{q})^T\hat{Q}^j_{\max}$$

3. $H^j_{pq,3} = M_3 + M_3^T$ with
$$M_3 = -\hat{Q}^{j\ T}_{\max}\hat{L}^j_{p}(\hat{G}^j)^{-1}\hat{Z}^j_{q}\hat{Q}^j_{\max} - \hat{Q}^{j\ T}_{\max}\hat{L}^j_{q}(\hat{G}^j)^{-1}\hat{Z}^j_{p}\hat{Q}^j_{\max}$$

4. $H^j_{pq,4} = M_4 + M_4^T$ with $M_4 = 0$ if $t_j = n_j$, and otherwise
$$M_4 = \hat{Q}^{j\ T}_{\max}\hat{Z}^j_{p}\hat{Q}^j_{\rm rest}(S^j)^{-1}\hat{Q}^{j\ T}_{\rm rest}\hat{Z}^j_{q}\hat{Q}^j_{\max},$$
where
$$S^j = \mathrm{Diag}(\hat{\lambda}_{\max} - \hat{\lambda}^j_{t_j+1}, \ldots, \hat{\lambda}_{\max} - \hat{\lambda}^j_{n_j}).$$

Observe that the computation of M_4 seems to require explicit knowledge of all the eigenvalues and eigenvectors of $(\hat{A}^j, \hat{B}^j)$ rather than just the first t_j. However, a clever observation of Xianjian Ye [13] allows us to compute M_4 using only $\hat{\Lambda}^j_{\max}$ and $\hat{Q}^j_{\max}$. Let
$$T^j = \hat{\lambda}_{\max}\hat{B}^j - \hat{A}^j + \hat{B}^j\hat{Q}^j_{\max}\hat{Q}^{j\ T}_{\max}\hat{B}^j.$$
Then the term $\hat{Q}^j_{\rm rest}(S^j)^{-1}\hat{Q}^{j\ T}_{\rm rest}$ in M_4 is given by
$$\hat{Q}^j_{\rm rest}(S^j)^{-1}\hat{Q}^{j\ T}_{\rm rest} = (T^j)^{-1} - \hat{Q}^j_{\max}\hat{Q}^{j\ T}_{\max}.$$
Hence all references to $\hat{Q}^j_{\rm rest}$ and $\hat{\Lambda}^j_{\rm rest}$ can be eliminated from the equations for H^j_{pq}.

Remark: In the case $B(x) = I$ and $A(x)$ is affine there is only one term contributing to the Hessian, namely $H_{pq,4}$ with $\hat{Z}_p$ and $\hat{Z}_q$ replaced by $\hat{A}_p$ and $\hat{A}_q$ respectively [10,12]. The reason is that the second derivatives of $A(x)$ are identically zero. In the case of an affine pencil, however, the second derivatives of $A(x)$ and $B(x)$ are zero, of course, but those of the Choleski factor $G(x)$ are not. Indeed, they are the matrices $L_{ij}(x)$. Hence the terms $H_{pq,1}$, $H_{pq,2}$, and $H_{pq,3}$.

We now describe our implementation of the local algorithm.

skew-symmetric, and it follows that (x, Y, ω, θ) solves (18) if and only if the pencil $(A^j(x), B^j(x))$ has eigenvalues

$$(\underbrace{\omega, \ldots, \omega}_{t_j}, \theta^j_{t_j+1}, \ldots, \theta^j_{n_j}).$$

In the case $t_j = 0$, the latter condition is an empty one, indicating that the block F^j can be omitted from F in this case. The idea of replacing the nondifferentiable constraints (16) by an equation of the form (18) based on a matrix exponential formulation goes back to [6]. Indeed, consider the case where the number of variables and constraints in the quadratic program (11), (12) are the same, i.e. K is square, or equivalently, the quantity in (13) equals $m + 1$. Then, provided K is nonsingular, the solution of the quadratic program is completely defined by the constraint (12). In the case that $B(x) = I$, this essentially reduces to the step defined by Modified Method I in [6], the only difference being that in [6], the multiple eigenvalue $\lambda_{\max}$ is prescribed. This iteration can be viewed as a variation on Newton's method for solving (18), but the analysis is somewhat complicated because (a) the definition of F, which depends on $\hat{Q}$, changes at every step of the iteration and (b) it is necessary to remove the leading t_j by t_j block of the variables Y^j from the formulation of F, in order to obtain a well posed iteration, leading to a sequence of nonlinear equations which are solvable only in the limit. Further details are available in [6], [12].

The matrix W is derived from the Hessian of the Lagrangian function associated with the nonlinear program (17), (18), evaluated at $x = \hat{x}$, $\omega = \hat{\lambda}_{\max}$, $Y^j = 0$, and $\Theta^j = \hat{\Lambda}^j_{\text{rest}}$. A detailed discussion in the case $B(x) = I$ is given in [12]. Specifically, W is an $m \times m$ symmetric matrix whose (p, q)-entry is given by

$$w_{pq} = \sum_{j=1}^{l} \langle U^j, H^j_{pq} \rangle$$

where H^j_{pq} is a symmetric $t_j \times t_j$ matrix computed as follows. It is empty if $t_j = 0$. Otherwise let $\hat{G}^j = G^j(\hat{x})$ be the Choleski factor of $B^j(\hat{x})$, let $\hat{L}^j_p$ denote the matrix defined by (2) for the block $B^j(\hat{x})$ and similarly $\hat{L}^j_{pq}$ for the matrix defined by (4). Finally write $\hat{Z}^j_p = Z^j_p(\hat{x})$. Then

$$H^j_{pq} = H^j_{pq,1} + H^j_{pq,2} + H^j_{pq,3} + H^j_{pq,4}$$

$$\min_{x,\omega} \quad \omega$$

subject to

$$\lambda_1^j(x) = \ldots = \lambda_{t_j}^j(x) = \omega, \quad 1 \le j \le l. \tag{16}$$

The objective function is now a smooth function of x and ω, but the constraints, of course, are not. We replace these nondifferentiable constraints with a system of nonlinear equations to get the following nonlinear program:

$$\min_{x,\omega} \quad \omega \tag{17}$$

subject to

$$F(x, Y, \omega, \theta) = 0, \tag{18}$$

where $F(x, Y, \omega, \theta)$ is a block diagonal symmetric $n \times n$ matrix

$$F(x, Y, \omega, \theta) = F^1(x, Y, \omega, \theta) \oplus \ldots \oplus F^l(x, Y, \omega, \theta).$$

The matrices $F^j(x, Y, \omega, \theta)$ are defined as follows. The vector θ is given by

$$\theta = (\theta_{t_1+1}^1, \ldots, \theta_{n_1}^1, \ldots, \theta_{t_l+1}^l, \ldots, \theta_{n_l}^l).$$

We write

$$\Theta^j = \mathrm{Diag}(\theta_{t_j+1}^j, \ldots, \theta_{n_j}^j).$$

Let

$$Y = Y^1 \oplus \ldots \oplus Y^l$$

where Y^j is a skew symmetric $n_j \times n_j$ matrix, and let

$$D = D^1 \oplus \ldots \oplus D^l$$

with $D^j = \omega I \oplus \Theta^j$, where I is the $t_j \times t_j$ identity matrix. Then

$$F^j(x, Y, \omega, \theta) = D^j - e^{-Y^j} \hat{Q}^{jT} (G^j)^{-1}(x) A^j(x) (G^j)^{-T}(x) \hat{Q}^j e^{Y^j},$$

where $G^j(x)$ is the Choleski factor of $B^j(x)$ and $\hat{Q}^j$ is the matrix of eigenvectors for the pencil $(A^j(\hat{x}), B^j(\hat{x}))$. Observe that the definition of F depends on $\hat{x}$ through $\hat{Q}$. The matrix exponential e^{Y^j} is orthogonal since Y^j is

with U^j empty if $t_j = 0$. If (d, ω) is a solution of (11), (12), then the optimality conditions for quadratic programs give

$$\sum_{j=1}^{l} \langle U^j, \hat{Q}^{j\,T}_{\max} \left(\hat{A}^j_i - \hat{\lambda}_{\max} \hat{B}^j_i \right) \hat{Q}^j_{\max} \rangle \; + \; (Wd)_i = 0, \tag{14}$$

for $1 \leq i \leq m$ and

$$\operatorname{tr} U = \sum_{j=1}^{l} \operatorname{tr} U^j = 1. \tag{15}$$

Now observe that if the multiplicity tolerance $\tau = 0$, the vector h is zero and (12) can be rewritten as

$$\sum_{i=1}^{m} d_i \, \hat{Q}^{j\,T}_{\max} \left(\hat{A}^j_i - \hat{\lambda}_{\max} \hat{B}^j_i \right) \hat{Q}^j_{\max} = \omega I, \quad 1 \leq j \leq l,$$

and, since the matrices $\hat{Q}^{j\,T}_{\max}(\hat{A}^j_i - \hat{\lambda}_{\max}\hat{B}^j_i)\hat{Q}^j_{\max}$ are the diagonal blocks of the matrix $Q^T_{\max}(\hat{x})(A_i(\hat{x}) - \lambda_{\max}(\hat{x})B_i(\hat{x}))Q_{\max}(\hat{x})$ of corollary 1, we conclude that a solution (d, ω) of (12) satisfies

$$\lambda'_{\max}(\hat{x}; d) = \omega.$$

If, moreover, (d, ω) is a solution of (11) then

$$\omega + \frac{1}{2} d^T W d \leq 0,$$

since $(0, 0)$ is feasible. But W is positive definite so that

$$\lambda'_{\max}(\hat{x}; d) \leq -\frac{1}{2} d^T W d \leq 0.$$

Hence d is a descent direction for $\lambda_{\max}$ unless $d = 0$. But if $d = 0$ then (14) and (15) show that all optimality conditions for $\lambda_{\max}$ are satisfied at $\hat{x}$ except possibly the nonnegative definite condition on U. If U does have a negative eigenvalue theorem 2 shows how to obtain a descent direction by splitting $\lambda_{\max}$.

We next describe how to compute the matrix W. If the multiplicity t of $\lambda_{\max}$ at the optimal value x^* is known, the problem of minimizing $\lambda_{\max}$ over x can be rewritten as

$$\min_{\omega,d} \ \omega + \frac{1}{2} d^T W d \tag{11}$$

subject to

$$K \begin{pmatrix} \omega \\ d \end{pmatrix} = h, \tag{12}$$

where $\omega \in \Re$ and W is a positive definite matrix that will be described below. The matrix K is given by

$$K = \begin{bmatrix} K^1 \\ \vdots \\ K^l \end{bmatrix}$$

where K^j is the matrix

$$[\mathrm{vec}I \ \ -\mathrm{vec}(\hat{Q}^{j\,T}_{\max}(\hat{A}^j_1 - \hat{\lambda}_{\max}\hat{B}^j_1)\hat{Q}^j_{\max}) \cdots \ -\mathrm{vec}(\hat{Q}^{j\,T}_{\max}(\hat{A}^j_m - \hat{\lambda}_{\max}\hat{B}^j_m)\hat{Q}^j_{\max})],$$

with I the identity matrix of size $t_j \times t_j$ and $\hat{A}^j_i = \hat{A}^j_i(\hat{x})$, $\hat{B}^j_i = \hat{B}^j_i(\hat{x})$. Clearly, K^j is empty if $t_j = 0$. The vector h is given by

$$h = \begin{bmatrix} h^1 \\ \vdots \\ h^l \end{bmatrix}$$

where

$$h^j = \mathrm{vec}(\hat{\Lambda}^j_{\max} - \hat{\lambda}_{\max} I)$$

if $t_j > 0$ and h^j is empty otherwise.

The formulation of the quadratic program can be motivated as follows. The constraints (12) consist of

$$\sum_{j=1}^{l} \frac{t_j(t_j+1)}{2} \tag{13}$$

scalar constraints. The multipliers of these scalar constraints can be assembled into a block diagonal dual matrix estimate

$$U = U^1 \oplus \ldots \oplus U^l$$

$\hat{\lambda}_{\max} = \lambda_{\max}(\hat{x})$, $\hat{\lambda}_i = \lambda_i(\hat{x})$, we order the eigenvalues of each block in decreasing order,

$$\hat{\lambda}_1^j \geq \ldots \geq \hat{\lambda}_{n_j}^j, \quad 1 \leq j \leq l,$$

and we define $t_j \geq 0$ by

$$\hat{\lambda}_{\max} - \hat{\lambda}_{t_j}^j \leq \tau \max(1, |\hat{\lambda}_{\max}|),$$

and

$$\hat{\lambda}_{\max} - \hat{\lambda}_{t_j+1}^j > \tau \max(1, |\hat{\lambda}_{\max}|).$$

We write

$$\hat{\Lambda}_{\max}^j = \mathrm{Diag}(\hat{\lambda}_1^j, \ldots, \hat{\lambda}_{t_j}^j),$$

for the diagonal matrix consisting of the first t_j eigenvalues of the j^{th}-block. If $t_j = 0$, $\hat{\Lambda}_{\max}^j$ is an empty matrix. Let

$$\hat{\Lambda}_{\mathrm{rest}}^j = \mathrm{Diag}(\hat{\lambda}_{t_j+1}^j, \ldots, \hat{\lambda}_{n_j}^j)$$

and

$$\hat{\Lambda}^j = \hat{\Lambda}_{\max}^j \oplus \hat{\Lambda}_{\mathrm{rest}}^j.$$

Let $\hat{Q}$ be a matrix whose columns form a complete set of eigenvectors for the pencil at $\hat{x}$. Clearly $\hat{Q}$ is also block diagonal,

$$\hat{Q} = \hat{Q}^1 \oplus \ldots \oplus \hat{Q}^l.$$

We now write $\hat{Q}_{\max}$ for the matrix obtained from $\hat{Q}$ by discarding all but the first t_j eigenvectors of the j^{th} block, for $1 \leq j \leq l$. It is also block diagonal, i.e.

$$\hat{Q}_{\max} = \hat{Q}_{\max}^1 \oplus \ldots \oplus \hat{Q}_{\max}^l,$$

where $\hat{Q}_{\max}^j$, of size $n_j \times t_j$, corresponds to the j^{th}-block and is empty if $t_j = 0$. Finally let $\hat{Q}_{\mathrm{rest}}^j$ be such that

$$\hat{Q}^j = [\hat{Q}_{\max}^j \; \hat{Q}_{\mathrm{rest}}^j].$$

The quadratic program that is solved to compute the step $\hat{d}$ is given as follows:

Since optimality conditions (7) and (9) are satisfied, this equation reduces to

$$-\delta = \beta\theta,$$

The conclusion follows. ∎

It is now easy to generate a descent direction for $\lambda_{\max}(x)$ if U is indefinite. Indeed, U has a negative eigenvalue $\theta < 0$, so choosing β negative in the previous theorem produces d with $\lambda'_{\max}(x; d) < 0$.

4 The Local Algorithm

Consider a block diagonal pencil $(A(x), B(x))$ and let l denote the number of blocks. Thus

$$A(x) = A^1(x) \oplus \ldots \oplus A^l(x), \quad B(x) = B^1(x) \oplus \ldots \oplus B^l(x)$$

with $A^i(x)$, $B^i(x)$ in $\mathcal{SR}^{n_i \times n_i}$, $B^i(x) > 0$, for $1 \leq i \leq l$. Let $n = n_1 + \ldots + n_l$ be the dimension of the pencil and let $\lambda_{\max}(x)$ denote the largest eigenvalue of $(A(x), B(x))$.

We also assume that $A(x)$ and $B(x)$ are affine functions of x. Thus

$$A(x) = A_0 + x_1 A_1 + \ldots x_m A_m,$$

and

$$B(x) = B_0 + x_1 B_1 + \ldots x_m B_m.$$

Most of the content of this section is still valid for a twice continuously differentiable pencil-valued function $(A(x), B(x))$ with the notable exception of the computation of the matrix W introduced below.

Given an initial point x^0 in a neighborhood of a local minimizer x^* of $\lambda_{\max}(x)$, we show how to generate a sequence of iterates converging to x^*. Let $\hat{x}$ denote the current point. The new iterate is set to $\hat{x} + \hat{d}$ where $\hat{d}$ is the solution of a certain quadratic program. Let $t = (t_1, \ldots, t_l)$ be the vector of multiplicities for $\lambda^*_{\max} \equiv \lambda_{\max}(x^*)$, i.e. $t_j \geq 0$ is the multiplicity of $\lambda^*_{\max}$ in the block $(A^j(x^*), B^j(x^*))$. Of course, t is not known and needs to be estimated. This is done with the use of a multiplicity tolerance τ, based on the eigenvalues at the current iterate $\hat{x}$. More precisely, we write

Now for $v \in \partial\lambda_{\max}(x)$ we have

$$\begin{aligned}\langle v, d\rangle &= \sum_{j=1}^{m} v_j d_j \\ &= \sum_{j=1}^{m} d_j \langle U, Q_{\max}^T(x)\,(A_j(x) - \lambda_{\max}(x)B_j(x))\,Q_{\max}(x)\rangle \\ &= \langle U, \sum_{j=1}^{m} d_j Q_{\max}^T(x)\,(A_j(x) - \lambda_{\max}(x)B_j(x))\,Q_{\max}(x)\rangle.\end{aligned}$$

The result follows from equation (1). ∎

We next discuss how to generate a descent direction for $\lambda_{\max}(x)$ that splits the maximum eigenvalue in case we have found a dual matrix U satisfying all the optimality conditions except (8).

Theorem 2 *Let x and U satisfy (7) and (9). Let θ be an eigenvalue of U with normalized eigenvector v, and let $\beta \in \Re$. Suppose $d \in \Re^m$ and $\delta \in \Re$ form a solution of the following equation*

$$\sum_{j=1}^{m} d_j Q_{\max}^T(x)\,(A_j(x) - \lambda_{\max}(x)B_j(x))\,Q_{\max}(x) - \delta I = \beta v v^T. \qquad (10)$$

Then the directional derivative $\lambda'_{\max}(x;d)$ is given by

$$\lambda'_{\max}(x;d) = \begin{cases} \beta(1-\theta) & if \quad \beta > 0 \\ -\beta\theta & if \quad \beta \le 0 \end{cases}$$

Proof: Let us write

$$M(d) = \sum_{j=1}^{m} d_j Q_{\max}^T(x)\,(A_j(x) - \lambda_{\max}(x)B_j(x))\,Q_{\max}(x).$$

Then equation (10) gives

$$M(d) = \delta I + \beta v v^T,$$

so that all the eigenvalues of $M(d)$ are equal to δ except one which equals $\delta + \beta$. Thus $\lambda'_1(x;d) = \max\{\delta, \delta+\beta\}$ by corollary 1. Now, taking the inner product of equation (10) with the matrix U yields

$$\sum_{j=1}^{m} d_j \langle U, Q_{\max}^T(x)\,(A_j(x) - \lambda_{\max}(x)B_j(x))\,Q_{\max}(x)\rangle - \delta \,\mathrm{tr}\, U = \beta\theta.$$

$$
\begin{aligned}
Q_{\max}^T Z_j(x) Q_{\max} &= Q_{\max}^T [A_j(x) - L_j(x) G^{-1}(x) A(x) \\
&\quad - A(x) G^{-T}(x) L_j^T(x)] Q_{\max} \\
&= Q_{\max}^T [A_j(x) - \lambda_{\max}(L_j(x) G^{-1}(x) B(x) \\
&\quad + B(x) G^{-T}(x) L_j^T(x))] Q_{\max} \\
&= Q_{\max}^T [A_j(x) - \lambda_{\max}(L_j(x) G^T(x) + G(x) L_j^T(x))] Q_{\max} \\
&= Q_{\max}^T [A_j(x) - \lambda_{\max} B_j(x)] Q_{\max}.
\end{aligned}
$$

The result follows. ∎

A necessary condition for x to minimize $\lambda_{\max}(x)$ is that $0 \in \partial \lambda_{\max}(x)$. By theorem 1 this can be rewritten as follows. Let t denote the multiplicity of $\lambda_{\max}(x)$. Then a necessary condition for x to minimize $\lambda_{\max}$ is that there exists a dual matrix $U \in \mathcal{S}\Re^{t \times t}$ such that

$$
\begin{aligned}
\operatorname{tr} U &= 1 \qquad (7) \\
U &\geq 0 \qquad (8)
\end{aligned}
$$

and, for $1 \leq j \leq m$,

$$
\langle U, Q_{\max}^T(x) \left(A_j(x) - \lambda_{\max}(x) B_j(x) \right) Q_{\max}(x) \rangle = 0 \qquad (9)
$$

where $Q_{\max}(x)$ is as in theorem 1.

As a corollary of theorem 1 we compute the directional derivative of $\lambda_{\max}(x)$.

Corollary 1 *Let $\lambda'_{\max}(x; d)$ denote the directional derivative of $\lambda_{\max}$ in the direction d, i.e.*

$$
\lambda'_{\max}(x; d) = \lim_{\epsilon \downarrow 0} \frac{\lambda_{\max}(x + \epsilon d) - \lambda_{\max}(x)}{\epsilon}.
$$

Then $\lambda'_{\max}(x; d)$ is equal to the largest eigenvalue of the matrix

$$
\sum_{j=1}^{m} d_j Q_{\max}^T(x) \left(A_j(x) - \lambda_{\max}(x) B_j(x) \right) Q_{\max}(x).
$$

Proof: Since $\lambda_{\max}$ is regular at x the directional derivative $\lambda'_{\max}(x; d)$ exists and is given by (see [3, Proposition 2.1.2])

$$
\lambda'_{\max}(x; d) = \max\{\langle v, d \rangle \mid v \in \partial \lambda_{\max}(x)\}.
$$

and define $Q(x) = G^{-1}(x)P(x)$.

We are now ready to compute the generalized gradient of $\lambda_{\max}(x)$.

Theorem 1 *Assume that the multiplicity of $\lambda_{\max}(x)$ is t and let $Q_{\max}(x)$ be the submatrix of $Q(x)$ whose columns form a complete set of eigenvectors for $\lambda_{\max}(x)$. Then the generalized gradient of $\lambda_{\max}(x)$ is the set*

$$\partial\lambda_{\max}(x) = \left\{ v \in \Re^m \mid v_j = \langle U, Q_{\max}^T(x) \left(A_j(x) - \lambda_{\max}(x) B_j(x) \right) Q_{\max}(x) \rangle \right\} \tag{5}$$

where U runs over all nonnegative definite symmetric $t \times t$ matrices with $\operatorname{tr} U = 1$.

Proof: Let C denote an $n \times n$ symmetric matrix. The generalized gradient of the maximum eigenvalue $\lambda_{\max} = \lambda_{\max}(C)$ viewed as a function of C was computed in [11]. If the multiplicity of $\lambda_{\max}(C)$ is t and if $P_{\max}$ is an $n \times t$ matrix whose columns form a complete set of orthonormal eigenvectors for $\lambda_{\max}(C)$ then

$$\partial\lambda_{\max}(C) = \{ V \in \mathcal{S}\Re^{n \times n} \mid V = P_{\max} U P_{\max}^T \},$$

where U runs over the $t \times t$ symmetric matrices with $U \geq 0$ and $\operatorname{tr} U = 1$. We call these matrices *dual* matrices. The generalized gradient of the maximum eigenvalue $\lambda_{\max}(x)$ of the symmetric matrix $G^{-1}(x)A(x)G^{-T}(x)$ now follows from the chain rule (see [3]). We get

$$\partial\lambda_{\max}(x) = \{ v \in \Re^m \mid v_j = \langle V, G^{-1}(x) Z_j(x) G^{-T}(x) \rangle \} \tag{6}$$

with V as above. We have an equality in (6) instead of a mere inclusion because $\lambda_{\max}(x)$ is a regular function [3]. Indeed, equation (1) expresses $\lambda_{\max}(x)$ as the maximum over the matrices V of $\langle V, G^{-1}(x)A(x)G^{-T}(x) \rangle$, and, for a fixed V, the function

$$x \ \mapsto \ \langle V, G^{-1}(x) A(x) G^{-T}(x) \rangle$$

is differentiable, hence regular (see [3, theorems 2.8.2 and 2.8.6.]). We have $P_{\max} = P_{\max}(x)$ and we write $Q_{\max} = Q_{\max}(x) = G^{-T}(x) P_{\max}(x)$. Observe that

$$\langle P_{\max} U P_{\max}^T, G^{-1}(x) Z_j(x) G^{-T}(x) \rangle = \langle U, P_{\max}^T G^{-1}(x) Z_j(x) G^{-T}(x) P_{\max} \rangle.$$

Now the columns of $Q_{\max}$ are generalized eigenvectors for $\lambda_{\max} \equiv \lambda_{\max}(x)$ and we have

3 Optimality Conditions

Consider a symmetric definite pencil valued function $(A(x), B(x))$ of a vector of real parameters $x \in \Re^m$. We assume that $A(x)$ and $B(x)$ are twice continuously differentiable in x and we write $A_j(x)$, $B_j(x)$ for the partial derivatives of $A(x)$ and $B(x)$ with respect to x_j.

Let $G(x)$ denote the Choleski factor of $B(x)$ and let $L_j(x)$ denote the partial derivative of $G(x)$ with respect to x_j. Thus $L_j(x)$ is the unique lower triangular matrix that solves the equation

$$L_j(x)G^T(x) + G(x)L_j^T(x) = B_j(x). \tag{2}$$

Moreover, it follows that

$$\frac{\partial}{\partial x_j}G^{-1}(x) = -G^{-1}(x)L_j(x)G^{-1}(x),$$

and

$$\frac{\partial}{\partial x_j}(G^{-1}(x)A(x)G^{-T}(x)) = G^{-1}(x)Z_j(x)G^{-T}(x),$$

where the matrix $Z_j(x)$ is given by

$$Z_j(x) \equiv A_j(x) - L_j(x)G^{-1}(x)A(x) - A(x)G^{-T}(x)L_j^T(x). \tag{3}$$

For future reference, we also define

$$L_{ij}(x) = \frac{\partial}{\partial x_i}(L_j(x)). \tag{4}$$

Thus $L_{ij}(x)$ is that lower triangular matrix which solves the following equation.

$$L_{ij}(x)G^T(x) + G(x)L_{ij}^T(x) = B_{ij}(x) - [L_i(x)L_j^T(x) + L_j(x)L_i^T(x)].$$

Clearly, $L_{ij}(x) = L_{ji}(x)$.

Finally, write

$$\lambda_{\max}(x) = \lambda_1(x) \geq \ldots \geq \lambda_n(x)$$

for the eigenvalues of $(A(x), B(x))$, let $P(x)$ be an orthogonal matrix of eigenvectors for $G^{-1}(x)A(x)G^{-T}(x)$, so that

$$P^T(x)G^{-1}(x)A(x)G^{-T}(x)P(x) = \Lambda(x)$$

that B is positive definite. A symmetric definite pencil (A, B) consists of a pair of matrices A, B in $\mathcal{S}\Re^{n\times n}$ with $B > 0$.

We write $\langle\, , \rangle$ for the Frobenius inner product on the space of $n \times n$ matrices. Thus

$$\langle M, N\rangle = \text{tr } M^T N.$$

Let "vec" denote the operator mapping $\mathcal{S}\Re^{n\times n}$ into $\Re^{n(n+1)/2}$ defined by

$$\text{vec}A = (a_{11}, \sqrt{2}a_{12}, \ldots, \sqrt{2}a_{1n}, a_{22}, \sqrt{2}a_{23}, \ldots, \sqrt{2}a_{2n}, a_{33}, \sqrt{2}a_{34}, \ldots, a_{nn})$$

for $A = (a_{ij}) \in \mathcal{S}\Re^{n\times n}$. Observe that for two symmetric matrices M and N we have

$$(\text{vec}M)^T(\text{vec}N) = \langle M, N\rangle.$$

Also, let

$$C_1 \oplus \ldots \oplus C_k$$

denote the block diagonal matrix with blocks $C_1, \ldots, C_k$.

Given a symmetric definite pencil (A, B), let G denote the Choleski factor of B. Hence G is a lower triangular matrix with positive diagonal entries such that $GG^T = B$. Then the symmetric matrix $G^{-1}AG^{-T}$ has the same eigenvalues as the pencil. Let us write $\lambda_{\max} = \lambda_1 \geq \ldots \geq \lambda_n$ for these eigenvalues and Λ for the diagonal matrix

$$\Lambda = \text{Diag}(\lambda_1, \ldots, \lambda_n).$$

Let $p_1, p_2, \ldots, p_n$ denote a set of orthonormal eigenvectors for $G^{-1}AG^{-T}$, and let $P = [p_1 \ldots p_n]$ denote the orthogonal matrix with columns p_i, $1 \leq i \leq n$. Then the columns of the matrix $Q = G^{-T}P = [G^{-T}p_1, \ldots, G^{-T}p_n]$ are eigenvectors for the pencil. Hence we have

$$P^T G^{-1} A G^{-T} P = \Lambda, \quad P^T P = I$$

and

$$AQ = BQ\Lambda, \quad Q^T BQ = I.$$

We have the following characterization of $\lambda_{\max}$ [11].

$$\lambda_{\max} = \max\{\langle U, G^{-1}AG^{-T}\rangle \mid U \in \mathcal{S}\Re^{n\times n},\ \text{tr } U = 1,\ U \geq 0\} \qquad (1)$$

In particular, we see that $\lambda_{\max}$ is a convex function of A; however, it is only quasiconvex as a function of B [1].

bounds for the structured singular value and the computation of structured Lyapunov functions [2]. The salient feature of this problem is the lack of smoothness. Indeed, it is well known that the eigenvalues of a matrix are not differentiable as functions of the entries of the matrix when their multiplicity exceeds one. The optimization process, however, tends to force the first few eigenvalues to coalesce. Thus standard optimization techniques cannot be applied.

The special case $B = I$, that is the problem of minimizing the maximum eigenvalue of a symmetric matrix $A(x)$, has been studied extensively (see [11] and the references therein, as well as [5,8]). The case where $B(x)$ is constant is entirely similar and is briefly discussed in [11]. Algorithms have been developed in [1] and [9] to solve the problem when the pencil depends affinely on the parameter vector, and in [7] for the general case. The algorithm of Boyd and El Ghaoui in [1] is based on the method of centers and exhibits very good global behavior but slow local convergence.

We propose a hybrid algorithm, combining the robustness of the method of centers with rapid local convergence, to efficiently solve the affine case. More precisely, we propose to follow the path of centers to the vicinity of a solution and then to switch to a quadratically convergent local scheme. Such an approach was suggested in [1]. The local algorithm results from an extension of the work presented in [10,11,12] to a pencil valued function $(A(x), B(x))$. The algorithm is implemented so as to take full advantage of a block diagonal structure, if present. Block diagonal pencils occur frequently in applications from control theory.

The paper is organized as follows. Some notation and conventions are introduced in Section 2. The generalized gradient and directional derivatives of the maximum eigenvalue are computed in Section 3 and the optimality conditions are derived. The local algorithm is described in Section 4, and the global algorithm in Section 5. In Section 6 we present two numerical examples.

2 Preliminaries

Let $\mathcal{S}\Re^{n\times n}$ denote the set of $n \times n$ symmetric matrices. For $B \in \mathcal{S}\Re^{n\times n}$, the notation $B \geq 0$ means that B is nonnegative definite, and $B > 0$ means

A Hybrid Algorithm for Optimizing Eigenvalues of Symmetric Definite Pencils

Jean-Pierre A. Haeberly* and Michael L. Overton†

NYU Computer Science Dept. Technical Report 624
February 1993

Abstract

We present an algorithm for the optimization of the maximum eigenvalue of a symmetric definite pencil depending affinely on a vector of parameters. The algorithm uses a hybrid approach, combining a scheme based on the method of centers, developed by Boyd and El Ghaoui, with a new quadratically convergent local scheme. A convenient expression for the generalized gradient of the maximum eigenvalue of the pencil is also given, expressed in terms of a dual matrix. The algorithm computes the dual matrix which establishes the optimality of the computed solution.

1 Introduction

In this paper we consider the problem of minimizing the maximum eigenvalue of a symmetric definite pencil $(A(x), B(x))$ depending on a vector parameter $x \in \Re^m$. Many problems arising in control theory can be formulated in these terms. Most notable among these are the computation of

*Department of Mathematics, Fordham University.

†Computer Science Department, Courant Institute of Mathematical Sciences, New York University. The work of this author was supported in part by National Science Foundation Grant CCR-9101649.