A subspace method for unimodal symmetric eigenvalue optimization problems involving large scale matrices

Jeroen De Vlieger *Karl Meerbergen*

Report TW 616, July 2012



Katholieke Universiteit Leuven
Department of Computer Science
Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

A subspace method for unimodal symmetric eigenvalue optimization problems involving large scale matrices

Jeroen De Vlieger *Karl Meerbergen*

Report TW 616, July 2012

Department of Computer Science, KU Leuven

Abstract

We consider the solution of eigenvalue optimization problems involving large symmetric positive definite matrices. To cope with the large scale, we propose a subspace projection method, based on the concept of bundle methods. The theory is generally applicable to matrices that satisfy some smoothness conditions. For the important case of affine parameters, the objective function is quasi convex. We present two methods: one method builds in an iterative way, a small set of smooth constraints, so that the eigenvalue optimization problem can be solved by a smooth optimization method. The second method is a subspace projection or reduction method where in each iteration an eigenvalue optimization problem with small size matrices needs to be solved. Both methods are provably convergent for unimodal problems. Numerical examples are given to illustrate the theory.

Keywords : Eigenvalue optimization, quasi-convexity, large scale eigenvalue problems.
MSC : Primary : 15A18, 65F15, 90C90.

A subspace method for unimodal symmetric eigenvalue optimization problems involving large scale matrices

Jeroen Devlieger Karl Meerbergen

July 27, 2012

Abstract

We consider the solution of eigenvalue optimization problems involving large symmetric positive definite matrices. To cope with the large scale, we propose a subspace projection method, based on the concept of bundle methods. The theory is generally applicable to matrices that satisfy some smoothness conditions. For the important case of affine parameters, the objective function is quasi convex. We present two methods: one method builds in an iterative way, a small set of smooth constraints, so that the eigenvalue optimization problem can be solved by a smooth optimization method. The second method is a subspace projection or reduction method where in each iteration an eigenvalue optimization problem with small size matrices needs to be solved. Both methods are provably convergent for unimodal problems. Numerical examples are given to illustrate the theory.

1 Introduction

Consider the generalized symmetric eigenvalue problem

$$A(x)u = \lambda B(x)u \tag{1}$$

where matrices A and B depend on parameters $x \in \mathcal{H} \subset \mathbb{R}^p$. The set $\mathcal{H}$ is compact and is such that $B(x)$ is symmetric positive definite for all $x \in \mathcal{H}$. We assume that A and B are large and sparse. The parametric dependency can be affine but can be non-linear too [2]. Parametric eigenvalue problems arise, e.g., in the design of vibrating structures, where the design parameters are to be chosen so that the first eigenfrequency is maximized. Minimizing the largest eigenvalue or maximizing the smallest eigenvalue are related mathematical problems. We therefore only focus on the minimal maximum eigenvalue of (1) for $x \in \mathcal{H} \subset \mathbb{R}^p$, since this problem is better known in the literature on eigenvalue optimization.

When B is independent of x and is positive definite, and A has the form

$$A = A_0 + \sum_{j=1}^{p} x_j A_j \tag{2}$$

the maximum or largest eigenvalue of (1) is a convex function of x. Numerical methods for solving the optimization problem heavily rely on this property. Such methods are key ingredients for the solution of semidefinite programs (SDP). Therefore, the SDP literature presents efficient methods for the convex minimal maximum eigenvalue problem, usually, with small scale matrices. For large scale problems, bundle methods can be used [19]. They are subspace projection methods, where the large scale problem (1) is projected on a small scale problem, which is then solved by a standard SDP method [10]. The importance of bundle methods for large scale eigenvalue problems in terms of computational complexity is well motivated in [10]. A bundle or subspace projection method approximates the largest eigenvalue function by a better approximation than the classical quadric or other approximations on function evaluations and derivatives or gradients. (Since the current objective function may not be differentiable in all points, subgradients can be considered also.) The reduction method that we present interpolates the function value and its gradient, if the latter exists.

When, in addition, B has the form

$$B = B_0 + \sum_{j=1}^{p} x_j B_j \tag{3}$$

the optimization problem no longer is an SDP. The maximum eigenvalue function is not convex, but quasi-convex, and thus unimodal. SDP methods have been extended to this case, see, e.g., the interior point method in [9]. This method is particularly useful for small scale problems. Applications of this form arise from topology optimization [1]. When the parameter dependency is not affine, the eigenvalue function may be multimodal. However, we only consider the case where the optimization problem is unimodal, so that local methods can find the global minimum.

In this paper, we develop a reduction method for (1) and prove its convergence where A and B are large and sparse and depend on x. The dependency may be arbitrary, i.e., not necessarily affine, but we assume that the optimization problem is unimodal in $\mathcal{H}$ and $A(x)$ and $B(x)$ are smooth functions in x. The proof of convergence, developed in this paper, is based on the notion of *support* required for Kelley's cutting plane methods [12]. A support is a function whose value is smaller than the objective function, but matches the function value in a number of selected points. We rely on the fact that the Rayleigh quotient is real and is bounded from above by the largest eigenvalue. This property will allow us to define Lipschitz continuous supports of the eigenvalue function and then prove convergence of the bundle method.

The theoretical framework in the paper relies on two assumptions. We first assume that a good eigenvalue solver for (1) is available. Since we assume matrices are large, an iterative method should be used, such as the Lanczos method [8] or the Jacobi-Davidson method [18]. In the paper, the focus is not on the solution of (1) for a given x. We just assume a good eigenvalue solver for computing the largest (or smallest) eigenvalue is available. The cost of the optimization method will be expressed in terms of function evaluations, which are, in this case, the number of solves of (1). Also, we assume that eigenvalues are

computed exactly. See [14] for inexact eigenvalue solves in convex quadratic semi-definite programming. Since we present a subspace method, small scale eigenvalue problems will be solved also. We will employ the QZ method for these problems. We do not exploit possible reductions in computation time of solving (1) given the fact that a sequence of eigenvalue problems for different x is solved.

Second, we assume that the eigenvalue optimization problem can be solved efficiently when the matrices A and B have small size. This is required for solving the reduced problem obtained by projection on a subspace. Depending on the type of dependency on x, various methods can be used. Since the objective function may not be differentiable, gradient based methods such as BFGS are not preferred. One could use subgradient sampling instead [5]. Alternatively, we can use gradient free methods such as level-set methods and their generalizations for multi-variate problems. An overview is given in [19]. In this paper, we propose an alternative problem that is built from a sequence of smooth constraints. This problem can be written as a linear program, when B does not depend on x and A has affine dependence on x.

The paper is organized as follows. In §2, we present properties of the parametric eigenvalue, with focus on affine parameters. Section 3 presents the main convergence theory of an iterative procedure based on supports that leads to a smooth optimization problem. This is the main contribution of the paper. In §4, we present the subspace projection method, mentioned higher, which we call the reduction method in this paper. The theory from §3 is used to prove its convergence. Section 5 illustrates the theory with numerical examples, including one application from civil engineering. We close the paper with the main conclusions in §6.

We use the following notation throughout the paper. Matrices are denoted by Roman upper case letters and vectors by Roman lower case letters. The vector and matrix transpose are denote by $\cdot^T$. The inner product between vectors x and y is thus denoted as $y^T x$. Functions in one or more variables are also denoted by Roman lower case letters. Subscript i indicates the ith component of a vector. Superscript (i) indicates the ith iteration of an iterative process. Throughout the paper we use $u_{\max}$ to refer to an eigenvector of the maximal eigenvalue

$$u_{\max} : \mathcal{H} \to \mathbb{R}^n : x \mapsto u \in \text{eigenspace}(\lambda_{\max}(x)) \tag{4}$$

2 Properties of the minimal maximum eigenvalue

Formally, the *Minimal Maximal Eigenvalue Problem* (5) can be defined as a global optimization problem of the largest eigenvalue over a parametrized symmetric positive pencil:

$$\begin{aligned} \lambda^*_{\max} := \; & \min_x \max_{\lambda, v} \lambda \\ & \text{st} : \begin{cases} A(x)v = \lambda B(x)v, \\ x \in \mathcal{H} \subset \mathbb{R}^p, \\ v \in \mathbb{R}^n \end{cases} \end{aligned} \tag{5}$$

The term *MMEP* will be used throughout the paper to refer to both the actual optimization problem as well as a reference to the underlying generalized eigenvalue problem. The symmetric positive definite pencil $\big(A(x), B(x)\big)$ will therefore be called the *MMEP pencil*. From the context, it will be clear whether MMEP is the algebraic eigenvalue problem or the eigenvalue optimization problem.

We now review a number of important properties of the MMEP. Since matrices A and B are real and B is positive definite, the eigenvalues are *real* and thus can we impose the following ordering:

$$\lambda_{\max}(x) = \lambda_1(x) \geq \lambda_2(x) \geq \ldots \geq \lambda_n(x) = \lambda_{\min}(x) \quad , \quad \forall x \in \mathcal{H}$$

The eigenvalue optimization problem thus determines the minimum of $\lambda_{\max}$. The eigenvalues of a matrix are continuous functions of the coefficients of the matrix [20]. This also holds for generalized eigenvalue problems, when we assume B to be strictly positive definite. We will give a more precise statement in §3.2.

Eigenvalue functions cross each other when, for a given $x \in \mathcal{H}$, the largest eigenvalue has multiplicity higher than one. Therefore, the MMEP is usually a *non-smooth* optimization problem. This suggests the use of derivative free methods.

When the matrices depend on affine parameters, the eigenvalues enjoy nice properties, which we now review. In this case, there are A_i, B_i, $i = 0, \ldots, p$ such that

$$A(x) = A_0 + \sum_{i=1}^{p} x_i A_i \quad \in \mathbb{S}^n(\mathcal{H}), \tag{6a}$$

$$B(x) = B_0 + \sum_{i=1}^{p} x_i B_i \quad \in \mathbb{S}^n_{++}(\mathcal{H}), \tag{6b}$$

where $\mathbb{S}^n$ is the set of real symmetric matrices of dimension n and $\mathbb{S}^n_{++}$ the subset of $\mathbb{S}^n$ of strictly positive definite matrices. Consequently, the pencil $\big(A(x), B(x)\big)$ is symmetric positive definite on $\mathcal{H}$.

Consider the following illustration.

Example 1 *Figure 1 shows the five eigenvalues of a mono-variate* 5×5 *MMEP pencil for* $x \in [-4, 4]$ *with the following matrices:*

$$A_0 = \begin{bmatrix} 0.58 & 0.64 & 0.04 & -0.54 & -0.68 \\ 0.64 & 1.50 & -0.30 & -0.30 & 1.02 \\ 0.04 & -0.30 & -0.12 & -1.20 & 0.55 \\ -0.54 & -0.30 & -1.20 & -1.09 & -0.76 \\ -0.68 & 1.02 & 0.55 & -0.76 & 1.62 \end{bmatrix}, \quad A_1 = \begin{bmatrix} 1.92 & 0.07 & -1.54 & -1.31 & 0.28 \\ 0.07 & -0.95 & -0.20 & -0.05 & -0.58 \\ -1.54 & -0.20 & -0.72 & 0.45 & 0.17 \\ -1.31 & -0.05 & 0.45 & -1.88 & 0.33 \\ 0.28 & -0.58 & 0.17 & 0.33 & -0.16 \end{bmatrix},$$

$$B_0 = \begin{bmatrix} 52.62 & -0.62 & -0.94 & -1.20 & 0.41 \\ -0.62 & 50.93 & 0.44 & 0.72 & -0.28 \\ -0.94 & 0.44 & 50.86 & 0.25 & -0.71 \\ -1.20 & 0.72 & 0.25 & 51.55 & -0.80 \\ 0.41 & -0.28 & -0.71 & -0.80 & 51.98 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 1.61 & -0.16 & -0.11 & -0.24 & -1.27 \\ -0.16 & 0.98 & 0.24 & -0.56 & 1.77 \\ -0.11 & 0.24 & 1.44 & 1.38 & 0.58 \\ -0.24 & -0.56 & 1.38 & -1.89 & -0.02 \\ -1.27 & 1.77 & 0.58 & -0.02 & -0.12 \end{bmatrix},$$

Observe that $\lambda_{\max} : [-4, 4] \to \mathbb{R}$ *is continuous and unimodal, while the intermediate eigenvalues may have several local minima and maxima. Also note the non-differentiability of the optimum due to the crossing of two eigenvalue functions.*

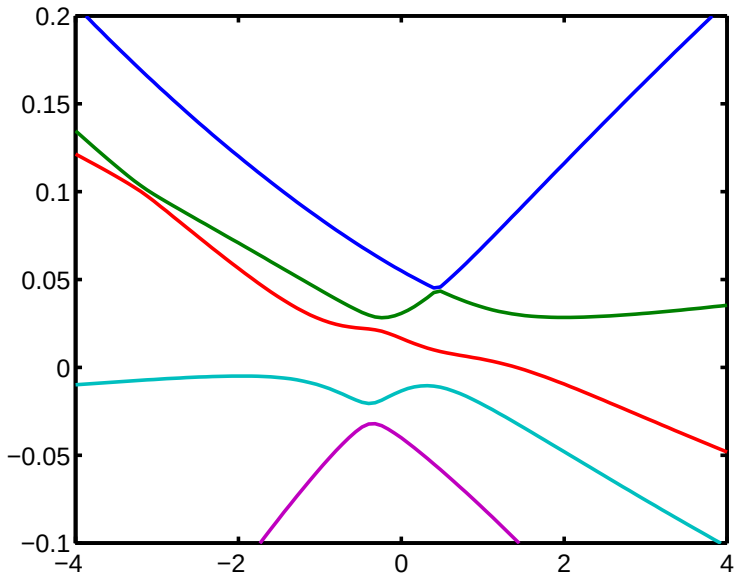


Figure 1: The eigenvalues of a mono-variate 5×5 MMEP pencil for $\mathcal{H} = [-4, 4]$. The eigenvalue $\lambda_i(x)$ is a real valued continuous but not differential function. In addition, $\lambda_{\max}(x)$ is unimodel.

Lemma 1 *Let (λ, v) be a simple eigenpair of (5), where A and B are defined by (6). The partial derivative of the eigenvalue is then equal to*

$$\frac{\partial \lambda}{\partial x_j} = \frac{v^T \big(A_j - \lambda B_j\big) v}{v^T B(x) v} \tag{7}$$

Note that (7) is not uniquely defined when the eigenvalue is not simple, since the corresponding eigenvector direction v is not unique. In practice this means that the eigenvalue functions are non-smooth in those points where the eigenvalue functions cross each other (see Figure 1). In the context of non-smooth functional analysis, we can however state that the 'gradient' of $\lambda_{\max}$ defined by (7) is a subgradient independent of the multiplicity where v is an arbitrary eigenvector associated with λ.

We define the *Rayleigh-Ritz ratio* or *Rayleigh quotient* of the MMEP pencil $\big(A(x), B(x)\big)$ as follows:

$$R : \mathcal{H} \times \mathbb{R}^n \setminus \{0\} \to \mathbb{R} : (x, v) \mapsto \frac{v^T A(x) v}{v^T B(x) v} \tag{8}$$

and the derived function

$$R_v : \mathcal{H} \to \mathbb{R} : x \mapsto R(x, v) \ . \tag{9}$$

with $v \in \mathbb{R}^n \setminus \{0\}$ a constant. Observe that R is analytical on the specified domain when A and B are analytical in x. The denominator has no roots because $B(x)$ is positive definite in $\mathcal{H}$.

Lemma 2 *Given the eigenpair (λ, v) of the MMEP pencil $\big(A(x^*), B(x^*)\big)$ for a given $x^* \in \mathcal{H}$, then*

$$R(x^*, v) = \lambda \tag{10}$$

meaning that the Rayleigh-Ritz ratio maps eigenvectors onto eigenvalues.

For the following lemma, see [17] and [11, Theorem 4.2.2,p. 176].

Lemma 3

$$\lambda_{\min}(x) \leq R(x, v) \leq \lambda_{\max}(x) \quad \textit{for all} \quad v \in \mathbb{R}^n, x \in \mathcal{H} \ . \tag{11}$$

The bounds in Lemma 3 are reached in the corresponding eigenvectors according to Lemma 2. This leads to the key property of the eigenvalue optimization problem that can now be stated as follows.

Corollary 4 *For all $x \in \mathcal{H}$:*

$$\lambda_{\max}(x) = \max_{v \in \mathbb{R}^n} R(x, v) \tag{12a}$$

$$\lambda_{\min}(x) = \min_{v \in \mathbb{R}^n} R(x, v) \tag{12b}$$

When x is chosen so that $B(x)$ is positive definite, the Rayleigh-Ritz ratio exists and is differentiable. The fact that Rayleigh-Ritz ratios are differentiable but eigenvalue functions are not will be used in §3.2.

We now formally introduce the concept of quasi-convexity and show its importance for the case of affine parameters. For a more extensive overview of alternative definitions, properties and corresponding proofs see, for example, [4].

Definition 1 ([4]) *Given a convex domain $\mathcal{D} \subseteq \mathbb{R}^p$. Then the function $f : \mathcal{D} \to \mathbb{R}$ is* quasi-convex *if and only if*

$$\forall a, b \in \mathcal{D}, \forall \tau \in [0, 1] \quad : \quad f(\tau a + (1 - \tau) b) \ \leq \ \max\big\{f(a), f(b)\big\} \ .$$

In words, this means that, just like convex functions, there can be *only one local minimum* which is simultaneously the global minimum of the function f on $\mathcal{D}$. As a consequence, the *global minimizer* of a quasi-convex function can be found using a *local optimization* approach. Proof. [Proof by contradiction] Assume that there are two different points $a, b \in \mathcal{D}$ such that $f(a) < f(b)$ for which f reaches a local minimum. If we now consider the line through these points $p(\tau) = \tau a + (1 - \tau) b$ then according to the definition, $f(p(t))$ cannot be larger then $f(b)$ for all $\tau \in [0, 1]$. This entails that the function $f(x)$ cannot increase and will eventually decrease when when we move from b to a meaning that $f(b)$ could not have been a local minimum. □

Lemma 5 *Given a convex domain $\mathcal{D} \in \mathbb{R}^n$ and a multi-variate function $f : \mathcal{D} \to \mathbb{R}$, then f is quasi-convex iff for all $a, b \in \mathcal{D}$ the mono-variate function $g : [0, 1] \to \mathbb{R} : \tau \mapsto f(\tau a + (1 - \tau) b)$ is quasi-convex.*

The lemma allows us to prove the quasi-convexity of multi-variate functions based on an analysis of mono-variate restrictions onto line segments which is usually much easier as we will see in the proof of the next lemma.

Lemma 6 *Let b, $d \in \mathbb{R}^p$ and α, $\gamma \in \mathbb{R}$, then the linear fractional function*

$$\mathbf{D} \to \mathbb{R} : x \mapsto \frac{\alpha + \sum x_i b_i}{\gamma + \sum x_i d_i} \tag{13}$$

is quasi-convex on the domain where its denominator is strictly positive:

$$\mathbf{D} = \{x \in \mathbb{R}^p \mid \gamma + \sum x_i d_i > 0\}. \tag{14}$$

Proof. The proof shows that any sub-levelset of a linear fractional function f with strictly positive denominator, restricted to the domain $\mathcal{D}$, is an intersection of 2 halfspaces. I.e. any sublevelset is convex and hence the function is quasi-convex.

Given a linear fractional function

$$f : \mathbb{R}^n \to \mathbb{R} : x \mapsto \frac{a + x^T b}{c + x^T d}$$

restricted to the domain

$$\mathcal{D} = x \in \mathbb{R}^n | c + x^T d > 0$$

Note that $\mathcal{D}$ is an open halfspace, hence its a convex set.

Then the α-sublevelset of f which is a subset of $\mathbb{R}^n$ is defined as follows:

$$x \in \mathbb{R}^n | f(x) \leq \alpha, x \in \mathcal{D} \ .$$

Substituting the expressions for f and rewritting the expression as an intersection, gives the following expression for an α-sublevelset of f:

$$x \in \mathbb{R}^n | (a - \alpha c) + x^t (b - \alpha d) \leq 0 \cap \mathcal{D} \ .$$

This is a convex set because a halfspace is a convex set and any intersection of convex sets is itself also convex. Hence a linear fractional function f is quasi-convex on $\mathcal{D}$, which proves the lemma. □

In the remainder of the paper, we assume that the Rayleigh-Ritz ratio (8) is Lipschitz continuous. The following lemma shows this for affine parameters.

Lemma 7 *In the case of affine parameters, $R_v(x)$ is a linear fractional function in x. This means that for any fixed $v \in \mathbb{R}^n$, there exists a finite constant K_R such that*

$$\left| \frac{\partial R(x; v)}{\partial x_i} \right| < K_R$$

for any $x \in \mathcal{H}$.

Proof. As mentioned earlier, the Rayleigh-Ritz ratio (8) is an analytical function on $\mathcal{H} \times \mathbb{R}^n$. Hence R_v is Lipschitz continuous on $\mathcal{H}$ iff the partial derivative to x:

$$\frac{\partial R_v(x)}{\partial x_i} = \frac{\partial R(x;v)}{\partial x_i}$$

with

$$\frac{\partial R(x;v)}{\partial x_i} = \frac{(v^T B(x) v)(v^T A_i v) - (v^T A(x) v)(v^T B_i v)}{(v^T B(x) v)^2} \tag{15}$$

is bounded on $\mathcal{H}$.

Given that all matrices involved are bounded then the denominator of (15) must also be bounded. Since $B(x) \in \mathbb{S}^n_{++}(\mathcal{H})$, it follows that the denominator of (15) is bounded below by some $\lambda_{\min}(B) > 0$. This means that for any fixed $v \in \mathbb{R}^n$, there exists a finite constant K_R such that

$$\left| \frac{\partial R(x;v)}{\partial x_i} \right| < K_R = \frac{\|B\|\|A_i\| + \|A\|\|B_i\|}{\lambda_{\min}(B)}$$

for all $x \in \mathcal{H}$. This proves the lemma. □

The following lemma is useful for proving the convergence theorem from Section 3.

Lemma 8 *Given a set of quasi-convex functions f_i on a convex domain $\mathcal{D} \subseteq \mathbb{R}^n$. Then*

$$g(x) = \max_i \{f_i(x)\}$$

is also quasi-convex on $\mathcal{D} \in \mathbb{R}^n$.

We now arrive at the key properties of MMEPs with affine parameters.

Theorem 9 *Given an MMEP pencil as defined by* (6)*, then the corresponding Rayleigh-Ritz ratio R is a quasi-convex function over the convex domain $\mathcal{H} \times \{v\}$.*

Proof. Note that the Rayleigh-Ritz ratio is a linear fractional function in $x \in \mathcal{H}$ for any fixed vector $v \in \mathbb{R}^n$:

$$R(x,v) = \frac{v^T A(x) v}{v^T B(x) v} = \frac{v^T A_0 v + \sum x_i v^T A_i v}{v^T B_0 v + \sum x_i v^T B_i v}$$

Combine this with the fact that its denominator is strictly positive on $\mathcal{H}$ because $B(x) \in \mathbb{S}^n_{++}(\mathcal{H})$ and the proof follows directly from Lemma 6. □

Theorem 10 *The largest eigenvalue $\lambda_{\max} : \mathcal{H} \to \mathbb{R}$ of an MMEP pencil with affine parameters is a quasi-convex function on the hypercube $\mathcal{H}$.*

Proof. Recall that (12a) formulates the maximal eigenvalue as the maximum over an infinite amount of Rayleigh-Ritz Ratios, while Lemma 9 shows us that every possible Rayleigh-Ritz ratio of an MMEP pencil is quasi-convex. The theorem now directly follows from the application of Lemma 8, which states that the maximum over a set of quasi-convex functions is itself quasi-convex. This is the proof of the theorem. □

Note that $\lambda_{\max}$ is a convex function of x in the case where $B \equiv B_0$. This case is extensively studied in the literature as mentioned in the Introduction. See Section 3.2 for a discussion of this case.

We now understand the properties of the MMEP allowing the development of a numerical method for solving it. Three comments are in order. (1) Local optimization approaches are tractable, since $\lambda_{\max}$ is unimodal. (2) A fundamental difficulty that needs to be addressed however, is the fact that the function is not everywhere differentiable. So we need to employ techniques that can deal with this non-smoothness. (3) The underlying matrices can potentially be very large, emphasizing the need for a reduction step.

3 Convergence theory for a sequence of supports

In this section, we introduce a methodology to compute the global minimum of a function $f : \mathcal{D} \to \mathbb{R}$ with $\mathcal{D}$ a compact set:

$$f^* := \min_{x \in \mathcal{D}} f(x) \ .$$

The idea is inspired by Kelley's classical Cutting Plane Method [12] for minimizing a convex problem. It relies on the existence of Lipschitz continuous supports.

Definition 2 *Given a function $f : \mathcal{D} \to \mathbb{R}$ and $t \in \mathcal{D}$.*

Define the set of supports *of f in t as*

$$\mathcal{S}(f,t) := \left\{ s : \mathcal{D} \to \mathbb{R} \;\middle|\; \begin{array}{l} \forall x \in \mathcal{D} : s(x) \leq f(x), \\ s(x)|_{x=t} = f(t) \end{array} \right\} \tag{16}$$

We call any function $s \in \mathcal{S}(f,t)$ a support *of f in t.*

Theorem 11 (convergence theorem) *Given a sequence of points $(x^{(i)}, \eta^{(i)}) \in \mathcal{D} \times \mathbb{R}$, with $i \in \mathcal{I}_k = 1, 2, \ldots, k$ and $\mathcal{D}$ a compact set. Each point of the sequence is related to all the previous points in the following way:*

$$\begin{aligned} (x^{(k)}, \eta^{(k)}) = \ & \min_{x,\eta} \eta \\ & s.t. : \begin{cases} \eta \geq s^{(i)}(x), & \forall i \in \mathcal{I}_{k-1}, \\ x \in \mathcal{D} \end{cases} \end{aligned} \tag{17}$$

We further assume that, for all indices $i \in \mathcal{I}_{k-1}$, the constraints $s^{(i)}$ are Lipschitz continuous supports with K_s a finite constant. In symbols:

$$s^{(i)} \in \mathcal{S}(f, x^{(i)}), \tag{18a}$$

$$\forall a, b \in \mathcal{D} : |s^{(i)}(a) - s^{(i)}(b)| \leq K_s \|a - b\|. \tag{18b}$$

The sequence $\eta^{(i)}$ then converges to f^*.

Proof. The proof of this theorem depends on the existence of Lipschitz continuous supports for any possible argument value. Assume that the sequence $\eta^{(k)}$ does not converge to f^*. This means that for any $k^* > 0$ there exists an $r > 0$ such that

$$\forall k \geq k^* : f^* - \eta^{(k)} \geq r > 0 \ . \tag{19}$$

Recall that $(x^{(k)}, \eta^{(k)})$ is the solution of (17). Therefore, the following constraints are satisfied:

$$\eta^{(k)} \geq s^{(i)}(x^{(k)}) \quad \forall i \in \mathcal{I}_{k-1} \ .$$

It then follows that

$$s^{(i)}(x^{(i)}) - s^{(i)}(x^{(k)}) \geq s^{(i)}(x^{(i)}) - \eta^{(k)}$$

and based on Definition 2, we can write this as

$$s^{(i)}(x^{(i)}) - s^{(i)}(x^{(k)}) \geq f(x^{(i)}) - \eta^{(k)}.$$

Since $f^* \leq f(x)$ for all $x \in \mathcal{D}$ and $s^{(i)}(x)$ is Lipschitz continuous, we conclude that

$$K_s \|x^{(i)} - x^{(k)}\| \geq s^{(i)}(x^{(i)}) - s^{(i)}(x^{(k)}) \geq f(x^{(i)}) - \eta^{(k)} \geq f^* - \eta^{(k)} \geq r > 0 \ .$$

This essentially means that

$$\forall i < k, k \geq k^* : \|x^{(i)} - x^{(k)}\| \geq \frac{r}{K_s} > 0. \tag{20}$$

It is not possible to construct an infinite sequence of points x_i in the compact set $\mathcal{D}$, such that the distance between any two points is larger than or equal to some finite value r. Therefore, the assumption that the sequence $y^{(k)}$ does not converge to the global minimum f^* is wrong. □

Lemma 12 *Given a function $f : \mathcal{D} \to \mathbb{R}$. Suppose that for any point $x \in \mathcal{D}$, there is a Lipschitz continuous support $s \in \mathcal{S}(f, x)$ with a Lipschitz constant K_s. Then f is also Lipschitz continuous with the same Lipschitz constant K_s.*

Proof. Given any two points $a, b \in \mathcal{D}$ such that

$$0 \leq f(a) - f(b)$$

Let $s \in \mathcal{S}(f, a)$ then we have that

$$0 \leq f(a) - f(b) \leq f(a) - s(b) = s(a) - s(b) \leq K_s \|a - b\|$$

and, so, we have

$$|f(a) - f(b)| \leq K_s \|a - b\|$$

which proves the lemma. □

In the following subsections we first explain how Lipschitz continuous supports can be obtained from Rayleigh Ritz ratios and then how the subproblem defined by (17), can be solved as a smooth MMEP of diagonal matrices.

3.1 Rayleigh-Ritz support functions

In this section, we consider the use of Rayleigh-Ritz ratios to construct Lipschitz continuous supports.

Theorem 13 *Let $(\lambda_{\max}(x), u_{\max}(x)) \in \mathbb{R} \times \mathbb{R}^n$ be the eigenpair associated with the maximum eigenvalue of (1). Let the following function*

$$g := R_{u_{\max}(a)} : \mathcal{H} \to \mathbb{R} : x \mapsto R(x, u_{\max}(a))$$

be Lipschitz continuous. Then, g is a Lipschitz continuous support of $\lambda_{\max}$ in a.

Proof. Definition 2 requires two properties to be satisfied independently of $a \in \mathcal{H}$:

$$\forall x \in \mathcal{H} : g(x) \leq \lambda_{\max}(x),$$
$$g(a) = \lambda_{\max}(a),$$

both of which are trivially satisfied by the Rayleigh-Ritz ratio as established in respectively Lemma 3 and Lemma 2. □

The theorem can be applied to MMEPs with affine parameters, since, following Lemma 7, R is Lipschitz continuous.

Corollary 14 *The maximal eigenvalue function $\lambda_{\max}$ of an MMEP pencil is a Lipschitz continuous function on $\mathcal{H}$, due to the existence of the Lipschitz continuous supports R_a for all $a \in \mathcal{H}$, cfr. Lemma 12. Hence we have that $\lambda_{\max} \in C^0(\mathcal{H})$, as mentioned earlier in the problem formulation.*

This leads us now to the reformulation of the subproblem (17):

Theorem 15 *Consider the sequence of points $(x^{(k)}, \eta^{(k)}) \in \mathcal{H} \times \mathbb{R}$, generated in the following way:*

$$(x^{(k)}, \eta^{(k)}) = \min_{x,\eta} \eta \tag{21}$$
$$st : \begin{cases} \eta \geq R_{u_{\max}(x^{(i)})}(x), & \forall i \in \mathcal{I}_{k-1}, \\ x \in \mathcal{H} \end{cases}$$

If R_a is Lipschitz continuous for all $a \in \mathcal{H}$, the sequence $\eta^{(i)}$ converges to $\lambda^*_{\max}$.

3.2 Diagonal minimal maximum eigenvalue problems

We show a connection between (21) and a diagonal MMEP, which we denote by DMMEP. Since the eigenvalues of a diagonal matrix are readily available, a function evaluation for an optimization method is cheap.

Define the following diagonal MMEP, where the $u^{(i)} := u_{\max}(x^{(i)}), i = 1, \ldots, k$:

$$\begin{aligned} A_D &= \operatorname{diag}(\{u^{(i)T} A(x) u^{(i)}\}_{i=1,\ldots,k} \\ B_D &= \operatorname{diag}(\{u^{(i)T} B(x) u^{(i)}\}_{i=1,\ldots,k} \ . \end{aligned}$$

Corollary 16 *The DMMEP (diagonal MMEP) formulation is a* non-smooth *problem, see* (5) *with* $A = A_D$ *and* $B = B_D$*, the global minimizer* $\lambda^*_{\max}$ *but it can be expressed as the solution of the* smooth *constrained optimization problem:*

$$\begin{aligned} \lambda^*_{\max} = \min_{x,\eta} \eta \\ s.t. : \begin{cases} \eta \geq R_{e_j}(x) & , \forall j \in \mathcal{I}_n \\ x \in \mathcal{H} \end{cases} \end{aligned} \tag{22}$$

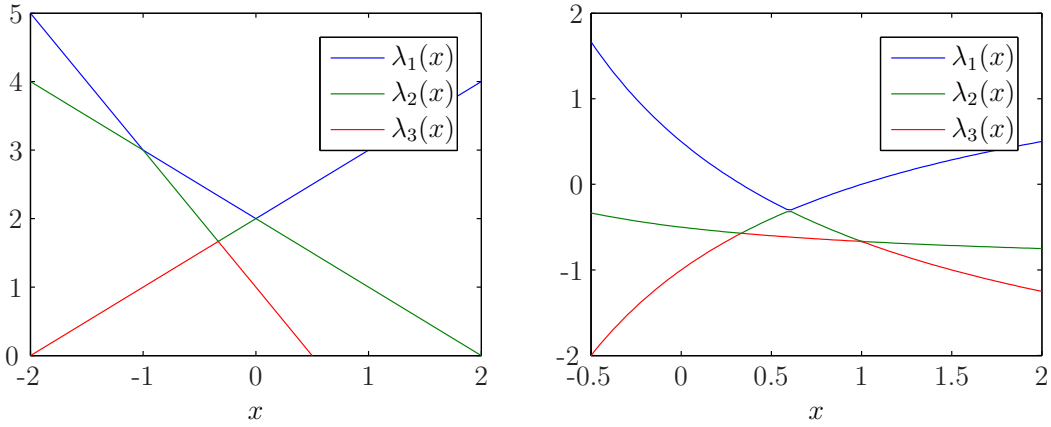
The choice of solution method for solving (21) depends on the properties of the problem. We give a few examples. When A and B are affine in x, the Rayleigh-Ritz ratios are linear fractional functions. The DMMEP is a smooth nonlinear constraint optimization problem. Suitable methods are interior point methods, SQP methods, active set methods, e.g. See [16] for an overview. All these methods are *local* optimization methods and require that the goal and constraint functions are *differentiable* for reasons of convergence. In the case of eigenvalue optimization the objective function is not guaranteed to be differentiable. The subproblem (21) satisfies these conditions.

See Figure 2 for two examples from MMEPs with affine parameters. When $B \equiv B_0$ (constant B), the Rayleigh-Ritz ratios are linear. In this case, linear programming or semi-definite programming techniques can be used []. When B does depend on x, the Rayleigh quotients are nonlinear and quasi convex.

4 A subspace method for the minimal maximum eigenvalue problem

In the previous section, we proposed an MMEP solver that generates a sequence of vectors, $x^{(k)} \in \mathcal{H}$, and corresponding eigenpairs $(\lambda_{\max}(x^{(k)}), u_{\max}(x^{(k)}))$. The eigenvectors $u_{\max}(x^{(k)})$ were used to define Lipschitz continuous supports in the form of Rayleigh-Ritz ratios.

In this section, we will show that a) by projecting the MMEP pencil onto the subspace generated by a set of eigenvectors, we construct a reduced MMEP pencil whose largest eigenvalue is a Lipschitz continuous support with respect to the original MMEP pencil and b) the sequence of maximal Ritz values obtained by solving a sequence of small scale reduced MMEPs converges to the solution of the original large MMEP.



(a) $A_0 = \mathrm{diag}(2,2,1)$, $A_1 = \mathrm{diag}(1,-1,-2)$, $B = I$

(b) $A_0 = \mathrm{diag}(1,-1,-2)$, $A_1 = \mathrm{diag}(-3,-1,2)$, $B_0 = 2I$ and $B_1 = I$

Figure 2: Illustration of the eigenvalues of two mono-variate 3×3 DMMEPs, illustrating the non-smooth but unimodal character of $\lambda_{\max}$. Note that (a) corresponds to a linear program, while (b) corresponds to a quasi-convex problem.

We expect advantages of this approach to be multi-fold: the only large scale operation to perform is computing the largest eigenvalue of a large sparse matrix pencil, most of the optimization effort is performed on small scale reduced problems which yields much cheaper function evaluations, and the generated supports are better approximations of $\lambda_{\max}$ than the supports from the DMMEP which results in faster convergence.

Let $V \in \mathbb{R}^{n\times k}$ be a full rank matrix and assume $k \ll n$. Recall the MMEP pencil (1). Now define the MMEP pencil

$$(V^T A(x)V, V^T B(x)V) \in \mathbb{S}^k(\mathcal{H}) \times \mathbb{S}^k_{++}(\mathcal{H}).$$

Let θ_V be the largest eigenvalue of this pencil, i.e.

$$\theta_V : \mathcal{H} \to \mathbb{R} : x \mapsto \max_{z\neq 0} \frac{z^T(V^T A(x)V)z}{z^T(V^T B(x)V)z} \ . \tag{23}$$

We call θ_V the largest Ritz value of (1).

Lemma 17 *Given two full rank matrices V and $W \in \mathbb{R}^{n\times k}$ such that Range$(V) \subseteq$ Range(W), then*

$$\forall x \in \mathcal{H} : \theta_V(x) \leq \theta_W(x) \ .$$

Proof. The proof follows from the fact that

$$\theta_V(x) = \max_{y\in \mathrm{Range}(V), y\neq 0} \frac{y^T A(x)y}{y^T B(x)y} \ .$$

Since W spans a larger subspace, the maximum must be larger for W than for V.

$$\forall x \in \mathcal{H} : \theta_V(x) = \max_{\substack{y \neq 0 \\ y \in \text{Range}(V)}} \frac{y^T A(x) y}{y^T B(x) y} \quad \leq \quad \max_{\substack{y \neq 0 \\ y \in \text{Range}(W)}} \frac{y^T A(x) y}{y^T B(x) y} = \theta_W(x)$$

□

Theorem 18 *Let R_a be Lipschitz continuous for all $a \in \mathcal{H}$. Let $u_{\max}$ be an eigenvector (4), associated with the largest eigenvalue of (1). Given a full rank matrix $V \in \mathbb{R}^{n \times k}$ and $a \in \mathcal{H}$ such that $u_{\max}(a) \in$ Range(V), then θ_V is a Lipschitz continuous support of $\lambda_{\max}$ in a:*

$$\theta_V \in \mathcal{S}(\lambda_{\max}, a), \tag{24}$$

$$\forall b \in \mathcal{H} : |\theta_V(a) - \theta_V(b)| \leq K_V \|a - b\| \tag{25}$$

with K_V a finite Lipschitz constant.

Proof. The observation that θ_V is Lipschitz continuous on $\mathcal{H}$ is a direct result of Corallary 14. To prove that θ_V is a support in $a \in \mathcal{H}$, we again have to show the following two properties:

$$\forall x \in \mathcal{H} : \theta_V(x) \leq \lambda_{\max}(x),$$
$$\theta_V(x)|_{x=a} = \lambda_{\max}(a)$$

The first property follows from Lemma 17 because Range$(V) \subseteq \mathbb{R}^n$.

To prove the second property, recall (23), which states that:

$$\forall x \in \mathcal{H}, \forall y \in \mathbb{R}^r : \theta_V(x) \geq \frac{y^T V^T A(x) V y}{y^T V^T B(x) V y} \ . \tag{26}$$

Furthermore, if $u_{\max}(a) \in \text{Range}(V)$, then there exists a vector $c \in \mathbb{R}^r$ such that $u_{\max}(a) = Vc$. Substituting c into (26) then results in:

$$\theta_V(a) \geq \frac{c^T V^T A(a) V c}{c^T V^T B(a) V c} = \lambda_{\max}(a) \tag{27}$$

If $\lambda_{\max}(a)$ is both bigger and smaller than $\theta_V(a)$, (see Lemma 17 and (27), respectively), $\lambda_{\max}(a) = \theta_V(a)$, which proves the theorem. □

Theorem 19 *Given the sequence of points $(x^{(i)}, \theta^{(i)*}) \in \mathcal{H} \times \mathbb{R}$ for $i \in \mathcal{I}_k = 1, 2, \ldots, k$. Let each k'th tuple $(x^{(k)}, \theta^{(k)*})$ be defined as the solution of the reduced $(k-1) \times (k-1)$ MMEP:*

$$(x^{(k)}, \theta^{(k)*}) = \min_{x, \eta} \eta$$
$$s.t. : \begin{cases} \eta \geq \theta_{V^{(k-1)}}(x) \\ x \in \mathcal{H} \end{cases} \tag{28}$$

where $V^{(k-1)} \in \mathbb{R}^{n\times(k-1)}$ *is a basis for the space spanned by the maximal eigenvectors* (4)*:*

$$\mathit{Range}(V^{(k-1)}) = \mathit{span}\left(\left\{u_{\max}(x^{(1)}), u_{\max}(x^{(2)}), \cdots, u_{\max}(x^{(k-1)})\right\}\right). \tag{29}$$

The sequence $\theta^{(k)*}$ *of reduced MMEP solutions then converges to* $\lambda^*_{\max}$*, the solution of the original MMEP.*

Recall from Theorem 11 that each new point $(x^{(k)}, \theta^{(k)*})$ is related to *all* the previous points by including all the respective supports as constraints, whereas (28) includes only *one* constraint in the form of a reduced MMEP. However, from (29) and Theorem 18, it follows that

$$\forall k > 0, \forall i \in \mathcal{I}_k : \theta_{V^{(k)}} \in \mathcal{S}(\lambda_{\max}, x^{(i)}) \ . \tag{30}$$

This implies that $\theta_{V^{(k)}}$ is a support of $\lambda_{\max}$ in $\{x^{(1)}, x^{(2)}, \cdots, x^{(k)}\}$. Since

$$\forall k, m > 0, \text{ with } k < m \text{ and thus } \mathrm{Range}(V^{(k)}) \subseteq \mathrm{Range}(V^{(m)}) \ , \tag{31}$$

we can deduce, as for Lemma 17

$$\forall k, m > 0, k < m \text{ and } \forall x \in \mathcal{H} : \theta_{V^{(k)}}(x) \leq \theta_{V^{(m)}}(x) \ . \tag{32}$$

Therefore, there is no reason to include *all* the previously generated reduced MMEP pencils but only the last one.

Comparing Theorems 15 and 19, we see that the first generates the supports $R_{u_{\max}(x^{(k)})}$ by projecting the MMEP pencil on the individual maximal eigenvectors $u_{\max}(x^{(k)})$, while the latter aggregates these eigenvectors $u(x^{(k)})$ to generate a reduced MMEP pencil. From (29) it follows that

$$\forall k > 0, \forall i \in \mathcal{I}_k : u_{\max}(x^{(i)}) \in \mathrm{Range}(V^{(k)}) \ , \tag{33}$$

hence

$$\forall k > 0, \forall i \in \mathcal{I}_k, \forall x \in \mathcal{H} : \theta_{V^{(k)}}(x) \geq R_{u_{\max}(x^{(i)})}(x) \ . \tag{34}$$

This means that the sequence $\theta^{(k)*}$ generated by Theorem 19 converges at least as fast as the sequence $\eta^{(k)}$ generated by Theorem 15.

The following algorithm gives a description of a precedure using expansion, reduction and extraction

Algorithm 1 (RMMEP)

Input: *A, B: symmetric generalized eigenvalue problem*
$x \in \mathcal{H}$: constraint on x,
$B(x)$ is positive definite on $\mathcal{H}$
ϵ: absolute tolerance
$\lambda_{\max}(A, B)$: routine for computing a $(\lambda_{\max}, u)$ pair of the generalized eigenvalue problem
Output: $\lambda^*_{\max}$
1: $x \leftarrow$ *initial value*

2: ***for*** $k = 1, 2, \cdots$ ***do***
3: ***expansion****: compute* $(\lambda, u) = \lambda_{\max}(A(x), B(x))$
4: *Expand* V *with* u *and orthogonalize.*
5: ***reduction****: compute the reduced MMEP pencil matrices*
6: ***extraction****: compute the solution* (x, θ^*) *of the* $k \times k$ *reduced MMEP pencil* (23).
7: ***if*** $\lambda_{\max}(x) - \theta^* \leq \epsilon$ ***then***
8: ***return*** $\lambda_{\max}(x)$
9: ***end if***
10: ***end for***

A stopping criterion is easily derived from the following observation. The Ritz values are always lower bounds to the largest eigenvalue for any x. So, the minimal largest Ritz value is a lower bound to the minimal largest eigenvalue. Second, the value of $\lambda_{\max}(x)$ is an upper bound to the minimum. Hence, we have

$$\theta^{(k)*} \leq \lambda^\star \leq \lambda_{\max}(x^{(k)}) \ ,$$

An error estimation is thus given by

$$\lambda_{\max}(x^{(k)}) - \theta^{(k)*} \ .$$

Example 2 *Figure 3 shows the state of RMMEP after one, two and three iterations. They show* $\lambda_{\max}$ *of an example of a mono-variate MMEP pencil and the generated* $\theta_{V^{(k)}}$ *supports of respectively a* 1×1*,* 2×2 *and* 3×3 *reduced MMEP pencil.*

The dots represent actual evaluations of $\lambda_{\max}$ *performed by the routine, which is exactly one per iteration. The diamonds represent the respective solutions* $\theta^{(k)*}$ *of reduced MMEP.*

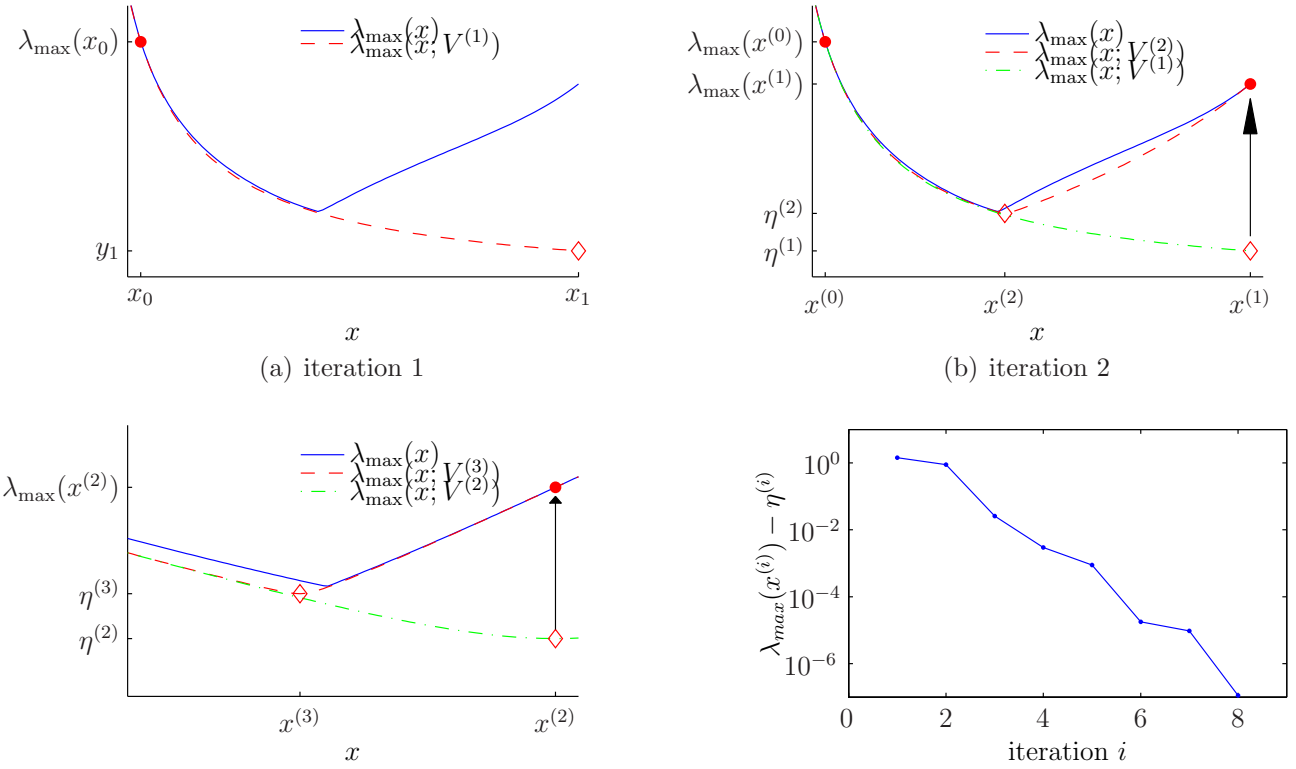
5 Numerical examples

In this section, we present numerical examples to illustrate the convergence theory of the previous sections. First, we solve a one-parameter ($p = 1$) civil engineering problem and then a multivariate ($p > 1$) problem of large size. All runs were performed using Matlab R2012a on a machine with two Intel Xeon E5645 hexa-core processors running at 2.40GHz and 48GB of main memory.

5.1 Implementation issues for univariate problems

For univariate problems, the reference is a level-set method [6]. This is an iterative method that typically converges very fast, quadratically or faster, depending on the version. The simplest level-set method is the bisection method. The method works as follows.

1: Let the starting value be y^*.
2: **while** not converged **do**



(a) iteration 1

(b) iteration 2

(c) iteration 3: Close-up

(d) Absolute error in function of the iteration number

Figure 3: Progress of the subspace projection method after respectively one,two and three iterations and the absolute error after each iteration.

3: Let λ be the largest eigenvalue of (1) with $x = y^*$.
4: Find all values of y for which λ is the largest eigenvalue of $A(y)u = \lambda B(y)u$ (If the problem is uni-modal, there are at most two such values: $y^{(1)}$ and $y^{(2)}$.)
5: Let y^* be the mean of these two values: $y^* = 0.5(y^{(1)} + y^{(2)})$.
6: **end while**

This algorithm requires the computation of the largest eigenvalue of (1) on each iteration. In addition, it requires the computation of the values $y^{(1)}$ and $y^{(2)}$. One of those values is y^*, so only the other value needs to be computed. This can again be written as the solution of the eigenvalue problem (1) where the role of λ and x is interchanged: λ is fixed and x is the unknown eigenvalue. In other words, we need to solve two eigenvalue problems for each iteration. When A and B do not depend on x in an affine way, the implementation of the level-set method becomes more complicated, since the corresponding eigenvalue problem is nonlinear and the level may consist of a number of undetermined points, which may be larger than two.

For a large scale eigenvalue problem, the largest eigenvalue is computed by the Lanczos method. For computing the smallest eigenvalue of large scale engineering applications, we

can use the spectral transformation Lanczos method with zero shift. The solution of the eigenvalue problem to compute the levelset is more complicated. In the example from §5.3, this boils down to a polynomial eigenvalue problem of degree three. All real eigenvalues of this problem need to be computed. This is, in general, not such a simple problem to solve when matrices are large and sparse.

The reduction method is in this respect appealing. Of course, the Lanczos method is still used for computing the largest (or smallest) eigenvalue for a given x, but the level-set problem is only used for solving the reduced problem. Since this problem is small, we can use the QZ method for both eigenvalue problems at each iteration of the level-set method. In addition, the iterations of the level-set method can be reduced by choosing $y = x^{(k)}$ as the starting value. In particular, in the last iterations, where $x^{(k)}$ is already close to the optimal solution, only a few iterations (maybe one or two) of the level-set method are required.

5.2 Implementation issues for multivariate problems

For multivariate problems, the situation is more complicated. When A and B are affine, we can use, e.g., the interior point method [9] for the reduced problem. For large scale eigenvalue problems, this method is not feasible. There exist methods that are based on the levelset concept in the multivariate case [3]. The DMMEP is a smooth problem and we can use, e.g., active set methods.

For the reduction method, the solution of the kth subproblem can be used as a starting value for the solution of subproblem $k + 1$. For example, when a quasi Newton method can be used, $x^{(k)}$ is likely to be a good starting guess for the solution of the $k + 1$st subproblem. When the reduced problem is solved using the DMMEP method, the constraints of subproblem k are also used as initial constraints for subproblem $k + 1$.

5.3 Civil engineering application

The first problem concerns the process floor of a nano-electronics cleanroom laboratory that requires high demands on vibrations. The process floor measures 75m by 52m, and can be subdivided into 6 modules of about 26m by 25m, which are each supported by 25 columns. Each module encompasses 16 spans measuring about $6 \times 6\text{m}^2$ (Figure 4(a)) [15]. A 3D beam element model of a quarter-module is constructed using the finite element MATLAB toolbox StaBiL [7]. Symmetry boundary conditions on the edges of the floor and a fully clamped column base lead to a finite element model of the structure which consists of 30 beam elements and $n = 57$ degrees of freedom. The only unknown of the problem is the height h of the beams that needs to be chosen so that the first eigenvalue is maximized in order to reduce low frequency vibrations in the structure. (This is mathematically equivalent to minimizing the largest eigenvalues for the problem where A is replaced by $-A$.) This is a multimodal problem, but we have chosen the interval $\mathcal{H}$ so that the problem has a single local minimum. The interesting feature of this model is that the parameter dependency is not affine, and so, the function is not necessarily quasi convex.

In Figure 5(a), we show the smallest eigenvalue of a reduced problem of dimension two. Clearly, the support for the problem of order two is not unimodal. This is acceptable since A and B do not have affine dependency on the parameters, so that quasi-convexity is no longer guaranteed. The theory of §3 remains valid, since the eigenvalue function is unimodal in $\mathcal{H}$. The level-set method now may have to deal with multimodal functions, which makes its implementation slightly more difficult, in particular, for large scale problems.

Levelsets are determined using the Matlab function `polyeig`. The method required 0.173 seconds, where the reduction method with levelset for the reduced problem required 0.093 seconds. As mentioned before, one iteration of the levelset method requires the solution of two eigenvalue problems. For eigenvalue problems of larger dimensions, the timings would be even more in favour of the reduction method, since the computation of the levelset would only be required for a small reduced problem.

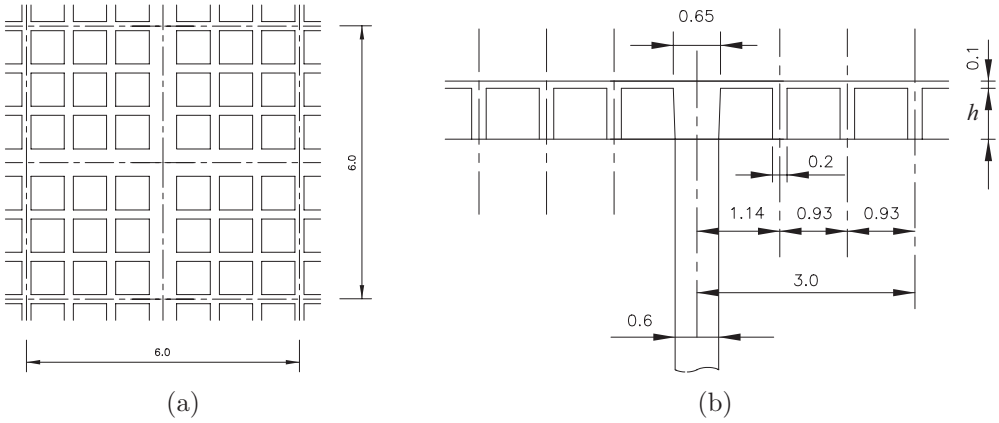


Figure 4: Concrete floor supported by parametrized beams

We now discuss the convergence history of the methods used to solve this problem; see Figure 5(b). The levelset method is our reference. It converges fast, but the reduction method is faster. In fact, the reduction method builds a reduced model that is quickly pretty close to the exact eigenvalue function. The DMMEP method, which is a smooth optimization problem, is clearly the slowest method.

5.4 Laplacian

We generated the matrices A and B as follows. Matrix A is minus the 3D discrete Laplacian. It is constructed as

$$A = L_1 \otimes I \otimes I + I \otimes L_1 \otimes I + I \otimes I \otimes L_1$$

where L_1 is the tridiagonal matrix with -2 on the main diagonal and 1 on the upper and lower diagonals. The matrix is of order $n = 27,000$. Matrix B is a diagonal matrix of the form (3) with parameter dependent diagonal entries, where $B_0 = I$ and the main diagonal of B_k is determined by the following Matlab-code:

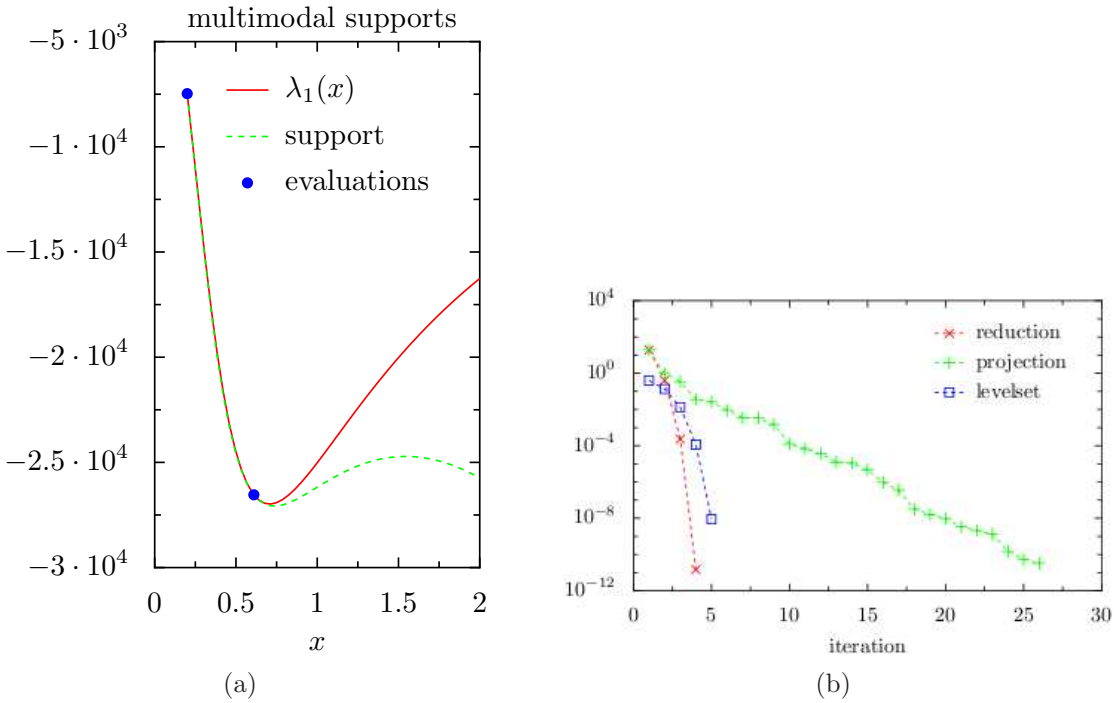


Figure 5: Multimodal support of reduced problem from two interpolation points and convergence history of numerical methods

```
steps = [1,2,3,4,5,6,12,15,9,10] ;
m_diag = ones(n,1) ;
indices = [] ;
offset = floor(n/(2*steps(k))) ;
for l=1:steps(k)
  indices = [indices, offset*(2*l-1)+(1:offset)] ;
end
m_diag( indices ) = -1 ;
B_k = spdiags( 0.3*m_diag, 0, n, n) ;
```

Matrix B is chosen so that the optimum is reached for $x = 0$ and so that $\nabla\lambda_{\max}$ exists and vanishes for $x = 0$. Those matrices could be interpreted as parameter dependent densities for the heat equation or the Helmholtz equation discretized by finite differences. The largest eigenvalue is quasi convex, since the parameter dependency is affine. We aim at maximizing the largest eigenvalue, which is, at the same time, the eigenvalue nearest zero. We therefore use, for a given x, the matlab code `eigs` [13]:

```
[z,e] = eigs( A(x), B(x), 1, 0 ) ;
```

We chose $\mathcal{H} = [0, 1]^p$.

We have used the following methods.

Method	$p = 2$ Time (s)	Evaluations of $\lambda_{\max}(x)$	% eigs	$p = 5$ Time (s)	Evaluations of $\lambda_{\max}(x)$	% eigs
TR	233.67	16	—	682.20	75	—
RTR	42.33	7	76	115.16	13	85
DMMEP	207.82	32	97	590.55	76	95
RDMMEP	78.61	7	69	184.34	13	62
Method	$p = 10$ Time (s)	Evaluations of $\lambda_{\max}(x)$	% eigs			
TR	774.16	80	—			
RTR	175.17	19	80			
DMMEP	1299.71	165	94			
RDMMEP	266.79	15	56			

Table 1: Comparison of the different methods for $p = 2$ and $p = 5$

TR The reference code is the trust region method from Matlab's `fmincon`. Note that this method is a gradient method and is not guaranteed to work for non-smooth problems, but it performs well on our example, as expected, since it is a smooth problem.

RTR The second method is the reduction method using the trust region method from `fmincon` to solve the reduced problem.

DMMEP The third method is the DMMEP method. We used the active set method from `fmincon` to solve the subproblem.

RDMMEP The last method is the reduction method using DMMEP for solving the subproblem.

Table 1 shows, for the different methods, the total execution time (computed using Matlab's `tic/toc`), the number of large scale eigenvalue problems solved, using `eigs`, the percentage of the time spent in `eigs` for $p = 2$, $p = 5$ and $p = 10$ parameters. We conclude that most of the time is indeed spent in the large scale eigenvalue solver. However, the time spent in the solution of the reduced problems is significant.

Table 2 shows details on the iterations for the RTR and RDMMEP methods for solving the five parameter problem. The solution of the reduced (sub)problem becomes cheaper for the last iterations. The reason for RTR is that a good starting value for the solution of the $k + 1$st subproblem is available, namely $x^{(k)}$. Indeed, in the last iteration, only one iteration of the subproblem is required for a sufficiently accurate solution. The reduction of iterations for the RDMMEP method only takes place in the last two iterations. Only a few iterations are required for solving the subproblem in the first iterations.

RTR	37 1	19 1	34	22	17	10	7	3	1	1
RDMMEP	2 1	3 1	2	2	5	2	2	2	54	2

Table 2: Number of iterations for the solution of the kth subproblem for the RTR and RDMMEP methods, when $p = 5$.

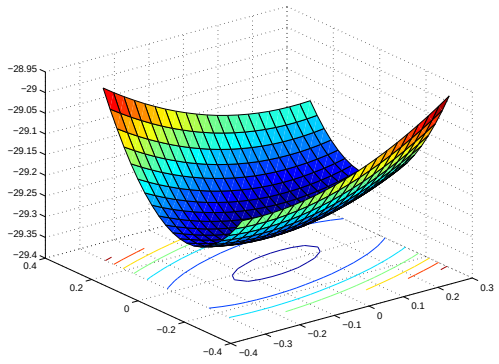


Figure 6: Eigenvalue function for $p = 2$.

6 Conclusions

We presented two methods for solving uni-modal eigenvalue optimization problems and proved their convergence. The first is a reduction or subspace method which reduces the size of the eigenvalue problem, where the second translates the eigenvalue optimization problem into a smooth optimization problem (DMMEP).

For univariate problems, our numerical experiments show iteration counts of the same order for the reduction method and a levelset method. However, for large scale eigenvalue problems, a levelset method is harder to implement and an iteration is more expensive than for the reduction method.

For the multivariate problems, we showed speedup for a variety of numerical methods. We experienced that the proposed method based on the DMMEP converges slowly. Significant speed-ups can be obtained in combination with reduction. However, the time spent in the solution of the reduced problem is significant, which illustrates the importance of good optimization methods for the reduced problem.

Acknowledgements

The authors are grateful to Geert Lombaert and Ellen Simoen for providing the example of the concrete floor of a nonotechnology laboratory [15]. We also thank Emre Mengi for reading a draft version of the paper.

This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State Science Policy Office. The research is also partially funded by the Research Council KU Leuven grants PFV/10/002 (Optimization in Engineering) and OT/10/038 (Multi-parameter model order reduction and its applications). The scientific responsibility rests with its author(s).

References

[1] W. Achtziger, M. Kočvara, On the maximization of the fundamental eigenvalue in topology optimization, Struct Multidisc Optim 34 (2007) 181–195.

[2] S. Adhikari, M. Friswell, Random matrix eigenvalue problems in structural dynamics, International Journal of Numerical Methods in Engineering 69 (2007) 562–591.

[3] S. Boyd, L. Ghaoui, Method of centers for minimizing generalized eigenvalues, Linear Alg. Appl. 188–189 (0) (1993) 63–111.

[4] S. Boyd, L. Vandenberghe, Convex Optimization, vol. 25, Cambridge University Press, 2004.

[5] J. Burke, A. Lewis, M. Overton, A robust gradient sampling algorithm for nonsmooth, nonconvex optimization, SIAM Journal on Optimization 15 (3) (2005) 751–779.

[6] R. Byers, A bisection method for measuring the distance of a stable matrix to the unstable matrices, SIAM Journal on Scientific and Statistical Computing 9 (5) (1988) 875–881.

[7] D. Dooms, G. De Roeck, G. Degrande, S. François, Een interactieve en adaptieve toepassing voor de statische en dynamische analyse van structuren (StaBIL), Technical Report BWM-2009-21, Department of Civil Engineering, KU Leuven (2009).

[8] R. G. Grimes, J. G. Lewis, H. D. Simon, A shifted block Lanczos algorithm for solving sparse symmetric generalized eigenproblems, SIAM Journal on Matrix Analysis and Applications 15 (1994) 228–272.

[9] J.-P. Haeverly, M. Overton, Optimizing eigenvalues of symmetric definite pencils, in: American Control Conference, vol. 1, IEEE, 1994, pp. 836–839.

[10] C. Helmberg, F. Rendl, A spectral bundle method for semidefinite programming, SIAM Journal on Optimization 10 (3) (2000) 673–696.

[11] R. Horn, C. Johnson, Topics in matrix analysis, Cambridge University Press, 1991.

[12] J. E. Kelley, Jr., The Cutting-Plane Method for Solving Convex Programs, Journal of the Society for Industrial and Applied Mathematics 8 (4) (1960) 703.

[13] R. B. Lehoucq, D. C. Sorensen, C. Yang, ARPACK Users Guide: Solution of Large Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods, SIAM, Philadelphia, PA, USA, 1998.

[14] H. Lin, An inexact spectral bundle method for convex quadratic semidefinite programming, Computational Optimization and Applications (2011) 1–45.

[15] G. Lombaert, E. Simoen, The process floor of the nano-electronics laboratory at IMEC, private communication (2012).

[16] J. Nocedal, S. Wright, Numerical optimization, Operations Research, Springer, 1999.

[17] B. N. Parlett, The Symmetric Eigenvalue Problem, Prentice Hall series in computational mathematics, Prentice-Hall, 1980.

[18] G. Sleijpen, H. van der Vorst, A Jacobi-Davidson iteration method for linear eigenvalue problems, SIAM Journal on Matrix Analysis and Applications 17 (1996) 401–425.

[19] L. Vandenberghe, S. Boyd, Semidefinite programming, SIAM Review 38 (1) (1996) 49–95.

[20] J. Wilkinson, The algebraic eigenvalue problem, Clarendon Press, Oxford, UK, 1965.