Large-Scale Minimization of the Pseudospectral Abscissa

Nicat Aliyev* Emre Mengi†

August 17, 2022

Abstract

This work concerns the minimization of the pseudospectral abscissa of a matrix-valued function dependent on parameters analytically. The problem is motivated by robust stability and transient behavior considerations for a linear system that has optimization parameters. We describe a subspace procedure to cope with the setting when the matrix-valued function is of large size. The proposed subspace procedure solves a sequence of reduced problems obtained by restricting the matrix-valued function to small subspaces, whose dimensions increase gradually. It possesses desirable features such as the global convergence of the minimal values of the reduced problem to the minimal value of the original problem, and a superlinear convergence exhibited by the decay in the errors of the minimizers of the reduced problems, which we prove in theory under mild assumptions. In mathematical terms, the problem we consider is a large-scale nonconvex minimax eigenvalue optimization problem such that the eigenvalue function appears in the constraint of the inner maximization problem. What makes it peculiar as compared to our previous works that concern the maximization of the distance to instability, and minimization of the $\mathcal{H}_\infty$ norm involving also nonconvex minimax eigenvalue optimization problems is that in those works the eigenvalue functions appear in the objective. Devising and analyzing a subspace framework for the minimax eigenvalue optimization problem at hand with the eigenvalue function in the constraint require special treatment that makes use of a Lagrangian and dual variables. There are notable advantages in minimizing the pseudospectral abscissa over maximizing the distance to instability or minimizing the $\mathcal{H}_\infty$ norm; the optimized pseudospectral abscissa provide quantitative information about the worst-case transient behavior, and the initial guesses for the parameter values to optimize the pseudospectral abscissa can be arbitrary, unlike the case to optimize the distance to instability and $\mathcal{H}_\infty$ norm that would normally require initial guesses yielding asymptotically stable systems.

*Czech Technical University, Department of Instrumentation and Control, Technicka 4, 16607, Prague, Czech Republic, E-Mail: nijat.aliyev@cvut.cz.

†Koç University, Department of Mathematics, Rumeli Feneri Yolu 34450, Sarıyer, Istanbul, Turkey, E-Mail: emengi@ku.edu.tr.

Keywords. pseudospectral abscissa, large scale, subspace framework, Hermite interpolation, Lagrangian, robust stability, eigenvalue optimization, nonconvex optimization

AMS Subject Classifications. 65F15, 93C05, 93D09, 90C26, 90C47

1 Introduction

The minimization of the spectral abscissa of a linear control system has drawn interest in the last couple of decades [8, 3]. A classical problem that can be tackled using the spectral abscissa minimization is the stabilization by static output feedback (SOF) problem; given matrices $A \in \mathbb{C}^{n\times n}$, $B \in \mathbb{C}^{n\times m}$, $C \in \mathbb{C}^{p\times n}$, find a controller $K \in \mathbb{C}^{m\times p}$ such that the system $x'(t) = (A + BKC)x(t)$ is asymptotically stable, equivalently $A + BKC$ has all of its eigenvalues on the open left-half of the complex plane. SOF is known to be a notoriously difficult problem [4]. Indeed, it has been shown that SOF when the entries of K are subject to box constraints is NP-hard [5, 23]. Mathematically, the spectral abscissa minimization is a nonconvex eigenvalue optimization problem that involves the minimization of the real part of the rightmost eigenvalue. Nonsmoothness at a locally optimal point minimizing the spectral abscissa can occur due to possible nonsimplicity of the rightmost eigenvalue at the local optimizer [6, 7]. Numerically speaking a bigger challenge is the non-Lipschitz nature of the rightmost eigenvalue at a local optimizer; when the rightmost eigenvalue is not simple at a local optimizer, it can change rapidly near the optimizer. In more formal terms, the spectral abscissa does not have to be Lipschitz continuous, not even locally, at a local optimizer, and this causes numerical difficulties to numerical algorithms, which are at least prone to rounding errors. A second difficulty with minimizing the spectral abscissa is that, even if a negative spectral abscissa guarantees an asymptotic decay, the system can still exhibit significant transient behavior before the eventual decay.

As a remedy to these problems with the spectral abscissa minimization, the pseudospectral abscissa minimization has been considered in the last two decades [9, 2]. Recall that the spectral abscissa of a matrix $A \in \mathbb{C}^{n\times n}$ is given by $\alpha(A) := \max\{\mathrm{Re}(z) \mid z \in \Lambda(A)\}$, where $\Lambda(\cdot)$ denotes the spectrum, i.e., the set of all eigenvalues, of its matrix argument. On the other hand, for a given real number $\epsilon > 0$, the *ϵ-pseudospectrum* of A consists of eigenvalues of all matrices within an ϵ-neighborhood of A, formally defined as

$$\Lambda_\epsilon(A) \;=\; \left\{z \in \mathbb{C} \mid z \in \Lambda(A+\Delta) \;\; \exists \Delta \in \mathbb{C}^{n\times n} \;\; \text{s.t.} \;\; \|\Delta\|_2 \le \epsilon\right\},$$

and the *ϵ-pseudospectral abscissa* of A is defined as $\alpha_\epsilon(A) := \max\{\mathrm{Re}(z) \mid z \in \Lambda_\epsilon(A)\}$, that is as the real part of the rightmost point in $\Lambda_\epsilon(A)$ [24, 25].

Unlike the spectral abscissa, the pseudospectral map $A \mapsto \alpha_\epsilon(A)$ is locally Lipschitz continuous [20, Corollary 7.2], [14, Corollary 3.4]. Even if nonsmoothness and nonconvexity are still present in the map $A \mapsto \alpha_\epsilon(A)$, changes as rapid as in the spectral abscissa map are not possible. A second virtue in using the pseudospectral abscissa rather than the spectral abscissa is that, if it is sufficiently small, not only the asymptotic decay but also a nice transient behavior is guaranteed, thanks to the Kreiss-matrix theorem [25, 16]. Formally,

$$\sup_{\epsilon>0} \frac{\alpha_\epsilon(A)}{\epsilon} \;\leq\; \sup_t \|e^{At}\|_2 \;\leq\; en \sup_{\epsilon>0} \frac{\alpha_\epsilon(A)}{\epsilon},$$

which implies the following for the system $x'(t) = Ax(t)$: we have $\sup_{\epsilon>0} \alpha_\epsilon(A)/\epsilon \leq \|x(t)\|_2 \leq en \sup_{\epsilon>0} \alpha_\epsilon(A)/\epsilon$ at all times $t \geq 0$. Equivalent and computationally plausible characterizations for $\Lambda_\epsilon(A)$ and $\alpha_\epsilon(A)$ are given by [25]

$$\begin{aligned} \Lambda_\epsilon(A) &= \{z \in \mathbb{C} \mid \sigma_{\min}(A - zI) \leq \epsilon\}, \\ \alpha_\epsilon(A) &= \max\{\mathrm{Re}(z) \mid \sigma_{\min}(A - zI) \leq \epsilon\}, \end{aligned} \tag{1.1}$$

where $\sigma_{\min}(\cdot)$ denotes the smallest singular value of its matrix argument.

The computation of the ϵ-pseudospectral abscissa and its derivatives require more work compared to that for the spectral abscissa. Whereas the spectral abscissa and derivatives can be obtained from eigenvalue computations merely, and even for large sparse matrices there is powerful software to retrieve rightmost eigenvalues and derivatives such as ARPACK [18], the computation of the pseudospectral abscissa involves an optimization problem. Yet, there are very good algorithms to compute the ϵ-pseudospectral abscissa. The criss-cross algorithm developed by Burke, Lewis and Overton is globally convergent at a quadratic rate, hence computes the pseudospectral abscissa very reliably and efficiently for small- to medium-size matrices [11, 10]. For the pseudospectral abscissa of larger-size matrices, there is a fixed-point iteration developed by Guglielmi et al. [13], which can further be accelerated with the subspace framework in [17].

1.1 Problem and Contributions

In this work, we assume that the matrix A depends on parameters in a smooth way, and, for a given $\epsilon > 0$, deal with the minimization of the ϵ-pseudospectral abscissa of A over the parametsr. Our focus is on the case when A is large; we describe a subspace procedure that reduces the size of A considerably but without altering the optimal parameter values. We prove in theory that the subspace procedure possesses desirable convergence properties such as global convergence and a superlinear rate of convergence. As a result, it enables us to solve the ϵ-pseudospectral abscissa

minimization problems involving parameter dependent matrices with sizes on the order of thousands.

Formally, we assume we are given a matrix-valued function $A : \Omega \to \mathbb{C}^{n\times n}$ of the form

$$A(x) = f_1(x)A_1 + \cdots + f_\kappa(x)A_\kappa, \tag{1.2}$$

where $A_1, \ldots, A_\kappa \in \mathbb{C}^{n\times n}$, the size of the matrices n is very large, and $f_1, \ldots, f_\kappa : \Omega \to \mathbb{R}$ are real-analytic on Ω, an open subset of $\mathbb{R}^d$ representing the permissible values for the parameters. The ϵ-pseudospectral abscissa minimization problem, for a prescribed $\epsilon > 0$, is the minimax problem

$$\begin{aligned} \min_{x\in\underline{\Omega}} \alpha_\epsilon(A(x)) &= \min_{x\in\underline{\Omega}} \max\{\mathrm{Re}(z) \mid z \in \Lambda_\epsilon(A(x))\} \\ &= \min_{x\in\underline{\Omega}} \max\{\mathrm{Re}(z) \mid \sigma_{\min}(A(x) - zI) \le \epsilon\} \end{aligned} \tag{1.3}$$

where $\underline{\Omega}$ is a compact subset of Ω.

The stabilization by static output feedback problem for given $A \in \mathbb{C}^{n\times n}$ $B \in \mathbb{C}^{n\times m}$, $C \in \mathbb{C}^{p\times n}$ can be treated by minimizing $\alpha_\epsilon(A + BKC)$ over $K \in \mathbb{C}^{m\times p}$ with entries constrained to lie in prescribed intervals. When n is large, such a stabilization problem falls into the setting of (1.3) as $A + BKC$ can represented in the form (1.2).

The problem at hand is a nonconvex large-scale minimax eigenvalue optimization problem. What makes it peculiar compared to our previous works [21, 1] is that the eigenvalue function appears in the constraint of the inner maximization problem. This is in contrast to [21] and [1] that introduce subspace frameworks for large-scale minimax eigenvalue optimization problems - specifically for the maximization of the distance to instability and minimization of the $\mathcal{H}_\infty$-norm, respectively - where the eigenvalue functions appear in the objective. Designing a subspace framework for a minimax problem with the eigenvalue function in the objective, and especially analyzing its convergence require a special treatment. For instance, when establishing the global convergence of the proposed framework and its superlinear convergence, we work on the Lagrangian as well as the dual variable as much as the primal variables.

Minimizing the pseudospectral abscissa and maximizing the distance to instability (more generally minimizing the $\mathcal{H}_\infty$-norm) are motivated by similar issues, most remarkably by the maximization of the robust stability and optimization of the transient behavior. However, there are advantages in optimizing the pseudospectral abscissa. First, by minimizing the pseudospectral abscissa we simultaneously minimize a lower bound on the worst transient behavior. Secondly, when maximizing the distance to instability or minimizing the $\mathcal{H}_\infty$ norm, the system at the initial guess for the parameter values should ideally be asymptotically stable, and finding such a guess may be a challenge. There is no such asymptotic stability concerns when minimizing the

pseudospectral abscissa; it does not harm in any way to start with parameter values that yield systems that are not asymptotically stable.

1.2 Outline

We introduce the subspace framework in the next section, then investigate its properties such as the derivatives of the original and reduced problems, and, consequently, deduce Hermite interpolation properties between the original and reduced problems. The proposed framework produces a sequence of reduced problems with sizes increasing gradually. In Section 3, we formally show that if, in finite dimension, the minimizers of the reduced problems stagnate, then the point of stagnation is actually a global minimizer of the original problem, and, in infinite dimension, the globally minimal values of the reduced problems in the limit converge to a globally minimal value of the original problem. Section 4 is devoted to a rate-of-convergence analysis of the proposed framework. In this section, under mild assumptions, we prove a superlinear convergence result for the errors of the reduced problems. A Matlab implementation of the proposed framework is made publicly available. In Section 5, we perform numerical experiments with this implementation on synthetic as well as benchmark examples from the $COMPl_eib$ collection [19], and observe that the deduced theoretical global convergence and superlinear convergence results hold in practice. The numerical results illustrate the efficiency and accuracy of the subspace framework in practice on large-scale pseudospectral abscissa minimization problems.

2 Subspace Framework

In this section, we present a subspace framework for the minimization of the pseudospectral abscissa of a large-scale parameter dependent matrix $A(x)$ of the form (1.2). The subspace idea introduced here is inspired by [15, 21]. However, the problem considered in [15] concerns the minimization or maximization of the jth largest eigenvalue of the given matrix-valued function for a prescribed j, whereas the problem in [21] concerns a minimax problem with an eigenvalue function in the objective. The problem here is also a minimax problem, but the eigenvalue (or the singular value) function appears in the constraint of the inner maximization problem, and, as a result, the problem at hand requires a special treatment.

We resort to one-sided projections to deal with the large size of $A(x)$. Specifically, given a subspace $\mathcal{V} \subseteq \mathbb{C}^n$ of dimension $k \ll n$, and a matrix $V \in \mathbb{C}^{n\times k}$ whose columns form an orthonormal basis for $\mathcal{V}$, we minimize the pseudospectral abscissa of the reduced matrix-valued function

$$A^V(x) := A(x)V = f_1(x)A_1V + \cdots + f_\kappa(x)A_\kappa V$$

instead of minimizing the pseudospectral abscissa of $A(x)$. Formally, defining the ϵ-pseudospectrum of the reduced matrix-valued function by

$$\Lambda_\epsilon(A^V(x)) := \left\{ z \in \mathbb{C} \;\middle|\; \sigma_{\min}(A^V(x) - zV) \leq \epsilon \right\}$$

and the corresponding ϵ-pseudospectral abscissa by

$$\begin{aligned} \alpha_\epsilon(A^V(x)) &:= \max\left\{ \mathrm{Re}(z) \;\middle|\; z \in \Lambda_\epsilon(A^V(x)) \right\} \\ &= \max\left\{ \mathrm{Re}(z) \;\middle|\; z \in \mathbb{C} \text{ s.t. } \sigma_{\min}(A^V(x) - zV) \leq \epsilon \right\}, \end{aligned} \tag{2.1}$$

we solve

$$\min_{x \in \underline{\Omega}} \alpha_\epsilon(A^V(x)) = \min_{x \in \underline{\Omega}} \max\left\{ \mathrm{Re}(z) \;\middle|\; z \in \mathbb{C} \text{ s.t. } \sigma_{\min}(A^V(x) - zV) \leq \epsilon \right\}. \tag{2.2}$$

rather than (1.3). Employing two-sided projections for the reduction of $A(x)$ may appear as a plausible strategy since $A(x)$ is a non-Hermitian matrix-valued function. However, two-sided projections cause convergence problems, such as the loss of a monotonicity property, which will be crucial in the convergence analysis discussed in Section 3.

We remark that $\Lambda_\epsilon(A^V(x))$ and $\alpha_\epsilon(A^V(x))$ are independent of the choice of the orthonormal basis used for $\mathcal{V}$, that is $\Lambda_\epsilon(A^{V_1}(x)) = \Lambda_\epsilon(A^{V_2}(x))$ and $\alpha_\epsilon(A^{V_1}(x)) = \alpha_\epsilon(A^{V_2}(x))$ for two different matrices V_1, V_2 whose columns form orthonormal bases for $\mathcal{V}$, as $\sigma_{\min}(A^{V_1}(x) - zV_1) = \sigma_{\min}(A^{V_2}(x) - zV_2)$ for any $z \in \mathbb{C}$ for such V_1, V_2. Accordingly, letting $\mathcal{V} := \mathrm{Col}(V)$, and hiding the dependence of the pseudospectra and pseudospectral abscissa on $A(x)$, we use the shorthand notations

$$\begin{aligned} \Lambda_\epsilon(x) &:= \Lambda_\epsilon(A(x)), \quad & \Lambda_\epsilon^{\mathcal{V}}(x) &:= \Lambda_\epsilon(A^V(x)), \\ \alpha_\epsilon(x) &:= \alpha_\epsilon(A(x)), \quad & \alpha_\epsilon^{\mathcal{V}}(x) &:= \alpha_\epsilon(A^V(x)) \end{aligned}$$

throughout the rest of this text.

The basic subspace framework for the minimization of $\alpha_\epsilon(x)$ is described in Algorithm 1. At each subspace iteration in Algorithm 1 in lines 6–10, first a small-scale reduced problem is solved in line 6, in particular, a global minimizer $\widetilde{x}$ is found for a reduced problem. Then a rightmost point $\widetilde{z}$ of $\Lambda_\epsilon(\widetilde{x})$ is determined in line 7. Finally, the subspace is expanded with the inclusion of a right singular vector corresponding to $\sigma_{\min}(A(\widetilde{x}) - \widetilde{z}I)$ in line 10.

Determining the rightmost point of $\Lambda_\epsilon(\widetilde{x})$ involves the large-scale matrix-valued function $A(x)$, and is usually the most expensive step computationally in a subspace iteration. For this task, we usually benefit in practice from the approach in [17], an approach that combines the globally-convergent criss-cross algorithm [10] for computing the pseudospectral abscissa with a subspace framework. On the other hand, the

Algorithm 1 The subspace framework to minimize $\alpha_\epsilon(x)$ over $\underline{\Omega}$

Input: The matrix valued function $A(x)$ and the feasible region $\underline{\Omega}$

Output: An estimate $\widehat{x}$ for $\arg\min_{x\in\underline{\Omega}} \alpha_\epsilon(x)$, and $\widehat{z} \in \mathbb{C}$ that is an estimate for a globally rightmost point in $\Lambda_\epsilon(\widehat{x})$

1: $x_1^{(1)}, \dots x_\eta^{(1)} \leftarrow$ initially chosen points in $\underline{\Omega}$.
2: $z_j^{(1)} \leftarrow \arg\max \left\{ \mathrm{Re}(z) \mid \sigma_{\min}(A(x_j^{(1)}) - zI) \le \epsilon \right\}$ for $j = 1, \dots, \eta$.
3: $v_j^{(1)} \leftarrow$ a right singular vector corr. to $\sigma_{\min}(A(x_j^{(1)}) - z_j^{(1)} I)$ for $j = 1, \dots, \eta$.
4: $\mathcal{V}_1 \leftarrow \mathrm{span}\left\{ v_1^{(1)}, \dots v_\eta^{(1)} \right\}$ and $V_1 \leftarrow$ an orthonormal basis for $\mathcal{V}_1$.
5: **for** $k = 2, 3, \dots$ **do**
6: $x^{(k)} \leftarrow \arg\min_{x\in\underline{\Omega}} \alpha_\epsilon^{\mathcal{V}_{k-1}}(x)$.
7: $z^{(k)} \leftarrow \arg\max \left\{ \mathrm{Re}(z) \mid \sigma_{\min}(A(x^{(k)}) - zI) \le \epsilon \right\}$.
8: **Return** $\widehat{x} \leftarrow x^{(k)}$, $\widehat{z} \leftarrow z^{(k)}$ if convergence occurred.
9: $v^{(k)} \leftarrow$ a unit right singular vector corresponding to $\sigma_{\min}(A(x^{(k)}) - z^{(k)} I)$.
10: $V_k \leftarrow \mathrm{orth}\left([V_{k-1} \;\; v^{(k)}] \right)$ and $\mathcal{V}_k \leftarrow \mathrm{Col}(V_k)$.
11: **end for**

small-scale reduced pseudspectral abscissa minimization problem is usually cheap to solve. For this task, we employ "eigopt", the globally convergent algorithm in [22], if there is only one parameter, or otherwise, if the matrix-valued function depends on multiple parameters, we use "GRANSO" [12]. Last but not the least, let us also note that the right singular vector of the large-scale matrix in line 9 should normally be calculated iteratively, for instance by means of "ARPACK". Further implementation details of the subspace framework, including the condition to check convergence employed in practice in line 8, are described in Section 5.1.

2.1 Basic Results Regarding the Subspace Framework

Next, we present two basic results concerning the subspace framework that will be crucial in the convergence analysis.

The first result is the monotonicity property with respect to the subspace $\mathcal{V}$. We refer to Lemma 3.1 and succeeding arguments in [17] for a proof.

Lemma 2.1 (Monotonicity)**.** *Let $\mathcal{V}, \mathcal{W}$ be two subspaces of $\mathbb{C}^n$ such that $\mathcal{V} \subseteq \mathcal{W}$, and V, W be matrices whose columns form orthonormal bases for $\mathcal{V}, \mathcal{W}$. Then the following assertions hold:*

(i) $\sigma_{\min}(A(x) - zI) \le \sigma_{\min}(A^W(x) - zW) \le \sigma_{\min}(A^V(x) - zV) \quad \forall z \in \mathbb{C}, \forall x \in \Omega.$

(ii) $\Lambda_\epsilon^{\mathcal{V}}(x) \subseteq \Lambda_\epsilon^{\mathcal{W}}(x) \subseteq \Lambda_\epsilon(x) \quad \forall x \in \Omega.$

(iii) $\alpha_\epsilon^{\mathcal{V}}(x) \leq \alpha_\epsilon^{\mathcal{W}}(x) \leq \alpha_\epsilon(x) \quad \forall x \in \Omega.$

The next result concerns the interpolation properties between the full and reduced problems, and is borrowed from [17, Lemma 3.2], where a proof is also provided.

Lemma 2.2. *For a given $\widetilde{x} \in \Omega$, and given subspace $\mathcal{V}$, the following are equivalent:*

(i) $\alpha_\epsilon^{\mathcal{V}}(\widetilde{x}) = \alpha_\epsilon(\widetilde{x})$.

(ii) *The subspace $\mathcal{V}$ contains a right singular vector corresponding to $\sigma_{\min}(A(\widetilde{x}) - \widetilde{z}I)$ for some $\widetilde{z} \in \Lambda_\epsilon(\widetilde{x})$ with $\mathrm{Re}(\widetilde{z}) = \alpha_\epsilon(\widetilde{x})$.*

Global convergence of the proposed subspace framework as well as its rapid convergence can be attributed to the interpolation properties between the full and the reduced problems. In a subsequent subsection, we establish interpolation properties between $\alpha_\epsilon(\cdot)$ and $\alpha_\epsilon^{\mathcal{V}_k}(\cdot)$ as well as between their first derivatives for the subspaces $\mathcal{V}_k$ generated by Algorithm 1. Before stating the result formally, we first derive formulas for the derivatives of $\alpha_\epsilon(\cdot)$ and $\alpha_\epsilon^{\mathcal{V}}(\cdot)$ for a given subspace $\mathcal{V}$ in the next subsection.

2.2 Derivatives of the Pseudospectral Abscissa

We consider the pseudospectral abscissa as a constrained optimization problem, and, based on optimality conditions for constrained optimization problems, we derive analytical formulas for the derivatives of the pseudospectral abscissa functions.

Letting $A(x, z) := A(x) - (z_1 + \mathrm{i}z_2)I$ and $\sigma(x, z) := \sigma_{\min}(A(x, z))$ for $z = (z_1, z_2) \in \mathbb{R}^2$, the ϵ-pseudospectral abscissa $\alpha_\epsilon(x)$ can be expressed as the constrained optimization problem

$$\alpha_\epsilon(x) \;=\; \max\{z_1 \;\mid\; z = (z_1, z_2) \in \mathbb{R}^2,\;\; \sigma(x, z) - \epsilon \leq 0\}. \tag{2.3}$$

We consider the Langrangian function

$$\mathcal{L}(x, z, \mu) \;:=\; z_1 - \mu(\sigma(x, z) - \epsilon) \tag{2.4}$$

associated with (2.3), where $\mu \geq 0$ is the Lagrange multiplier corresponding to the constraint $\sigma(x, z) - \epsilon \leq 0$.

For a given x, we denote the optimal $z = (z_1, z_2)$ and the corresponding μ for the optimization problem in (2.3) with $z(x) = (z_1(x), z_2(x))$ and $\mu(x)$, respectively. Moreover, we make use of the notations $y = (z, \mu)$ and $y(x) = (z(x), \mu(x))$. To ensure $\sigma(x, z(x))$ is differentiable, let us assume it is a simple singular value of $A(x, z(x))$. By the optimality conditions, $\partial\mathcal{L}(x, z(x), \mu(x))/\partial z_1 = 1 - \mu(x)\partial\sigma(x, z(x))/\partial z_1 = 0$ implying $\mu(x) \neq 0$, which in turn leads to

$$\sigma(x, z(x)) \;=\; \epsilon \tag{2.5}$$

by the complementarity conditions. It follows from (2.3) and (2.4) that

$$\alpha_\epsilon(x) \;=\; z_1(x) \;=\; \mathcal{L}(x, z(x), \mu(x))\,. \tag{2.6}$$

To guarantee the differentiability of $\alpha_\epsilon(x)$, we assume the satisfaction of the following conditions.

Assumption 2.3. *For a given $x \in \Omega$, we have that*

- **(i)** *the optimizer $z(x)$ of (2.3) is unique,*
- **(ii)** *the singular value $\sigma(x, z(x))$ of $A(x, z(x))$ is simple, and*
- **(iii)** *$\nabla_{yy}\mathcal{L}(x, y(x))$ is non-singular, where $\nabla_{yy}\mathcal{L}(\cdot)$ denotes the Hessian of $\mathcal{L}(\cdot)$ with respect to y.*

Condition (ii) of Assumption 2.3 implies that $\mathcal{L}(x, y)$ is differentiable with respect to y at $y = y(x)$ so that we can put first order optimality conditions in use. As argued in the previous paragraph, the first order optimality conditions already ensure $\mu(x) \neq 0$. To derive an expression for $\mu(x)$, we now exploit the first order optimality conditions

$$\nabla_y \mathcal{L}(x, y(x)) = 0. \tag{2.7}$$

for (2.3) in more detail. Using the analytical formulas for the derivatives of the singular value function (see, e.g., [22, Section 3.3]) in the last equation, we deduce

$$0 = 1 - \mu(x)\,\mathrm{Re}\left(u^* \frac{\partial A(x, z(x))}{\partial z_1} v\right) = 1 + \mu(x)\,\mathrm{Re}(u^* v)$$

and

$$0 = -\mu(x)\,\mathrm{Re}\left(u^* \frac{\partial A(x, z(x))}{\partial z_2} v\right) = -\mu(x)\,\mathrm{Im}(u^* v),$$

where u, v consist of a pair of consistent unit left and right singular vectors corresponding to $\sigma(x, z(x))$. The first equation implies $\mathrm{Re}(u^* v) \neq 0$. Furthermore, as $\mu(x) \neq 0$, the second equation gives rise to $\mathrm{Im}(u^* v) = 0$, that is $u^* v$ is real and so $u^* v = \mathrm{Re}(u^* v)$. It follows from the first equation that

$$\mu(x) \;=\; -\frac{1}{u^* v}.$$

Next we derive an analytical formula for $\nabla \alpha_\epsilon(x)$. Taking the gradient of both sides of (2.6), in particular using the chain rule, we get

$$\nabla \alpha_\epsilon(x) = \nabla_x \mathcal{L}(x, y(x)) + [y'(x)]^T \cdot \nabla_y \mathcal{L}(x, y(x))\,,$$

where $y'(x)$ denotes the $3\times d$ Jacobian matrix of y with respect to x. But $\nabla_y \mathcal{L}(x, y(x)) = 0$, so we have

$$\nabla \alpha_\epsilon(x) = \nabla_x \mathcal{L}(x, y(x)). \tag{2.8}$$

More specifically, denoting the jth component of x with x_j, we have

$$\frac{\partial \alpha_\epsilon(x)}{\partial x_j} = -\mu(x) \frac{\partial \sigma(x, z(x))}{\partial x_j}.$$

Again, using the analytical formula for the singular value function, and $\mu(x) = -1/(u^* v)$, which is real and nonzero, we obtain

$$\frac{\partial \alpha_\epsilon(x)}{\partial x_j} \;=\; \mathrm{Re}\left(\frac{u^* \frac{\partial A(x)}{\partial x_j} v}{u^* v} \right) \tag{2.9}$$

for $j = 1, \ldots, d$.

For the second derivatives of $\alpha_\epsilon(x)$, we differentiate (2.8) by applying the chain rule to obtain

$$\nabla^2 \alpha_\epsilon(x) \;=\; \nabla^2_{xx} \mathcal{L}(x, y(x)) + \nabla^2_{xy} \mathcal{L}(x, y(x)) \cdot y'(x) \;, \tag{2.10}$$

where $\nabla^2_{xy} \mathcal{L}(\cdot)$ is the $d \times 3$ Hessian matrix of $\mathcal{L}(\cdot)$ for which the partial derivatives are first taken with respect to x and then with respect to y, whereas $\nabla^2_{xx} \mathcal{L}(\cdot)$ is the standard $d \times d$ Hessian matrix of $\mathcal{L}(\cdot)$ with respect to x. Differentiating (2.7) with respect to x yields

$$\nabla^2_{yx} \mathcal{L}(x, y(x)) + \nabla^2_{yy} \mathcal{L}(x, y(x)) \cdot y'(x) \;=\; 0$$

which in turn gives rise to

$$\begin{aligned} y'(x) \;&=\; -[\nabla^2_{yy} \mathcal{L}(x, y(x))]^{-1} \nabla^2_{yx} \mathcal{L}(x, y(x)) \\ &=\; -[\nabla^2_{yy} \mathcal{L}(x, y(x))]^{-1} [\nabla^2_{xy} \mathcal{L}(x, y(x))]^T. \end{aligned} \tag{2.11}$$

Substituting (2.11) into (2.10), we obtain

$$\nabla^2 \alpha_\epsilon(x) = \nabla^2_{xx} \mathcal{L}(x, y(x)) - \nabla^2_{xy} \mathcal{L}(x, y(x)) \cdot [\nabla^2_{yy} \mathcal{L}(x, y(x))]^{-1} [\nabla^2_{xy} \mathcal{L}(x, y(x))]^T \tag{2.12}$$

where

$$\begin{aligned} [\nabla^2_{xx} \mathcal{L}(x, y(x))]_{j\ell} \;&=\; -\mu(x) \frac{\partial^2 \sigma(x, z(x))}{\partial x_j \partial x_\ell} \qquad j, \ell = 1, 2, \cdots, d, \\ [\nabla^2_{xy} \mathcal{L}(x, y(x))]_{j\ell} \;&=\; -\mu(x) \frac{\partial^2 \sigma(x, z(x))}{\partial x_j \partial y_\ell} \qquad j = 1, 2, \cdots d, \;\; \ell = 1, 2, 3, \\ [\nabla^2_{yy} \mathcal{L}(x, y(x))]_{j\ell} \;&=\; -\frac{\partial^2 \{\mu(x) \sigma(x, z(x))\}}{\partial y_j \partial y_\ell} \quad j, \ell = 1, 2, 3, \end{aligned}$$

and $(y_1, y_2, y_3) = (z_1, z_2, \mu)$. Expressions for the third partial derivatives of $\alpha_\epsilon(\cdot)$ can be derived in a similar way by differentiating (2.10) further.

If we repeat the arguments above for the reduced pseudospectral abscissa function $\alpha_\epsilon^{\mathcal{V}}(\cdot)$ for a given subspace $\mathcal{V}$, analogous formulas for the derivatives of $\alpha_\epsilon^{\mathcal{V}}(\cdot)$ can be obtained in terms of a matrix V whose columns form an orthonormal basis for $\mathcal{V}$. Specifically, the Lagrangian for the optimization problem associated with $\alpha_\epsilon^{\mathcal{V}}(\cdot)$ takes the form

$$\mathcal{L}^{\mathcal{V}}(x, z, \mu) \;:=\; z_1 - \mu(\sigma^{\mathcal{V}}(x, z) - \epsilon),$$

where $\sigma^{\mathcal{V}}(x, z) := \sigma_{\min}(A^V(x, z))$ and $A^V(x, z) := A^V(x) - zV$. Denoting the optimal z and the corresponding Lagrange multiplier μ now by $z^{\mathcal{V}}(x)$ and $\mu^{\mathcal{V}}(x)$ at a given x, the Lagrangian satisfies $\alpha_\epsilon^{\mathcal{V}}(x) = \mathcal{L}^{\mathcal{V}}(x, z^{\mathcal{V}}(x), \mu^{\mathcal{V}}(x))$. Applying steps analogous to the ones above, one can obtain

$$\nabla \alpha_\epsilon^{\mathcal{V}}(x) = \nabla_x \mathcal{L}^{\mathcal{V}}(x, y^{\mathcal{V}}(x)),$$

where $y^{\mathcal{V}}(x) = (z^{\mathcal{V}}(x), \mu^{\mathcal{V}}(x))$. In particular,

$$\frac{\partial \alpha_\epsilon^{\mathcal{V}}(x)}{\partial x_j} = \mathrm{Re}\left(\frac{[u^{\mathcal{V}}]^* \frac{\partial A^V(x)}{\partial x_j} v^{\mathcal{V}}}{[u^{\mathcal{V}}]^* V\, v^{\mathcal{V}}}\right) \tag{2.13}$$

for $j = 1, \ldots, d$, where $u^{\mathcal{V}}, v^{\mathcal{V}}$ form a pair of consistent unit left and unit right singular vectors corresponding to $\sigma^{\mathcal{V}}(x, z^{\mathcal{V}}(x))$.

The second derivatives of $\alpha_\epsilon^{\mathcal{V}}(x)$ are given by

$$\begin{aligned} \nabla^2 \alpha_\epsilon^{\mathcal{V}}(x) \;=\; & \nabla^2_{xx} \mathcal{L}^{\mathcal{V}}(x, y^{\mathcal{V}}(x)) \;- \\ & \nabla^2_{xy} \mathcal{L}^{\mathcal{V}}(x, y^{\mathcal{V}}(x)) [\nabla^2_{yy} \mathcal{L}^{\mathcal{V}}(x, y^{\mathcal{V}}(x))]^{-1} [\nabla^2_{xy} \mathcal{L}^{\mathcal{V}}(x, y^{\mathcal{V}}(x))]^T, \end{aligned} \tag{2.14}$$

where

$$\begin{aligned} [\nabla^2_{xx} \mathcal{L}^{\mathcal{V}}(x, y^{\mathcal{V}}(x))]_{j\ell} \;&=\; -\mu^{\mathcal{V}}(x) \frac{\partial^2 \sigma^{\mathcal{V}}(x, z^{\mathcal{V}}(x))}{\partial x_j \partial x_\ell} \qquad j, \ell = 1, 2, \cdots d, \\ [\nabla^2_{xy} \mathcal{L}^{\mathcal{V}}(x, y^{\mathcal{V}}(x))]_{j\ell} \;&=\; -\mu^{\mathcal{V}}(x) \frac{\partial^2 \sigma(x, z^{\mathcal{V}}(x))}{\partial x_j \partial y_\ell} \qquad j = 1, 2, \cdots d, \;\; \ell = 1, 2, 3, \\ [\nabla^2_{yy} \mathcal{L}^{\mathcal{V}}(x, y^{\mathcal{V}}(x))]_{j\ell} \;&=\; -\frac{\partial^2 \{\mu^{\mathcal{V}}(x) \sigma(x, z^{\mathcal{V}}(x))\}}{\partial y_j \partial y_\ell} \qquad j, \ell = 1, 2, 3. \end{aligned}$$

2.3 Hermite Interpolation of the Pseudospectral Abscissa

The next result concerns the Hermite interpolation properties between the full pseudospectral abscissa function $\alpha_\epsilon(\cdot)$ and its reduced counterpart $\alpha_\epsilon^{\mathcal{V}_k}(\cdot)$ for a subspace $\mathcal{V}_k$ generated by Algorithm 1.

Lemma 2.4 (Hermite Interpolation)**.** *The following assertions hold regarding the subspace $\mathcal{V}_k$ for every integer $k \geq 2$ and the points $x^{(\ell)}$ for $\ell = 2, \ldots, k$ generated by Algorithm 1:*

(i) $\alpha_\epsilon(x^{(\ell)}) = \alpha_\epsilon^{\mathcal{V}_k}(x^{(\ell)})$.

(ii) *If the conditions in Assumption 2.3 hold for $x = x^{(\ell)}$, then $\alpha_\epsilon(x)$ and $\alpha_\epsilon^{\mathcal{V}_k}(x)$ are differentiable at $x = x^{(\ell)}$, and*

$$\nabla \alpha_\epsilon(x^{(\ell)}) = \nabla \alpha_\epsilon^{\mathcal{V}_k}(x^{(\ell)}).$$

Proof. **(i)** From lines 9, 10 of Algorithm 1, we have $v^{(\ell)} \in \mathcal{V}_k$, a right singular vector corresponding to $\sigma_{\min}(A(x^{(\ell)}) - z^{(\ell)} I)$ for the rightmost point $z^{(\ell)}$ in $\Lambda_\epsilon(x^{(\ell)})$, for $\ell = 2, \ldots k$. As a result, the assertion follows from Lemma 2.2.

(ii) Differentiability of $\alpha_\epsilon(x)$ at $x = x^{(\ell)}$ is immediate, as Assumption 2.3 is satisfied at $x = x^{(\ell)}$. Let us show that $\alpha_\epsilon^{\mathcal{V}_k}(x)$ is also differentiable at $x = x^{(\ell)}$. As $v^{(\ell)} \in \mathcal{V}_k$, there exists a vector a such that $v^{(\ell)} = V_k a$, and so

$$\epsilon \geq \sigma_{\min}(A(x^{(\ell)}) - z^{(\ell)}) = \|(A(x^{(\ell)}) - z^{(\ell)})V_k a\|_2 \geq \sigma_{\min}(A(x^{(\ell)})V_k - z^{(\ell)}V_k).$$

This shows that $z^{(\ell)} \in \Lambda_\epsilon^{\mathcal{V}_k}(x^{(\ell)})$, and so $\alpha_\epsilon(x^{(\ell)}) = \mathrm{Re}(z^{(\ell)}) \leq \alpha_\epsilon^{\mathcal{V}_k}(x^{(\ell)})$. But the reverse inequality $\alpha_\epsilon(x^{(\ell)}) \geq \alpha_\epsilon^{\mathcal{V}_k}(x^{(\ell)})$ also holds due to Lemma 2.1 implying $\mathrm{Re}(z^{(\ell)}) = \alpha_\epsilon^{\mathcal{V}_k}(x^{(\ell)})$, i.e., $z^{(\ell)}$ is a rightmost point in $\Lambda_\epsilon^{\mathcal{V}_k}(x^{(\ell)})$.

We claim that $z^{(\ell)}$ is the unique rightmost point in $\Lambda_\epsilon^{\mathcal{V}_k}(x^{(\ell)})$. Suppose otherwise for the sake of contradiction, that is there exists $\widetilde{z} \neq z^{(\ell)}$ such that $\widetilde{z} \in \Lambda_\epsilon^{\mathcal{V}_k}(x^{(\ell)})$ and $\alpha_\epsilon^{\mathcal{V}_k}(x^{(\ell)}) = \mathrm{Re}(z^{(\ell)}) = \mathrm{Re}(\widetilde{z})$. But then, by Lemma 2.1, we have $\epsilon \geq \sigma_{\min}(A^{V_k}(x^{(\ell)}) - \widetilde{z} V_k) \geq \sigma_{\min}(A(x^{(\ell)}) - \widetilde{z} I)$, so $\widetilde{z} \in \Lambda_\epsilon(A(x^\ell))$. Consequently, $\widetilde{z}$ and $z^{(\ell)}$ are rightmost points in $\Lambda_\epsilon(x^{(\ell)})$, contradicting the assumption that $\Lambda_\epsilon(x^{(\ell)})$ has a unique rightmost point (i.e., contradicting Assumption 2.3 with $x = x^{(\ell)}$).

Furthermore, if $\sigma_{\min}(A^{V_k}(x^{(\ell)}) - z^{(\ell)}V_k)$ is not simple, then there exist two unit orthonormal vectors $a, \widetilde{a}$ that satisfy

$$\begin{aligned} \sigma_{\min}(A(x^{(\ell)}) - z^{(\ell)} I) &= \sigma_{\min}(A^{V_k}(x^{(\ell)}) - z^{(\ell)} V_k) \\ &= \left\|(A^{V_k}(x^{(\ell)}) - z^{(\ell)} V_k)a\right\|_2 = \left\|(A^{V_k}(x^{(\ell)}) - z^{(\ell)} V_k)\widetilde{a}\right\|_2. \end{aligned}$$

But then $\omega = V_k a$ and $\widetilde{w} = V_k \widetilde{a}$ are orthonormal vectors satisfying

$$\sigma_{\min}(A(x^{(\ell)}) - z^{(\ell)} I) = \left\|(A(x^{(\ell)}) - z^{(\ell)})\omega\right\|_2 = \left\|(A(x^{(\ell)}) - z^{(\ell)})\widetilde{\omega}\right\|_2,$$

which contradicts with the simplicity of $\sigma_{\min}(A(x^{(\ell)}) - z^{(\ell)}I)$.

The arguments above show that $z^{(\ell)}$ is the unique rightmost point in $\Lambda_\epsilon^{\mathcal{V}_k}(x^{(\ell)})$, and that $\sigma_{\min}(A^{V_k}(x^{(\ell)}) - z^{(\ell)}V_k)$ is simple. Consequently, $\alpha_\epsilon^{\mathcal{V}_k}(x)$ is differentiable at $x = x^{(\ell)}$.

Finally, we show that the gradients of $\alpha_\epsilon(x)$ and $\alpha_\epsilon^{\mathcal{V}_k}(x)$ are equal at $x = x^{(\ell)}$. Let u, v be a pair of unit left and unit right singular vectors corresponding to $\sigma := \sigma_{\min}(A(x^{(\ell)}) - z^{(\ell)}I)$ satisfying

$$(A(x^{(\ell)}) - z^{(\ell)}I)v = \sigma u \qquad \text{and} \qquad u^*(A(x^{(\ell)}) - z^{(\ell)}I) = \sigma v^*.$$

Since $v \in \mathcal{V}_k$, there exists a unit vector a such that $v = V_k a$. Then the equations above can be rewritten in terms of a as follows:

$$(A^{V_k}(x^{(\ell)}) - z^{(\ell)}V_k)a = \sigma u \qquad \text{and} \qquad u^*(A^{V_k}(x^{(\ell)}) - z^{(\ell)}V_k) = \sigma a^*.$$

This means that u, a are a pair of unit left and unit right singular vectors corresponding to $\sigma_{\min}(A^{V_k}(x^{(\ell)}) - z^{(\ell)}V_k)$.

Using the analytical formulas derived in the previous subsection for the derivatives of $\alpha_\epsilon(\cdot)$ and $\alpha_\epsilon^{\mathcal{V}_k}()$, specifically using (2.9) and (2.13), we obtain

$$\begin{aligned} \frac{\partial \alpha_\epsilon(x^{(\ell)})}{\partial x_j} &= \text{Re}\left(\frac{u^* \frac{\partial A(x^{(\ell)})}{\partial x_j} v}{u^* v}\right) \\ &= \text{Re}\left(\frac{u^* \frac{\partial A^{V_k}(x^{(\ell)})}{\partial x_j} a}{u^* V_k a}\right) = \frac{\partial \alpha_\epsilon^{\mathcal{V}_k}(x^{(\ell)})}{\partial x_j}. \end{aligned}$$

for $j = 1, \ldots, d$.

□

3 Global Convergence

We now relate the sequence $x^{(2)}, x^{(3)}, x^{(4)}, \ldots$ generated by Algorithm 1 and the global minimizer of $\alpha_\epsilon(x)$ over all $x \in \underline{\Omega}$. The first result below asserts that if any two iterates are equal, then global convergence is achieved.

Theorem 3.1. *If two points $x^{(\ell)}, x^{(k)}$ with $2 \le \ell < k$ generated by Algorithm 1 are equal, then $x^{(\ell)}$ is a global minimizer of $\alpha_\epsilon(x)$ over all $x \in \underline{\Omega}$.*

Proof. It follows from the monotonicity property (i.e., Lemma 2.1) that

$$\alpha_\epsilon^{\mathcal{V}_\ell}(x^{(k)}) \;\le\; \alpha_\epsilon^{\mathcal{V}_{k-1}}(x^{(k)}) \;=\; \min_{x\in\underline{\Omega}} \alpha_\epsilon^{\mathcal{V}_{k-1}}(x) \;\le\; \min_{x\in\underline{\Omega}} \alpha_\epsilon(x).$$

Moreover, an implication of the interpolation property (i.e., Lemma 2.2) is that

$$\min_{x\in\underline{\Omega}} \alpha_\epsilon(x) \;\le\; \alpha_\epsilon(x^{(\ell)}) \;=\; \alpha_\epsilon^{\mathcal{V}_\ell}(x^{(\ell)}) \;=\; \alpha_\epsilon^{\mathcal{V}_\ell}(x^{(k)}),$$

where the last equality is due to $x^{(\ell)} = x^{(k)}$. Combining the inequalities above, $\min_{x\in\underline{\Omega}} \alpha_\epsilon(x) = \alpha_\epsilon^{\mathcal{V}_\ell}(x^{(k)}) = \alpha_\epsilon(x^{(\ell)})$ as desired. □

Remark. The subspace $\mathcal{V}_\ell$ is a subspace of $\mathbb{C}^n$ and contains $\mathcal{V}_{\ell-1}$ for every $\ell > 2$. As a result, we must have $\mathcal{V}_\ell = \mathcal{V}_{\ell-1}$ (so that $x^{(\ell+1)} = x^{(\ell)}$) for some $\ell > 2$. Theorem 3.1 implies $x^{(\ell)}$ for such an ℓ is a global minimizer of $\alpha_\epsilon(x)$ over $x \in \underline{\Omega}$.

The global convergence of the subspace framework can remarkably be extended even to the infinite dimensional setting when $A(x)$ is in essence an infinite dimensional matrix so that the subspaces can grow arbitrarily as we show next. Formally, following the practices in [15, 21], throughout the rest of this section we assume $A_1, \dots A_\kappa$ are linear bounded operators on $\ell_2(\mathbb{N})$, which denotes the inner product space of square summable complex infinite sequences equipped with the inner product $\langle v, w\rangle = \sum_{j=0}^{\infty} \overline{v}_j w_j$ and the norm $\|v\|_2 = \sqrt{\sum_{j=0}^{\infty} |v_j|^2}$. We additionally assume that $A(x)$ for all $x \in \underline{\Omega}$ has countably many singular values. The monotonicity result (i.e., Lemma 2.1) holds in the infinite dimensional setting as well. Moreover, an application of Algorithm 1 in this infinite dimensional setting still produces sequences satisfying Hermite interpolation, i.e., Lemma 2.4 and its proof still hold.

As shown in the proof of Lemma 2.4, if $u^{(k)}$ and $v^{(k)}$ denote a consistent pair of unit left and right singular vectors corresponding to $\sigma_{\min}(A(x^{(k)}) - z^{(k)}I)$, then $u^{(k)}$ and $a^{(k)}$ satisfying $v_k = V_k a^{(k)}$ form a consistent pair of left and right singular vectors corresponding to $\sigma_{\min}(A^{V_k}(x^{(k)}) - z^{(k)}V_k)$. As argued in Section 2.2 assuming $\sigma_{\min}(A^{V_k}(x^{(k)}) - z^{(k)}V_k)$ is simple, by exploiting the fact that $z^{(k)}$ is the rightmost point in $\Lambda_\epsilon(x^{(k)})$, and employing the associated first order optimality conditions, $[u^{(k)}]^* v^{(k)}$ is real and nonzero, indeed $[u^{(k)}]^* v^{(k)} < 0$. In terms of the reduced problem, this amounts to

$$0 > [u^{(k)}]^* v^{(k)} = [u^{(k)}]^* V_k a^{(k)} = -\frac{\partial \sigma_{\min}(A^{V_k}(x^{(k)}) - z^{(k)}V_k)}{\partial z_1}.$$

We remind that $z^{(k)}$ above is also the rightmost point in $\Lambda_\epsilon^{\mathcal{V}_k}(x^{(k)})$.

Indeed, letting $u^{\mathcal{V}_k}(x)$, $a^{\mathcal{V}_k}(x)$ be unit consistent left, right singular vectors corresponding to $\sigma_{\min}(A^{V_k}(x) - z^{\mathcal{V}_k}(x)V_k)$ with $z^{\mathcal{V}_k}(x)$ representing the rightmost point in $\Lambda_\epsilon^{\mathcal{V}_k}(x)$, the property

$$0 \ > \ u^{\mathcal{V}_k}(x)^* V_k a^{\mathcal{V}_k}(x) \tag{3.1}$$

holds uniformly for all $x \in \Omega$ where $\alpha_\epsilon^{\mathcal{V}_k}(x)$ is differentiable as we argue next. By applying the first-order optimality conditions to the Lagrangian $\mathcal{L}^{\mathcal{V}_k}(x, z, \mu) \ := \ z_1 - \mu(\sigma^{\mathcal{V}_k}(x, z) - \epsilon)$, we deduce

$$0 = 1 - \mu^{\mathcal{V}_k}(x) \operatorname{Re}\left(u^{\mathcal{V}_k}(x)^* \frac{\partial A^{V_k}(x, z^{\mathcal{V}_k}(x))}{\partial z_1} a^{\mathcal{V}_k}(x)\right) = 1 + \mu^{\mathcal{V}_k}(x) \operatorname{Re}(u^{\mathcal{V}_k}(x)^* V_k a^{\mathcal{V}_k}(x))$$

and

$$0 = -\mu^{\mathcal{V}_k}(x) \operatorname{Re}\left(u^{\mathcal{V}_k}(x)^* \frac{\partial A^{V_k}(x, z^{\mathcal{V}_k}(x))}{\partial z_2} a^{\mathcal{V}_k}(x)\right) = -\mu^{\mathcal{V}_k}(x) \operatorname{Im}(u^{\mathcal{V}_k}(x)^* V_k a^{\mathcal{V}_k}(x)).$$

From the first equation, $\mu^{\mathcal{V}_k}(x) \neq 0$, so, from the second equation, $\operatorname{Im}(u^{\mathcal{V}_k}(x)^* V_k a^{\mathcal{V}_k}(x)) = 0$, which shows $u^{\mathcal{V}_k}(x)^* V_k a^{\mathcal{V}_k}(x)$ is real. Additionally, $u^{\mathcal{V}_k}(x)^* V_k a^{\mathcal{V}_k}(x) \neq 0$ due to the first equation. Moreover, as $z^{\mathcal{V}_k}(x)$ is the rightmost point in $\Lambda_\epsilon^{\mathcal{V}_k}(x^{(k)})$, we have

$$0 \ \leq \ \frac{\partial \sigma_{\min}(A^{V_k}(x) - z^{\mathcal{V}_k}(x)V_k)}{\partial z_1} \ = \ -u^{\mathcal{V}_k}(x)^* V_k a^{\mathcal{V}_k}(x).$$

Combining this with $u^{\mathcal{V}_k}(x)^* V_k a^{\mathcal{V}_k}(x) \neq 0$ yields (3.1).

The uniform negativity feature in (3.1) is essential for the global convergence of the subspace framework. We indeed assume slightly more, namely a uniform negative local upper bound on the terms $u^{\mathcal{V}_k}(x)^* V_k a^{\mathcal{V}_k}(x)$ near $x^{(k)}$ as stated formally below. Throughout the rest of this section and next section, $\mathcal{B}(c, r)$ for a given $c \in \mathbb{R}^q$ and $r \in \mathbb{R}$ denotes the closed ball

$$\mathcal{B}(c, r) \ := \ \{\, \widetilde{c} \in \mathbb{R}^q \ \mid \ \|\widetilde{c} - c\|_2 \ \leq \ r \,\},$$

which reduces to the closed interval $[c - r, c + r]$ in case $q = 1$.

Assumption 3.2. *There exist $\varphi > 0$ and $\beta < 0$ such that for all $k \geq 2$ the following inequality holds:*

$$\beta \ > \sup_{x \in \Omega^{(k)} \cap \mathcal{B}(x^{(k)}, \varphi)} u^{\mathcal{V}_k}(x)^* V_k a^{\mathcal{V}_k}(x),$$

where $\Omega^{(k)}$ denotes the subset of Ω over which $\alpha_\epsilon^{\mathcal{V}_k}(x)$ is differentiable.

The following uniform local Lipschitz continuity result for $x \mapsto \alpha_\epsilon^{\mathcal{V}_k}(x)$ plays a prominent role when establishing the global convergence of the subspace framework.

Lemma 3.3 (Uniform Local Lipschitz Continuity)**.** *Suppose Assumption 3.2 is satisfied. There exist real scalars $\zeta > 0$ and $\varphi > 0$ such that for all $k \geq 2$ we have $\left\|\nabla \alpha_\epsilon^{\mathcal{V}_k}(x)\right\| \leq \zeta$ for all $x \in \Omega \cap \mathcal{B}(x^{(k)}, \varphi)$ where $\alpha_\epsilon^{\mathcal{V}_k}(x)$ is differentiable.*
Consequently, $\alpha_\epsilon^{\mathcal{V}_k}(x)$ is uniformly locally Lipschitz, that is there exist $\zeta > 0$ and $\varphi > 0$ such that for all $k \geq 2$ the following assertion holds:

$$\left|\alpha_\epsilon^{\mathcal{V}_k}(x) - \alpha_\epsilon^{\mathcal{V}_k}(\widetilde{x})\right| \leq \zeta \|x - \widetilde{x}\|_2 \quad \forall x, \widetilde{x} \in \Omega \cap \mathcal{B}(x^{(k)}, \varphi).$$

Proof. Let $\varphi > 0$ and $\beta < 0$ be as in Assumption 3.2, and $\widetilde{x} \in \Omega \cap \mathcal{B}(x^{(k)}, \varphi)$ be such that $\alpha_\epsilon^{\mathcal{V}_k}(\widetilde{x})$ is differentiable. The partial derivatives of $\alpha_\epsilon^{\mathcal{V}_k}(x)$ at $x = \widetilde{x}$ are given by

$$\begin{aligned}
\frac{\partial \alpha_\epsilon^{\mathcal{V}_k}(\widetilde{x})}{\partial x_j} &= \operatorname{Re}\left(\frac{u^{\mathcal{V}_k}(\widetilde{x})^* \frac{\partial\ A^{V_k}(\widetilde{x})}{\partial x_j} a^{\mathcal{V}_k}(\widetilde{x})}{u^{\mathcal{V}_k}(\widetilde{x})^* V_k a^{\mathcal{V}_k}(\widetilde{x})}\right) \\
&= \operatorname{Re}\left(\sum_{\ell=1}^{\kappa} \frac{\partial f_\ell(\widetilde{x})}{\partial x_j} \frac{u^{\mathcal{V}_k}(\widetilde{x})^* A_\ell V_k a^{\mathcal{V}_k}(\widetilde{x})}{u^{\mathcal{V}_k}(\widetilde{x})^* V_k a^{\mathcal{V}_k}(\widetilde{x})}\right)
\end{aligned}$$

for $j = 1, \ldots, d$, which yields

$$\begin{aligned}
\left|\frac{\partial \alpha_\epsilon^{\mathcal{V}_k}(\widetilde{x})}{\partial x_j}\right| &\leq \sum_{\ell=1}^{\kappa} \left|\frac{\partial f_\ell(\widetilde{x})}{\partial x_j}\right| \left|\frac{u^{\mathcal{V}_k}(\widetilde{x})^* A_\ell V_k a^{\mathcal{V}_k}(\widetilde{x})}{u^{\mathcal{V}_k}(\widetilde{x})^* V_k a^{\mathcal{V}_k}(\widetilde{x})}\right| \\
&\leq \sum_{\ell=1}^{\kappa} \left|\frac{\partial f_\ell(\widetilde{x})}{\partial x_j}\right| \frac{\|A_\ell\|_2}{|\beta|},
\end{aligned}$$

where the last summation is independent of $\mathcal{V}_k$. This shows the existence of a γ such that for all $k \geq 2$ the bound $\left\|\nabla \alpha_\epsilon^{\mathcal{V}_k}(x)\right\| \leq \zeta$ holds for all $x \in \Omega \cap \mathcal{B}(x^{(k)}, \varphi)$ where $\alpha_\epsilon^{\mathcal{V}_k}(x)$ is differentiable.

Due to the boundedness of the gradients, the uniform Lipschitz continuity follows from the continuity of $\alpha_\epsilon^{\mathcal{V}_k}(x)$, and a simple application of the mean value theorem. □

Now we are ready for the global convergence result. Its proof follows the footsteps of [15, Theorem 3.1], yet we present it below for completeness.

Theorem 3.4 (Global Convergence)**.** *Suppose Assumption 3.2 is satisfied. Then the following hold for Algorithm 1 in the infinite dimensional setting:*

- ***(i)*** *The limit of every convergent subsequence of $\left\{x^{(k)}\right\}$ is a global minimizer of $\alpha_\epsilon(x)$ over all $x \in \underline{\Omega}$.*
- ***(ii)*** $\lim_{k\to\infty} \alpha_\epsilon^{\mathcal{V}_k}(x^{(k+1)}) = \lim_{k\to\infty} \min_{x\in\underline{\Omega}} \alpha_\epsilon^{\mathcal{V}_k}(x) = \min_{x\in\underline{\Omega}} \alpha_\epsilon(x).$

Proof. Let $\{x^{(\ell_k)}\}$ be a convergent subsequence of $\{x^{(k)}\}$. By the monotonicity property (i.e., Lemma 2.1, part **(iii)**), we have

$$\min_{x\in\underline{\Omega}} \alpha_\epsilon(x) \;\geq\; \min_{x\in\underline{\Omega}} \alpha_\epsilon^{\mathcal{V}_{\ell_{k+1}}-1}(x) \;=\; \alpha_\epsilon^{\mathcal{V}_{\ell_{k+1}}-1}(x^{(\ell_{k+1})}) \;\geq\; \alpha_\epsilon^{\mathcal{V}_{\ell_k}}(x^{(\ell_{k+1})}). \tag{3.2}$$

Additionally, due to interpolation property (i.e., Lemma 2.4, part **(i)**),

$$\min_{x\in\underline{\Omega}} \alpha_\epsilon(x) \;\leq\; \alpha_\epsilon(x^{(\ell_k)}) \;=\; \alpha_\epsilon^{\mathcal{V}_{\ell_k}}(x^{(\ell_k)}). \tag{3.3}$$

By combining (3.2) and (3.3), we obtain

$$\alpha_\epsilon^{\mathcal{V}_{\ell_k}}(x^{(\ell_{k+1})}) \;\leq\; \min_{x\in\underline{\Omega}} \alpha_\epsilon(x) \;\leq\; \alpha_\epsilon^{\mathcal{V}_{\ell_k}}(x^{(\ell_k)}).$$

It follows from the uniform local Lipschitz continuity of $\alpha_\epsilon^{\mathcal{V}_k}(x)$ with respect to k (i.e., Lemma 3.3), and noting $\|x^{(\ell_{k+1})} - x^{(\ell_k)}\|_2 \leq \varphi$ for k large enough as $\{x^{(\ell_k)}\}$ is convergent, that

$$\lim_{k\to\infty} \left|\alpha_\epsilon^{\mathcal{V}_{\ell_k}}(x^{(\ell_{k+1})}) - \alpha_\epsilon^{\mathcal{V}_{\ell_k}}(x^{(\ell_k)})\right| \;\leq\; \lim_{k\to\infty} \zeta \left\|x^{(\ell_{k+1})} - x^{(\ell_k)}\right\|_2 \;=\; 0,$$

where ζ is the uniform Lipschitz constant in Lemma 3.3. This together with the interpolation property (i.e., Lemma 2.4) imply that

$$\lim_{k\to\infty} \alpha_\epsilon(x^{(\ell_k)}) \;=\; \lim_{k\to\infty} \alpha_\epsilon^{\mathcal{V}_{\ell_k}}(x^{(\ell_k)}) \;=\; \lim_{k\to\infty} \alpha_\epsilon^{\mathcal{V}_{\ell_k}}(x^{(\ell_{k+1})}) \;=\; \min_{x\in\underline{\Omega}} \alpha_\epsilon(x). \tag{3.4}$$

Using the continuity of $\alpha_\epsilon(x)$ completes the proof of part **(i)**.

Let $\alpha_* := \min_{x\in\underline{\Omega}} \alpha_\epsilon(x)$. For $p > k$, using monotonicity, we have

$$\begin{aligned} \alpha_* \;&\geq\; \min_{x\in\underline{\Omega}} \alpha_\epsilon^{\mathcal{V}_p}(x) \;=\; \alpha_\epsilon^{\mathcal{V}_p}(x^{(p+1)}) \\ &\geq\; \alpha_\epsilon^{\mathcal{V}_k}(x^{(p+1)}) \;\geq\; \alpha_\epsilon^{\mathcal{V}_k}(x^{(k+1)}). \end{aligned}$$

This shows that the sequence $\{\alpha_\epsilon^{\mathcal{V}_k}(x^{(k+1)})\}$ is monotonically increasing and bounded from above by α_*, so it is convergent. Consider the subsequence $\left\{\alpha_\epsilon^{\mathcal{V}_{\ell_{k+1}}-1}(x^{(\ell_{k+1})})\right\}$ of $\{\alpha_\epsilon^{\mathcal{V}_k}(x^{(k+1)})\}$, which satisfies

$$\alpha_\epsilon^{\mathcal{V}_{\ell_k}}(x^{(\ell_{k+1})}) \;\leq\; \alpha_\epsilon^{\mathcal{V}_{\ell_{k+1}}-1}(x^{(\ell_{k+1})}) \;\leq\; \alpha_*.$$

Since we have $\lim_{k\to\infty} \alpha_\epsilon^{\mathcal{V}_{\ell_k}}(x^{(\ell_{k+1})}) = \alpha_*$ from the first part, in particular from (3.4), $\left\{\alpha_\epsilon^{\mathcal{V}_{\ell_{k+1}}-1}(x^{(\ell_{k+1})})\right\}$ also converges to α_*. Finally, the limit of the convergent sequence $\{\alpha_\epsilon^{\mathcal{V}_k}(x^{(k+1)})\}$ must be the same as the limit α_* of its subsequence. □

4 Rate of Convergence

In this section, we prove that the rate of convergence of Algorithm 1 when $d = 1$ is superlinear under mild assumptions. It seems possible to generalize the arguments and the result when $d > 1$ provided that additional singular vectors are put into the subspace at every iteration at points close to the interpolation point employed by Algorithm 1; this extension is similar to the extension of Algorithm 1 to Algorithm 2 in [15] in the context of minimizing the jth largest eigenvalue for a prescribed j. However, we only focus on the case $d = 1$ to keep the presentation simpler.

Throughout, it is assumed that three consecutive iterates $x^{(k-1)}, x^{(k)}, x^{(k+1)}$ generated by Algorithm 1 for $d = 1$ are sufficiently close to x_*, a global minimizer of $\alpha_\epsilon(x)$ over $x \in \Omega$. An additional assumption that is kept throughout is that x_* is strictly in the interior of $\underline{\Omega}$. Moreover, by z_* we denote a point in $\Lambda(x_*)$ satisfying $\alpha_\epsilon(x_*) = \mathrm{Re}(z_*)$. Furthermore, we assume that the following conditions hold to ensure the differentiability of $\alpha_\epsilon(x)$ at x_*.

Assumption 4.1 (Smoothness)**.** *The following conditions are satisfied by x_* and z_*:*

(i) *z_* is the unique point in $\Lambda_\epsilon(x_*)$ such that $\mathrm{Re}(z_*) = \alpha_\epsilon(x_*)$.*

(ii) *$\sigma(x_*, z_*)$ is simple.*

The smoothness assumption above guarantees also the twice and higher order differentiability of the Lagrangian $\mathcal{L}(x, y)$ with respect to y. We keep another assumption throughout this section, which concerns the nondegeneracy of these second derivatives of the Lagrangian and its reduced counterpart. This assumption stated next ensures the existence of the second and higher order derivatives of the full and reduced pseudospectral functions. Note that, in this assumption and in subsequent arguments, $y_* = (z_*, \mu_*)$ with $\mu_* \in \mathbb{R}$ represents the pair of optimal points satisfying $\nabla_y \mathcal{L}(x_*, y_*) = 0$.

Assumption 4.2 (Nondegeneracy)**.** *For a given constant $\gamma > 0$, the following assertions hold:*

$$\sigma_{\min}\left(\nabla^2_{yy}\mathcal{L}(x_*, y_*)\right) \;\geq\; \gamma \qquad \textit{and} \qquad \sigma_{\min}\left(\nabla^2_{yy}\mathcal{L}^{\mathcal{V}_k}(x_*, y_*)\right) \geq \gamma.$$

We start our derivation of the rate of convergence with a lemma that asserts that the pseudospectral abscissa functions of the full and the reduced problems attained at a unique smooth optimizer (i.e., a unique right-most point in pseudospectra, that is three times differentiable with respect to x) around x_* under Assumptions 4.1 and 4.2. We omit its proof, as the proof of its part **(i)** is similar to [15, Proposition 2.9], and proofs of parts **(ii)**-**(iii)** to [21, Lemma 15].

Lemma 4.3. *Suppose that Assumption 4.1 and 4.2 hold. Then, there exist* $\nu_x, \nu_z, \nu_\mu > 0$ *independent of* k *such that* $\mathcal{B}(x_*, \nu_x) \subseteq \underline{\Omega}$ *that satisfy the following:*

(i) *The singular value functions* $\sigma(x,z)$ *and* $\sigma^{\mathcal{V}_k}(x,z)$ *are simple, and their first three derivatives in absolute value are bounded above by constants uniformly for all* $x \in \mathcal{B}(x_*, \nu_x)$ *and* $z \in \mathcal{B}(z_*, \nu_z)$*, where the constants are independent of* k*.*

(ii) *There exists a unique three times differentiable function* $x \in \mathcal{B}(x_*, \nu_x) \mapsto y(x) = (z(x), \mu(x)) \in \mathcal{B}(z_*, \nu_z) \times \mathcal{B}(\mu_*, \nu_\mu)$ *such that* $\alpha_\epsilon(x) = \mathrm{Re}(z(x))$*, as well as*

$$\nabla_y \mathcal{L}(x, y(x)) \;=\; 0 \qquad \textit{and} \qquad \sigma_{\min}(\nabla^2_{yy} \mathcal{L}(x, y(x))) \;\geq\; \gamma/2 \tag{4.1}$$

for all $x \in \mathcal{B}(x_*, \nu_x)$*.*

(iii) *There exists a unique three times differentiable function* $x \in \mathcal{B}(x_*, \nu_x) \mapsto y^{\mathcal{V}_k}(x) = (z^{\mathcal{V}_k}(x), \mu^{\mathcal{V}_k}(x)) \in \mathcal{B}(z_*, \nu_z) \times \mathcal{B}(\mu_*, \nu_\mu)$ *such that* $\alpha^{\mathcal{V}_k}_\epsilon(x) = \mathrm{Re}(z^{\mathcal{V}_k}(x))$*, and*

$$\nabla_y \mathcal{L}^{\mathcal{V}_k}(x, y^{\mathcal{V}_k}(x)) \;=\; 0 \qquad \textit{and} \qquad \sigma_{\min}(\nabla^2_{yy} \mathcal{L}^{\mathcal{V}_k}(x, y^{\mathcal{V}_k}(x))) \;\geq\; \gamma/2 \tag{4.2}$$

for all $x \in \mathcal{B}(x_*, \nu_x)$*.*

For the main rate-of-convergence result, as stated formally below, we also assume that the angle between the left and right singular vectors of $\sigma^{\mathcal{V}_k}(x, z^{\mathcal{V}_k}(x))$ obeys a certain bound on $\mathcal{B}(x_*, \nu_x)$, the interval in Lemma 4.3. This is similar to Assumption 3.2 in the global convergence proof. To this end, letting $u(x)$ and $v(x)$ be a pair of consistent unit left and right singular vectors corresponding to $\sigma(x, z(x))$, following the arguments in Section 2.2, in particular by an application of the first order optimality conditions to the constrained optimization characterization in (2.3) of $\alpha_\epsilon(x)$, we have $u(x)^* v(x) < 0$ for all $x \in \mathcal{B}(x_*, \nu_x)$ so that

$$\underline{\beta} := \max_{x \in \mathcal{B}(x_*, \nu_x)} u(x)^* v(x) \;<\; 0. \tag{4.3}$$

Similarly, as argued in Section 3, we have $u^{\mathcal{V}_k}(x)^* V_k v^{\mathcal{V}_k}(x) < 0$ for all $x \in \mathcal{B}(x_*, \nu_x)$, where $u^{\mathcal{V}_k}(x)$ and $v^{\mathcal{V}_k}(x)$ denote a consistent pair of unit left and right singular vectors corresponding to $\sigma^{\mathcal{V}_k}(x, z^{\mathcal{V}_k}(x))$.

Assumption 4.4. *For a given constant* β *small enough in absolute value and such that* $\underline{\beta} \leq \beta < 0$*, the subspace* $\mathcal{V}_k$ *is such that*

$$\beta \;\geq\; \max_{x \in \mathcal{B}(x_*, \nu_x)} u^{\mathcal{V}_k}(x)^* V_k v^{\mathcal{V}_k}(x),$$

where $\mathcal{B}(x_*, \nu_x)$ *is as in Lemma 4.3.*

The next result concerns the uniform boundedness of the derivatives of $\alpha_\epsilon(x)$ and $\alpha_\epsilon^{\mathcal{V}_k}(x)$.

Lemma 4.5. *Suppose that the conditions of Assumptions 4.1, 4.2 and 4.4 hold. There exists ν_x independent of k such that $\mathcal{B}(x_*, \nu_x) \subseteq \underline{\Omega}$, and the following assertions hold:*

- **(i)** *The pseudospectral functions $\alpha_\epsilon(x)$ and $\alpha_\epsilon^{\mathcal{V}_k}(x)$ are three-times differentiable for all $x \in \mathcal{B}(x_*, \nu_x)$.*
- **(ii)** *The first three derivatives of $\alpha_\epsilon(x)$ and $\alpha_\epsilon^{\mathcal{V}_k}(x)$ in absolute value are bounded above by constants uniformly for all $x \in \mathcal{B}(x_*, \nu_x)$, where the constants are independent of k.*

Proof. The three times differentiability of $\alpha_\epsilon(x)$ and $\alpha_\epsilon^{\mathcal{V}_k}(x)$ uniformly in an interval $\mathcal{B}(x_*, \nu_x)$ for some ν_x that is independent of k is immediate from Lemma 4.3.

The proof of boundedness of the first derivative of $\alpha_\epsilon(x)$ is similar to that in Lemma 3.3 for $\alpha_\epsilon^{\mathcal{V}_k}(x)$. In particular, noting $u(x)^* v(x) \leq \beta < 0$ for $x \in \mathcal{B}(x_*, \nu_x)$ due to (4.3), for any $\widetilde{x} \in \mathcal{B}(x_*, \nu_x)$, we have

$$\alpha_\epsilon'(\widetilde{x}) \;=\; \mathrm{Re}\left(\sum_{\ell=1}^{\kappa} \frac{\mathrm{d}f_\ell(\widetilde{x})}{\mathrm{d}x} \frac{u(\widetilde{x})^* A_\ell v(\widetilde{x})}{u(\widetilde{x})^* v(\widetilde{x})} \right) \quad \Longrightarrow \quad \left|\alpha_\epsilon'(\widetilde{x})\right| \;\leq\; \sum_{\ell=1}^{\kappa} \left| \frac{\mathrm{d}f_\ell(\widetilde{x})}{\mathrm{d}x} \right| \frac{\|A_\ell\|_2}{|\beta|}$$

implying the boundedness of $|\alpha_\epsilon'(x)|$ on $\mathcal{B}(x_*, \nu_x)$, as f_ℓ is real analytic for $\ell = 1, \ldots, \kappa$. As for the second derivatives of $\alpha_\epsilon(x)$, first observe

$$\mu(x) = -\frac{1}{u(x)^* v(x)} \;\leq\; \frac{1}{-\beta} \tag{4.4}$$

for all $x \in \mathcal{B}(x_*, \nu_x)$. Moreover, from the continuity of the singular values and Assumption 4.2, we have

$$\sigma_{\min}(\nabla_{yy}^2 \mathcal{L}(x, y(x))) \;\geq\; \gamma/2 \tag{4.5}$$

for all $x \in \mathcal{B}(x_*, \nu_x)$, where γ is as in Assumption 4.2, by choosing ν_x smaller if necessary. From formula (2.12) for the second derivative of $\alpha_\epsilon(x)$, we have

$$\left|\alpha_\epsilon''(x)\right| \;\leq\; |\mathcal{L}_{xx}(x, y(x))| + \left\|\nabla_{xy}^2 \mathcal{L}(x, y(x))\right\|_2^2 \, \|[\nabla_{yy}^2 \mathcal{L}(x, y(x))]^{-1}\|_2, \tag{4.6}$$

where $\mathcal{L}_{xx}(x, y)$ is the second derivative of $\mathcal{L}(x, y)$ with respect to x. Now, since the derivatives $\mathcal{L}_{xx}(x, y(x))$ and $\nabla_{xy}^2 \mathcal{L}(x, y(x))$ can be expressed fully in terms of $\mu(x)$ and the partial derivatives of $\sigma(x, y(x))$, the result follows from (4.4), (4.5) and part **(i)** of Lemma 4.3.

The uniform boundedness of the third derivatives of $\alpha_\epsilon(x)$, and the boundedness of the derivatives of the reduced pseduospectral abscissa function can be shown in a similar way. □

Interpolation properties between $\alpha_\epsilon(x)$ and $\alpha_\epsilon^{\mathcal{V}_k}(x)$ and their first derivatives hold at $x^{(k)}$. Even if these interpolation properties do not extend to the second derivatives, the second derivatives must be close at $x^{(k)}$ as shown next.

Lemma 4.6 (Proximity of the Second Derivatives)**.** *Suppose that Assumptions 4.1, 4.2 and 4.4 are satisfied. There exists a constant $C > 0$ independent of k such that*

$$\left|\alpha_\epsilon^{''}(x^{(k)}) - [\alpha_\epsilon^{\mathcal{V}_k}]''(x^{(k)})\right| \;\leq\; C\left|x^{(k)} - x^{(k-1)}\right|. \tag{4.7}$$

Proof. We assume without loss of generality that $x^{(k-1)}$, $x^{(k)}$ are strictly inside $\mathcal{B}(x_*, \nu_x)$ (i.e., the interval in Lemma 4.5), where $\alpha_\epsilon(x)$ and $\alpha_\epsilon^{\mathcal{V}_k}(x)$ are differentiable with uniform bounds on their first three derivatives independent of k.

Setting $h := x^{(k-1)} - x^{(k)}$, let us introduce the functions

$$l(t) := \alpha_\epsilon(x^{(k)} + th) \text{ and } l_k(t) := \alpha_\epsilon^{\mathcal{V}_k}(x^{(k)} + th) \tag{4.8}$$

for $t \in [0, 1]$. By applying the Taylor's theorem with third order remainder to $l(t)$ and $l_k(t)$ on the interval $(0, 1)$, we obtain

$$\begin{aligned} l(1) \;&=\; l(0) + l'(0) + l''(0)/2 + l'''(\eta)/6, \\ l_k(1) \;&=\; l_k(0) + l_k'(0) + l_k''(0)/2 + l_k'''(\eta_k)/6 \end{aligned}$$

for some constants $\eta, \eta_k \in (0, 1)$. Notice that $l(1) = l_k(1)$, $l(0) = l_k(0)$, and $l'(0) = l_k'(0)$ due to Lemma 2.4. Consequently,

$$\begin{aligned} \frac{\alpha_\epsilon''(x^{(k)})h^2 - [\alpha^{\mathcal{V}_k}]''(x^{(k)})h^2}{2} \;&=\; \frac{l''(0) - l_k''(0)}{2} \\ &=\; \frac{l_k'''(\eta_k) - l'''(\eta)}{6} \\ &=\; \frac{[\alpha^{\mathcal{V}_k}]'''(x^{(k)} + \eta_k h)h^3 - \alpha_\epsilon'''(x^{(k)} + \eta h)h^3}{6} \end{aligned} \tag{4.9}$$

As $x^{(k)} + \eta_k h, x^{(k)} + \eta h \in \mathcal{B}(x_*, \nu_x)$, the third derivatives $[\alpha^{\mathcal{V}_k}]'''(x^{(k)} + \eta_k h)$ and $\alpha_\epsilon'''(x^{(k)} + \eta h)$ on the righthand side of (4.9) in absolute value are bounded from above by a uniform constant U. Hence, we deduce from (4.9) that

$$\left|\alpha_\epsilon^{''}(x^{(k)}) - [\alpha_\epsilon^{\mathcal{V}_k}]''(x^{(k)})\right| \;\leq\; \frac{2U}{3}h \;=\; \frac{2U}{3}\left|x^{(k)} - x^{(k-1)}\right|$$

as desired. □

Now we are ready to state and prove the main result of this section, that is the superlinear convergence result regarding the iterates of Algorithm 1.

Theorem 4.7 (Superlinear Convergence)**.** *Suppose that Assumptions 4.1, 4.2 and 4.4 hold. Additionally, assume that $\alpha_\epsilon''(x_*) \neq 0$. Then, there exists a constant $\Upsilon > 0$ independent of k such that*

$$\frac{\left|x^{(k+1)} - x_*\right|}{\left|x^{(k)} - x_*\right| \max\{\left|x^{(k)} - x_*\right|, \left|x^{(k-1)} - x_*\right|\}} \;\leq\; \Upsilon. \tag{4.10}$$

Proof. Let ν_x be as in Lemma 4.5 so that $\alpha_\epsilon(x)$ and $\alpha_\epsilon^{\mathcal{V}_k}(x)$ are differentiable, indeed $\alpha_\epsilon''(x)$ and $[\alpha_\epsilon^{\mathcal{V}_k}]''(x)$ are Lipschitz continuous, inside $\mathcal{B}(x_*, \nu_x)$. Moreover, without loss of generality, assume $x^{(k-1)}, x^{(k)}, x^{(k+1)} \in \mathcal{B}(x_*, \nu_x)$.

By assumption $\alpha_\epsilon''(x_*) \neq 0$, so, by employing Lemma 4.6, we can assume $x^{(k-1)}$, $x^{(k)}$ are close enough so that $[\alpha_\epsilon^{\mathcal{V}_k}]''(x_*) \neq 0$. Indeed, if necessary by choosing ν_x even smaller, we can assume $\alpha_\epsilon''(x) \neq 0$, $[\alpha_\epsilon^{\mathcal{V}_k}]''(x) \neq 0$ for all $x \in \mathcal{B}(x_*, \nu_x)$.

Now, by the Taylor's theorem with integral remainder

$$0 \;=\; \alpha_\epsilon'(x_*) \;=\; \alpha_\epsilon'(x^{(k)}) + \int_0^1 \alpha_\epsilon''(x^{(k)} + \omega(x_* - x^{(k)}))(x_* - x^{(k)})\,\mathrm{d}\omega.$$

Exploiting the interpolation property $\alpha_\epsilon'(x^{(k)}) = [\alpha_\epsilon^{\mathcal{V}_k}]'(x^{(k)})$, and left-multiplying both sides of the last equation by $[\alpha_\epsilon''(x^{(k)})]^{-1}$, we deduce

$$\begin{aligned} 0 \;=\; & [\alpha_\epsilon''(x^{(k)})]^{-1} \cdot [\alpha_\epsilon^{\mathcal{V}_k}]'(x^{(k)}) \;+\; (x_* - x^{(k)}) \;+ \\ & [\alpha_\epsilon''(x^{(k)})]^{-1} \int_0^1 \left[\alpha_\epsilon''(x^{(k)} + \omega(x_* - x^{(k)})) - \alpha_\epsilon''(x^{(k)})\right](x_* - x^{(k)})\,\mathrm{d}\omega. \end{aligned} \tag{4.11}$$

As $[\alpha_\epsilon^{\mathcal{V}_k}]'(x^{(k+1)}) = 0$, a second order Taylor expansion of $[\alpha_\epsilon^{\mathcal{V}_k}]'(x)$ about $x^{(k)}$ gives rise to

$$0 \;=\; [\alpha_\epsilon^{\mathcal{V}_k}]''(x^{(k)})]^{-1}[\alpha_\epsilon^{\mathcal{V}_k}]'(x^{(k)}) \;+\; (x^{(k+1)} - x^{(k)}) \;+\; \mathcal{O}(\left|x^{(k+1)} - x^{(k)}\right|^2).$$

Subtracting the last equality from (4.11) side-by-side yields

$$\begin{aligned} 0 \;=\; & (x_* - x^{(k+1)}) \\ & +\; \left\{[\alpha_\epsilon''(x^{(k)})]^{-1} - [\alpha_\epsilon^{\mathcal{V}_k}]''(x^{(k)})]^{-1}\right\}\alpha_\epsilon'(x^{(k)}) \;+\; \mathcal{O}(\left|x^{(k+1)} - x^{(k)}\right|^2) \\ & +\; [\alpha_\epsilon''(x^{(k)})]^{-1} \int_0^1 \left[\alpha_\epsilon''(x^{(k)} + \omega(x_* - x^{(k)})) - \alpha_\epsilon''(x^{(k)})\right](x_* - x^{(k)})\,\mathrm{d}\omega. \end{aligned}$$

Finally, we take the absolute values and employ the triangle inequality in the last equation to obtain

$$\begin{aligned} \left|x^{(k+1)} - x_*\right| \leq\; & \left|[\alpha_\epsilon''(x^{(k)})]^{-1} - \left[[\alpha_\epsilon^{\mathcal{V}_k}]''(x^{(k)})\right]^{-1}\right|\left|\alpha_\epsilon'(x^{(k)})\right| + \mathcal{O}(|x^{(k+1)} - x^{(k)}|^2) \\ & + \left|[\alpha_\epsilon''(x^{(k)})]^{-1}\right| \int_0^1 \left|\alpha_\epsilon''(x^{(k)} + \omega(x_* - x^{(k)})) - \alpha_\epsilon''(x^{(k)})\right| \cdot \left|x_* - x^{(k)}\right| \mathrm{d}\omega. \end{aligned} \tag{4.12}$$

Now, let us consider the right-hand side of (4.12). It follows from Lemma 4.6, in particular (4.7), that $\left|[\alpha_\epsilon''(x^{(k)})]^{-1} - \left[[\alpha_\epsilon^{\mathcal{V}_k}]''(x^{(k)})\right]^{-1}\right| \leq c|x^{(k)} - x^{(k-1)}|$ for some constant c independent of k. Due to this observation and $\left|\alpha_\epsilon'(x^{(k)})\right| = \mathcal{O}(\left|x^{(k)} - x_*\right|)$, which can be seen from the Taylor expansion of $\alpha_\epsilon'(x)$ about $x^{(k)}$ and exploiting $\alpha_\epsilon'(x_*) = 0$, the first term on the right-hand side of (4.12) is bounded above by $\mathcal{O}(|x^{(k)} - x_*| \cdot |x^{(k)} - x^{(k-1)}|)$. The last term is bounded above by $\mathcal{O}(|x^{(k)} - x_*|^2)$ as $\left|[\alpha_\epsilon''(x^{(k)})]^{-1}\right|$ is bounded and $\alpha_\epsilon''(x)$ is Lipschitz continous in $\mathcal{B}(x_*, \nu_x)$. These observations for the right-hand side of (4.12) lead us to

$$|x^{(k+1)} - x_*| \leq \mathcal{O}(|x^{(k)} - x_*| \cdot |x^{(k)} - x^{(k-1)}|) + \mathcal{O}(|x^{(k+1)} - x^{(k)}|^2) + \mathcal{O}(|x^{(k)} - x_*|^2).$$

The desired result (4.10) follows from $|x^{(k)} - x^{(k-1)}| \leq 2\max\{|x^{(k)} - x_*|, |x^{(k-1)} - x_*|\}$, as well as $|x^{(k+1)} - x^{(k)}|^2 \leq 2(|x^{(k)} - x_*|^2 + |x^{(k+1)} - x_*|^2)$. □

5 Numerical Results

We have implemented the proposed subspace framework to minimize $\alpha_\epsilon(x)$ in Matlab. In this section, we perform numerical experiments using this implementation in Matlab 2020b on an an iMac with Mac OS 12.1 operating system, Intel® Core™ i5-9600K CPU and 32GB RAM. Before focusing on numerical experiments on synthetic examples and benchmark examples taken from the *COMPl$_e$ib* collection [19] in Sections 5.2 and 5.3, we first summarize a few important implementation details in the next subsection.

5.1 Implementation Details

5.1.1 Initial Interpolation Points

We choose the initial interpolation points $x_1, \ldots, x_\eta$ in line 1 of the subspace framework as follows unless they are made available explicitly. If there are multiple optimization parameters, then $x_1, \ldots, x_\eta$ are selected randomly in $\underline{\Omega}$. Otherwise, if there is only one optimization parameter, we choose them as equally-spaced points in $\underline{\Omega} := [L, U]$, i.e., $x_j := L + (j-1)\frac{U-L}{\eta-1}$ for $j = 1, \ldots, \eta$. In all of the experiments below, unless otherwise stated, the number of initial interpolation points is $\eta = 10$.

5.1.2 Termination Condition

In practice we terminate when the gap between the optimal values for the reduced optimization problems in two consecutive iterations is less than a prescribed tolerance. Formally, given a tolerance `tol`, we terminate at iteration $k \geq 3$ if

$$\alpha_\epsilon^{\mathcal{V}_{k-1}}(x^{(k)}) - \alpha_\epsilon^{\mathcal{V}_{k-2}}(x^{(k-1)}) \;<\; \texttt{tol}. \tag{5.1}$$

In practice we use $\texttt{tol} = 10^{-7}$. Additionally, we terminate if the number of subspace iterations exceeds a certain amount, which is 30 iterations in all of the numerical experiments below. But this second condition is never needed for the examples here as the condition in (5.1) with $\texttt{tol} = 10^{-7}$ is always met in fewer than 10 iterations.

5.1.3 Solution of the Reduced Optimization Problems

The minimization of $\alpha_\epsilon^{\mathcal{V}_{k-1}}(x)$ over $x \in \underline{\Omega}$ in line 6 of the proposed subspace framework is performed using either "eigopt" [22] or "GRANSO" [12].

If there is only one parameter, then we rely on "eigopt" as it converges globally provided a lower bound γ on the second derivatives of $\alpha_\epsilon^{\mathcal{V}_{k-1}}(x)$ where it is differentiable is chosen small enough. In all of the experiments below depending on one parameter, we set $\gamma = -400$, which seems to work well.

On the other hand, if there are multiple parameters, we depend on "GRANSO" for the solution of these reduced optimization problems. The reason is "eigopt" is usually slow when there are multiple optimization parameters, and the number of function evaluations needed to satisfy a certain accuracy increases quite rapidly with respect to the accuracy required. "GRANSO", on the other hand, can solve multiple-parameter problems quite efficiently. Yet, as it is based on BFGS and quasi-Newton methods, it converges locally. For the numerical examples depending on multiple parameters that we report below, this local convergence issue does not cause any problem.

The objective for the reduced optimization problem at the kth subspace iteration is $\alpha_\epsilon^{\mathcal{V}_{k-1}}(x)$ meaning $\alpha_\epsilon^{\mathcal{V}_{k-1}}(x)$ needs to be computed at several x. This requires finding the rightmost point of the rectangular pseudospectrum

$$\Lambda_\epsilon(A^{V_{k-1}}(x)) \;=\; \{z \in \mathbb{C} \mid \sigma_{\min}(A^{V_{k-1}}(x) - zV_{k-1}) \leq \epsilon\}.$$

For this purpose, we adopt the approach proposed in [17, Section 5.2], which is an extension of the quadratically convergent criss-cross algorithm for the ϵ-pseudospectral abscissa of a square matrix [10] to rectangular pencils. We remark that $A^{V_{k-1}}(x)$ and V_{k-1} are of size $n \times (k-1)$. Hence, at first look they are not small scale. Yet, as suggested in [17, Section 5.2], a reduced QR factorization

$$\begin{bmatrix} V_{k-1} & A^{V_{k-1}}(x) \end{bmatrix} \;=\; Q \begin{bmatrix} \widetilde{B} & \widetilde{A} \end{bmatrix}$$

yields $\widetilde{A}, \widetilde{B} \in \mathbb{C}^{(2k-2)\times(k-1)}$ such that

$$\Lambda_\epsilon(A^{V_{k-1}}(x)) \;=\; \Lambda_\epsilon(\widetilde{A}, \widetilde{B}) := \{z \in \mathbb{C} \mid \sigma_{\min}(\widetilde{A} - z\widetilde{B}) \leq \epsilon\}.$$

To summarize, to compute $\alpha_\epsilon^{\mathcal{V}_{k-1}}(x)$ the rightmost point of $\Lambda_\epsilon(\widetilde{A}, \widetilde{B})$ depending on small matrices is found cheaply using a criss-cross algorithm as in [10].

One subtle issue is that it is essential for the criss-cross algorithm to locate the rightmost point in $\Lambda_\epsilon(\widetilde{A},\widetilde{B})$ to start iterating initially from a point in the rightmost component of $\Lambda_\epsilon(\widetilde{A},\widetilde{B})$. As a heuristic, it is proposed in [17, Section 5.2] to initialize the criss-cross algorithm with the rightmost eigenvalue λ of the pencil $L(s) = \widetilde{A}_1 - s\widetilde{B}_1$ satisfying $\sigma_{\min}(\widetilde{A} - \lambda\widetilde{B}) \leq \epsilon$, where $\widetilde{A}_1, \widetilde{B}_1$ denote the upper $(k-1)\times(k-1)$ parts of $\widetilde{A},\widetilde{B}$. The difficulty is that there may not be such an eigenvalue of $L(\cdot)$ satisfying $\sigma_{\min}(\widetilde{A} - \lambda\widetilde{B}) \leq \epsilon$. As a safeguard, we additionally choose equally-spaced point $y_1,\ldots,y_k$ in a prescribed subinterval (for the examples below $[-2\mathrm{i}, 2\mathrm{i}]$) of the imaginary axis, then find the largest x such that $\sigma_{\min}(\widetilde{A} - (x + \mathrm{i}y_j)) = \epsilon$ for $j = 1,\ldots,\kappa$. Such largest x corresponding to y_j, call it $x_{R,j}$, for $j = 1,\ldots,\kappa$ is given by the imaginary part of one of the purely imaginary eigenvalues of the pencil

$$\mathcal{L}(s) \;=\; \begin{bmatrix} -y_j\widetilde{B}^* + \mathrm{i}\widetilde{A}^* & \epsilon I \\ -\epsilon I & y_j\widetilde{B} + \mathrm{i}\widetilde{A} \end{bmatrix} \;-\; s\begin{bmatrix} \widetilde{B}^* & 0 \\ 0 & \widetilde{B} \end{bmatrix}$$

with the largest imaginary part [17, Lemma 5.1]. To conclude, we initialize the criss-cross algorithm for rectangular pencils with the rightmost point among the points $x_{R,1},\ldots,x_{R,\kappa}$ and the point produced by the heuristic described above from [17, Section 5.2].

5.1.4 Finding the Rightmost Point of $\Lambda_\epsilon(x)$

The computation of the rightmost point of $\Lambda_\epsilon(x_j^{(1)})$ for $j = 1,2,\ldots,\eta$, and $\Lambda_\epsilon(x^{(j)})$ for $j = 2,\ldots,k$ is required by the subspace framework to form the subspace $\mathcal{V}_k$. This is usually the most intensive computational task, as it involves finding the rightmost point in the ϵ-pseudospectral abscissa of the large matrix $A(x)$ at various x. For this purpose, we either employ the original criss-cross algorithm for the computation of the pseudospectral abscissa [10] if the size of $A(x)$ is less than or equal to a prescribed amount, or the subspace framework in [17] if the size of $A(x)$ is larger than the prescribed amount. In the experiments below, this prescribed size is chosen as 1000.

5.2 Synthetic Examples Depending on One Parameter

We start by conducting numerical experiments with the synthetic examples used in [21, Section 7], that are publicly available[1]. Each example here concerns a matrix-valued function of the form $A(x) = A + xbc^T$ over $x \in \mathbb{R}$ in a prescribed interval for given $A \in \mathbb{R}^{n\times n}$, $b, c \in \mathbb{R}^n$.

Recall that the distance to instability of a matrix M is defined by

$$\mathcal{D}(M) \;:=\; \inf\{\, \|\Delta\|_2 \mid \Delta \in \mathbb{R}^{n\times n} \text{ s.t. } \Lambda(M+\Delta) \cap \mathbb{C}^+ \neq \emptyset \,\},$$

[1] `http://home.ku.edu.tr/~emengi/software/max_di/Data_&_Updates.html`

where $\mathbb{C}^+$ denotes the closed right-half of the complex plane. It follows from the definition of $\mathcal{D}(M)$ that, for every $\epsilon > 0$, we have

$$\mathcal{D}(M) \leq \epsilon \quad \Longleftrightarrow \quad \alpha_\epsilon(M) \geq 0. \tag{5.2}$$

In [21, Section 7], the distance to instability for each example $A(x)$ is maximized over x in a prescribed interval $\mathcal{I}$, and the maximal value of the distance to instability, as well as the maximizer are reported. Letting x_* be the global maximizer of $\mathcal{D}(A(x))$ over $x \in \mathcal{I}$, and $\mathcal{D}_* := \mathcal{D}(A(x_*))$, we deduce from (5.2) that

$$\min_{x \in \mathcal{I}} \alpha_\epsilon(A(x)) \;=\; \alpha_\epsilon(A(x_*)) \;=\; 0$$

for $\epsilon = \mathcal{D}_*$.

We illustrate the proposed subspace framework in Figure 1 to minimize $\alpha_\epsilon(x) := \alpha_\epsilon(A(x))$ for the example $A(x)$ in [21, Section 7] of order $n = 400$ over $x \in [-0.3, 0.2]$, and with $\epsilon = 0.12870882 \approx \mathcal{D}(A(x_*))$. The initial subspace $\mathcal{V}_1$ is chosen as the 2-dimensional subspace so that Hermite interpolation is attained between $\alpha_\epsilon(x)$ and $\alpha_\epsilon^{\mathcal{V}_1}(x)$ at $x = -0.2, 0.1$. The reduced function $\alpha_\epsilon^{\mathcal{V}_1}(x)$ is minimized over $x \in [-0.3, 0.2]$; its minimizer turns out to be $x^{(2)} = 0.0480905$. Then the subspace $\mathcal{V}_1$ is expanded into $\mathcal{V}_2$ so that $\alpha_\epsilon^{\mathcal{V}_2}(x)$ interpolates $\alpha_\epsilon(x)$ at $x = x^{(2)}$. The global minimizer of $\alpha_\epsilon^{\mathcal{V}_2}(x)$ is already quite close to the actual minimizer of $\alpha_\epsilon(x)$ as can be seen in Figure 1. The subspace framework on this example terminates after 4 subspace iterations. The iterates of the subspace framework are given in Table 1. It appears from the second and third columns of this table that $x^{(k)}$ converges to the minimizer x_* of $\alpha_\epsilon(x)$ at a superlinear rate, consistent with the superlinear convergence assertion of Theorem 4.7. Also, as expected, the globally smallest value of $\alpha_\epsilon(A(x))$ is about 0, and the computed global minimizer $x_* = -0.1056336$ is about the same as the global maximizer of $\mathcal{D}(A(x))$ reported in [21, Section 7].

Table 1: The iterates of the subspace framework to minimize $\alpha_\epsilon(A(x))$ for the example $A(x)$ in [21, Section 7] of order $n = 400$, and with $\epsilon \approx \mathcal{D}(A(x_*))$.

k	$x^{(k+1)}$	$\lvert x^{(k+1)} - x_* \rvert$	$\alpha_\epsilon^{\mathcal{V}_k}(x^{(k+1)})$
1	$-\underline{0}.0480905$	0.0575430	$-\underline{0.0}137274$
2	$-\underline{0.10}88387$	0.0032051	$-\underline{0.0000}501$
3	$-\underline{0.1056332}$	0.0000003	$-\underline{0.0000000}$
4	$-\underline{0.1056336}$	0.0000000	$-\underline{0.0000000}$

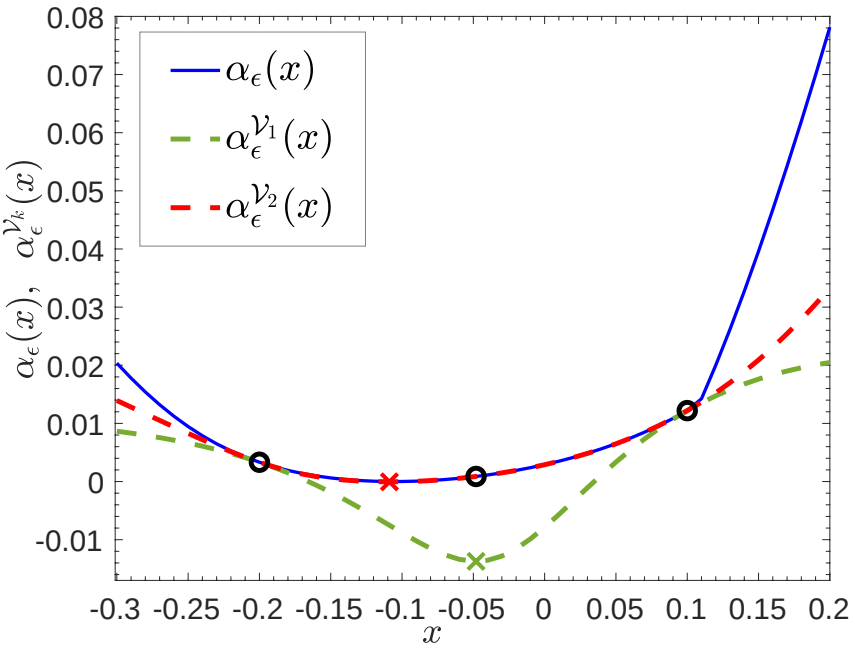


Figure 1: The progress of the subspace framework to minimize $\alpha_\epsilon(A(x))$ over $x \in [-0.3, 0.2]$ for the example $A(x)$ in [21, Section 7] of order $n = 400$, and with $\epsilon \approx \mathcal{D}(A(x_*))$. The black circles mark the interpolation points, whereas the crosses mark the global minimizers of $\alpha_\epsilon^{\mathcal{V}_1}(x)$ and $\alpha_\epsilon^{\mathcal{V}_2}(x)$.

The progress of the algorithm on this example can also be traced by looking at the pseudospectra. When $\alpha_\epsilon^{\mathcal{V}_1}(x)$ is minimized over x, it yields $\Lambda_\epsilon^{\mathcal{V}_1}(x^{(2)})$, the innermost dashed-dotted blue curve in the leftmost plot in Figure 2. Then the subspace $\mathcal{V}_1$ is expanded into $\mathcal{V}_2$ so that the rightmost point of $\Lambda_\epsilon^{\mathcal{V}_2}(x^{(2)})$ is the same as the rightmost point of $\Lambda_\epsilon(x^{(2)})$, i.e., the rightmost point of the solid orange curve representing the boundary of $\Lambda_\epsilon^{\mathcal{V}_2}(x^{(2)})$ is the same as the rightmost point of the dashed red curve representing the boundary of $\Lambda_\epsilon(x^{(2)})$ in the leftmost plot in Figure 2. Next $\alpha_\epsilon^{\mathcal{V}_2}(x)$ is minimized over x yielding $\Lambda_\epsilon^{\mathcal{V}_2}(x^{(3)})$, the dashed-dotted blue curve in the middle plot in Figure 2. Observe that $\Lambda_\epsilon^{\mathcal{V}_2}(x^{(3)})$ and $\Lambda_\epsilon^{\mathcal{V}_2}(x^{(2)})$ depicted by the orange curve on the left are quite similar as $x^{(2)}$ and $x^{(3)}$ are close. Now $\mathcal{V}_2$ is expanded into $\mathcal{V}_3$ so as to ensure that the rightmost points of $\Lambda_\epsilon^{\mathcal{V}_3}(x^{(3)})$ and $\Lambda_\epsilon(x^{(3)})$, the rightmost points of the solid orange and dashed red curves in the middle plot in Figure 2, are the same. Minimizing $\Lambda_\epsilon^{\mathcal{V}_3}(x)$ over x yields $\Lambda_\epsilon^{\mathcal{V}_3}(x^{(4)})$, the dashed-dotted curve in the rightmost plot in Figure 2 and so on. Note that, indeed, in these three plots the boundaries of $\Lambda_\epsilon^{\mathcal{V}_k}(x^{(k)})$ and $\Lambda_\epsilon(x^{(k)})$ are tangent to each other at their rightmost

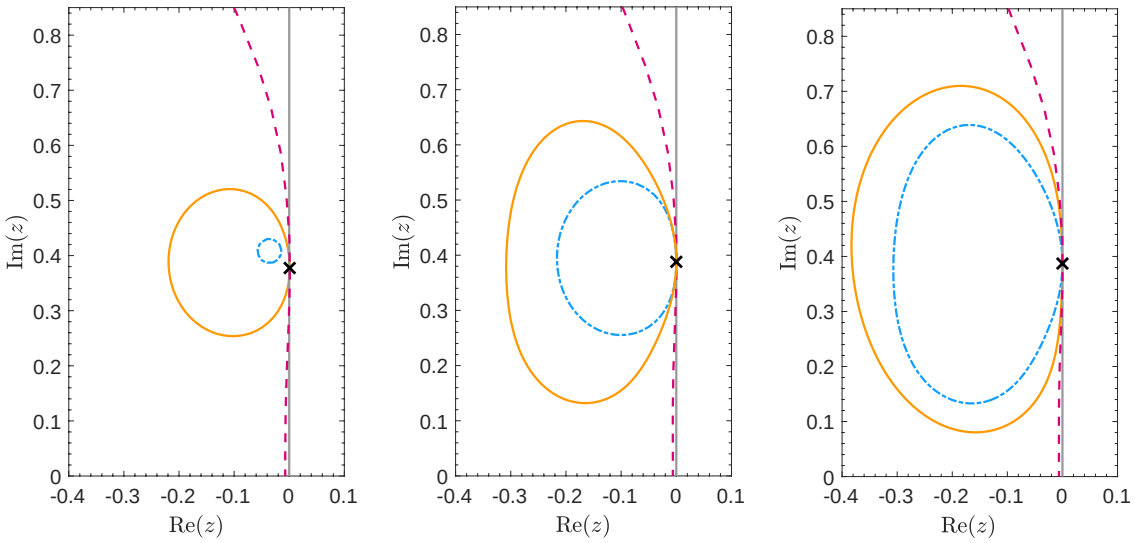


Figure 2: The subspace framework on the example $A(x)$ in [21, Section 7] of order $n = 400$ with $\epsilon \approx \mathcal{D}(A(x_*))$ is traced by looking at the ϵ-pseudospectrum. The dashed-dotted blue, solid orange, dashed red curves represent the boundaries of $\Lambda_\epsilon^{\mathcal{V}_k}(x^{(k+1)}), \Lambda_\epsilon^{\mathcal{V}_{k+1}}(x^{(k+1)}), \Lambda_\epsilon(x^{(k+1)})$ for $k = 1$ (left plot), $k = 2$ (middle plot), $k = 3$ (right plot). The crosses mark the rightmost points of $\Lambda_\epsilon^{\mathcal{V}_{k+1}}(x^{(k+1)}), \Lambda_\epsilon(x^{(k+1)})$. The ϵ-pseudospectrum $\Lambda_\epsilon(x^{(k)})$ for $k = 2, 3, 4$ are generated using EigTool [26].

points for $k = 2, 3, 4$ from left to right.

We have also performed experiments with the examples from [21, Section 7] of order $n = 200, 400, 800, 1200, 2000$ that concern the maximization of $\mathcal{D}(A(x))$ on the interval $[-3, 3]$, where $A(x) = A + xbc^T$ for given $A^{n\times n}, b, c \in \mathbb{R}^n$. In each case, we have minimized $\alpha_\epsilon(A(x))$ over $x \in [-3, 3]$ using the subspace framework for $\epsilon \approx \mathcal{D}(A(x_*))$ (to be precise for ϵ equal to the reported value of $\mathcal{D}(A(x_*))$ in [21, Section 7], which is eight decimal digit accurate). The results are listed in Table 2. The globally minimal value $\alpha_\epsilon(x_*)$ of $\alpha_\epsilon(x)$ is about 0 for each one of these examples as expected. Moreover, the global minimizers x_* of $\mathcal{D}(A(x))$ listed in the table are close to those reported in [21, Section 7]. The number of iterations until termination for each one of the examples is 3 to 6 indicating quick convergence. As for the runtimes, according to the table the main computational task that contributes to the runtime is the computation of the ϵ-pseudospectral abscissa of $A(x)$, which is required once per iteration as well as to form the initial subspaces. On the other hand, the reduced optimization problem that involves the minimization of $\alpha_\epsilon^{\mathcal{V}_k}(x)$ at the kth subspace

Table 2: The table concerns the application of the subspace framework to the examples from [21, Section 7] of order $n = 200, 400, 800, 1200, 2000$. In each case, $\alpha_\epsilon(A(x))$ is minimized over $x \in [-3, 3]$, where ϵ is the reported largest value of $\mathcal{D}(A(x))$ over $x \in [-3, 3]$. The last three columns list the total runtime (time), the time for the reduced minimization problems (red), the time to compute the rightmost point of $\Lambda_\epsilon(A(x))$ at various x (psa) in seconds.

n	$\epsilon \approx \mathcal{D}(A(x_*))$	x_*	$\alpha_\epsilon(x_*)$	iter	time	red	psa
200	0.01839422	0.09439	0	3	47.5	3.3	43.8
400	0.12870882	−0.10566	0	4	235.7	52.4	181.7
800	0.11545563	−0.05851	0.0000005	4	979.8	57.1	914.1
1200	0.07941192	−0.48625	0	3	616.6	138.8	471.8
2000	0.08436380	−0.04766	0.0000058	6	3210	59.7	3125.1

iteration is relatively cheap to solve. These two computational tasks determine the total runtime.

5.3 Benchmark Examples

Several benchmark examples for the stabilization by static output feedback problem are provided in the *COMPl$_e$ib* collection [19]. The problems that we consider here taken from *COMPl$_e$ib* concern finding a $K \in \mathbb{R}^{m\times p}$ such that $A + BKC$ has all of its eigenvalues in the open left-half of the complex plane for given $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$. Minimizing the ϵ-pseudospectral abscissa of $A + BKC$ over K for a prescribed ϵ has also been suggested in [19] for the robust stabilization of the system.

We focus on three examples. The first one is about the stabilization of a single-input-single-output system so that K is a scalar meaning there is only one parameter to be optimized, whereas the remaining two examples depend on multiple optimization parameters. Especially, the first and third examples involve relatively large matrices so that a direct minimization of the ϵ-pseudospectral abscissa does not seem feasible. In all of the examples, ϵ is chosen as 0.2.

NN18 *($n = 1006$, $m = p = 1$).* This example involves the stabilization of $A + xbc^T$ over $x \in \mathbb{R}$. We minimize $\alpha_\epsilon(x)$ for $\epsilon = 0.2$ over $x \in [-1, 1]$ by employing the subspace framework. The computed global minimizer is $x_* = -1$, and the computed globally smallest ϵ-pseudospectral abscissa is $\alpha_\epsilon(x_*) = -0.9149600$. The correctness of these computed values can be verified by looking at Figure 3. Note that the original matrix A is asymptotically stable, yet its ϵ-pseudospectral abscissa $\alpha_\epsilon(0) = -0.8$ is larger than $\alpha_\epsilon(x_*)$. Hence, optimizing $\alpha_\epsilon(x)$ over x yields a system that is more robustly stable compared to the original system. The subspace framework terminates after 3 subspace iterations, and the iterates generated are listed in Table 3. We also report

the runtimes of the subspace framework in Table 4. Once again the computation of $\alpha_\epsilon(x)$ required several times to form and expand the subspaces dominate the runtime, whereas the minimization of the reduced pseudospectral abscissa functions is computationally much cheaper.

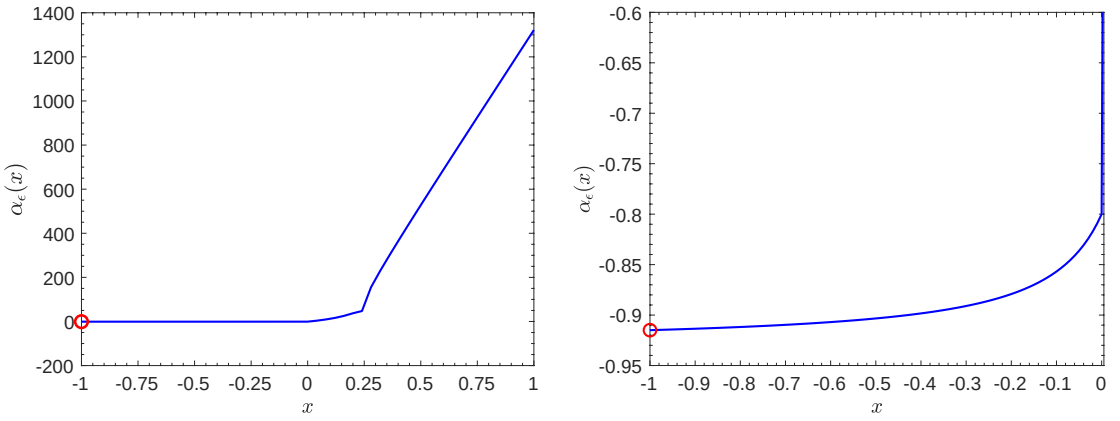


Figure 3: The plots of $\alpha_\epsilon(x)$ as a function of x for the NN18 example in the *COMPl$_e$ib* collection. The right-hand plot is a zoomed version of the left-hand plot focusing on $x \in [-1, 0.02]$.

Table 3: The iterates of the subspace framework to minimize $\alpha_\epsilon(x)$ over $x \in [-1, 1]$ for the NN18 example.

k	$x^{(k+1)}$	$\alpha_\epsilon^{\mathcal{V}_k}(x^{(k+1)})$
1	0.2319048	-1.0680310
2	-1.0000000	-0.9149600
3	-1.0000000	-0.9149600

Table 4: Optimal values for $\alpha_\epsilon(x)$ over $x \in [-1, 1]$ for the NN18 example obtained using the subspace framework, as well as the runtimes for the subspace framework. The last three columns list in seconds the total runtime, the time for the reduced minimization problems, and the time to compute a rightmost point in $\Lambda_\epsilon(x)$ several times.

x_*	$\alpha_\epsilon(x_*)$	$\alpha_\epsilon(0)$	time	red	psa
-1	-0.9149600	-0.8	101.2	9.3	87.8

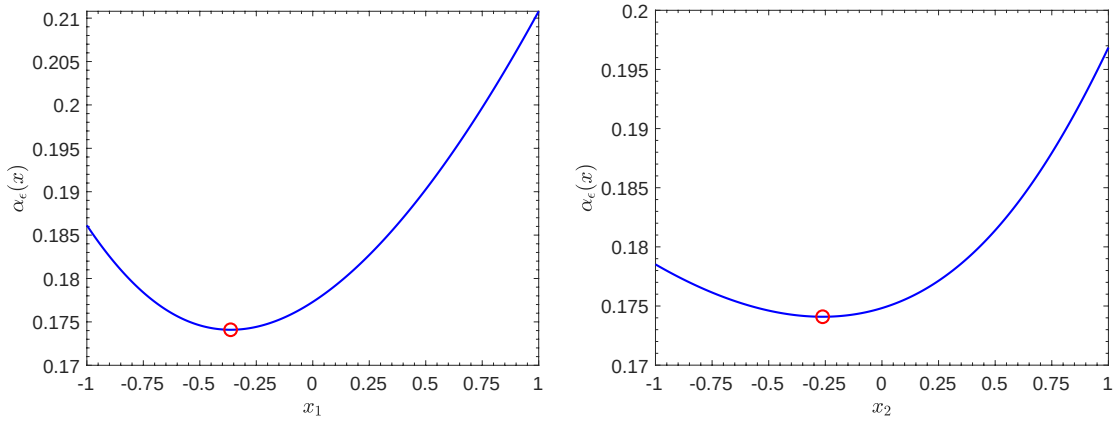


Figure 4: The plots of $\alpha_\epsilon(x)$ for $\epsilon = 0.2$ as a function of x for the HF1 example from the *COMPl$_e$ib* collection near its computed global minimizer $x_* = (x_{*,1}, x_{*,2})$. (Left) The plot of $x_1 \mapsto \alpha_\epsilon(x_1, x_{*,2})$. (Right) The plot of $x_2 \mapsto \alpha_\epsilon(x_{*,1}, x_2)$. The circles mark $(x_{*,1}, \alpha_\epsilon(x_*))$ and $(x_{*,2}, \alpha_\epsilon(x_*))$ on the left and on the right, respectively.

HF1 *($n = 130$, m=1, $p = 2$).* This example concerns the stabilization of $A+BKC$ with respect to $K \in \mathbb{R}^{1\times 2}$ for given $A \in \mathbb{R}^{130\times 130}, B \in \mathbb{R}^{130}, C \in \mathbb{R}^{2\times 130}$. We minimize $\alpha_\epsilon(x)$ using the subspace framework for $\epsilon = 0.2$ and $A(x) = A + x_1 BC(1,:)$ $+x_2\, BC(2,:)$ over $x \in [-1,1] \times [-1,1]$, where $C(j,:)$ denotes the jth row of C. The subspace framework terminates after two subspace iterations with the optimal value of x as $x_* = (-0.36363, -0.26188)$, and the corresponding minimal value of the ϵ-pseudospectral abscissa as $\alpha_\epsilon(x_*) = 0.1749194$. The accuracy of these computed optimal values can be verified from Figure 4. Once again the optimized system is more robustly stable compared to the original system with $\alpha_\epsilon(0) = 0.1810205$. The runtimes in Table 5 again confirm that the total runtime is determined by the time required for the computation of $\alpha_\epsilon(x)$ at several x.

Table 5: Optimal values for $\alpha_\epsilon(x)$ over $x \in [-1,1] \times [-1,1]$ for the HF1 example obtained using the subspace framework, and the runtimes for the subspace framework. The last three columns list the runtimes in seconds as in Table 4.

x_*	$\alpha_\epsilon(x_*)$	$\alpha_\epsilon(0)$	time	red	psa
$(-0.36363, -0.26188)$	0.1740919	0.1810205	11.4	0.5	10.7

HF2D2 *($n = 3796$, $m = 2$, $p = 3$).* This is a large-scale example that arises from a modeling of 2D heat flow [19, Section 3]. A stabilizer $K \in \mathbb{R}^{2\times 3}$ is sought so

that $A + BKC$ is asymptotically stable for given $A \in \mathbb{R}^{3796\times 3796}, B \in \mathbb{R}^{3796\times 2}, C \in \mathbb{R}^{3\times 3796}$. The original matrix A is unstable with spectral abscissa 0.2556862, and ϵ-pseudospectral abscissa for $\epsilon = 0.2$ equal to 0.4625511.

We express $A + BKC$ in the form $A(x) = A + \sum_{j=1}^{6} x_j A_j$, where $x_j = k_{1j}$, $x_{3+j} = k_{2j}$, and $A_j = B(:,1)C(j,:)$, $A_{3+j} = B(:,2)C(j,:)$ for $j = 1,2,3$, while $B(:,1)$ and $B(:,2)$ represent the first and second columns of B. We minimize $\alpha_\epsilon(x)$ for $\epsilon = 0.2$ with the constraints that $x_j \in [-1,1]$ for $j = 1,\ldots,6$. The subspace framework terminates after 4 subspace iterations with optimal x_* corresponding to the following matrix:

$$K_* = \begin{bmatrix} 1 & 0.33494 & 1 \\ 1 & -1 & 1 \end{bmatrix}.$$

It appears from Figure 5 that the computed global minimizer is accurate. The resulting matrix is asymptotically stable, as indeed $\alpha_\epsilon(x_*) = -0.4124020$. As depicted in Figure 6, whereas one of the components of $\Lambda_\epsilon(0)$ is fully on the right-hand side of the complex plane, the rightmost component of $\Lambda_\epsilon(x_*)$ is fully contained in the open left-half of the complex plane. In terms of the runtime, now the reduced optimization problems take more time than the time to compute $\alpha_\epsilon(x)$ as reported in Table 6. Yet, the overall runtime is quite reasonable considering the system at hand is of order 3796, and there are several optimization parameters.

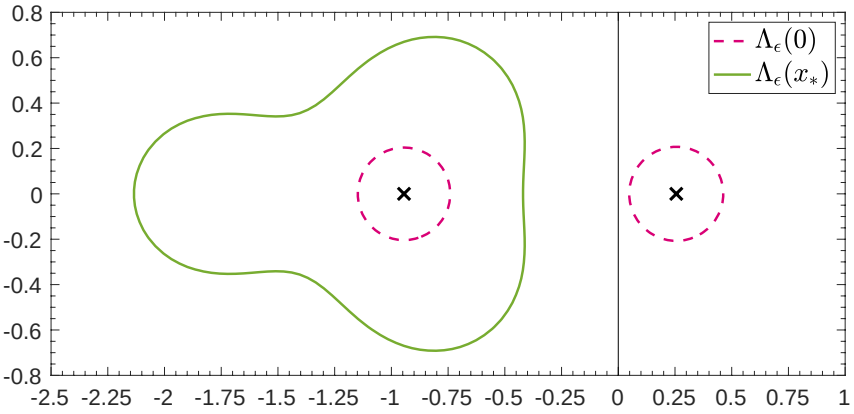
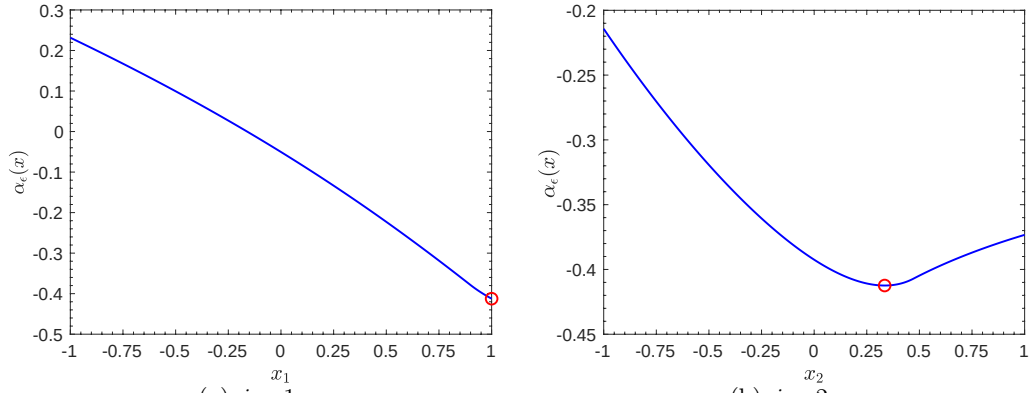


Figure 6: The plots of of the boundaries of the rightmost components of $\Lambda_\epsilon(0)$ and $\Lambda_\epsilon(x_*)$ for the HF2D2 example. The crosses mark the rightmost two eigenvalues of A, while the vertical line represents the imaginary axis.



(a) $j = 1$ (b) $j = 2$ (c) $j = 3$ (d) $j = 4$ (e) $j = 5$ (f) $j = 6$

Figure 5: The plots of $\alpha_\epsilon(x)$ for $\epsilon = 0.2$ as a function of x for the HF2D2 example near the computed global minimizer $x_* = (x_{*,1}, x_{*,2}, x_{*,3}, x_{*,4}, x_{*,5}, x_{*,6})$. Each plot corresponds to the graph of $x_j \mapsto \alpha_\epsilon(x_{*,j}[x_j])$, where $x_{*,j}[x_j]$ is equal to x_* except its jth component is x_j. The circle marks $(x_{*,j}, \alpha_\epsilon(x_*))$.

Table 6: Optimal values of $\alpha_\epsilon(x)$ over $x \in [-1,1]^6$ for the HF2D2 example by the subspace framework, and the runtimes for the subspace framework. The last three columns are runtimes in seconds as in Table 4.

$\alpha_\epsilon(x_*)$	$\alpha_\epsilon(0)$	time	red	psa
-0.4124020	0.4625511	46.1	29.1	14.5

6 Software

A Matlab implementation of the proposed subspace framework, that is Algorithm 1, is publicly available at `https://zenodo.org/record/6992092`. This implementation makes use of all of the implementation details described in Section 5.1. The numerical results on the benchmark examples in Section 5.3 can be reproduced by running the script `demo_on_benchmarks`.

7 Concluding Remarks

Minimization of the ϵ-pseudospectral abscissa of a matrix dependent on parameters for a prescribed $\epsilon > 0$ is motivated by robust stability and transient behavior considerations for the associated linear control system, as well as stabilization problems such as the stabilization by static output feedback. Here, we have proposed a subspace framework to minimize the ϵ-pseudospectral abscissa of a large matrix-valued function dependent on parameters analytically aiming at large-scale nature of the matrix-valued function. The large-scale matrix-valued functions are restricted to small subspaces, and the ϵ-pseudospectral abscissa of the resulting reduced small-scale matrix-valued functions is minimized. The subspaces are gradually expanded so as to attain Hermite interpolation properties between the ϵ-pseudospectral abscissa of the original and reduced matrix-valued functions at the minimizers of the reduced problems. We have proven the global convergence of the subspace framework in the infinite dimensional setting, that is the convergence of the optimal values of the reduced problems to the globally smallest value of the ϵ-pseudospectral of the original matrix-valued function, under mild assumptions. Additionally, we have shown that the rate of convergence of the minimizers of the reduced problems to the global minimizer of the original problem is superlinear, again under mild assumptions. The validity of these theoretical findings in practice is confirmed on synthetic and benchmark examples. The proposed framework makes it feasible to minimize the pseudospectral abscissa of matrix-valued functions of size on the order of thousands, such as the NN18 and HF2D2 examples from the *COMPl$_e$ib* collection, in a short time.

Some of the ingredients in an actual implementation of the overall subspace frame-

work are locally convergent. Specifically, the subspace framework requires the rightmost point of the ϵ-pseudospectrum of the original matrix-valued function at several parameter values, and if the matrix-valued function is really large, then it appears for this purpose one has to rely on a subspace framework such as the one proposed in [17] that converges locally. Additionally, if the minimization is over several parameters, then it appears that the reduced minimization problems must be fulfilled by employing a locally convergent optimization algorithm such as those based on Newton's method, e.g., "GRANSO" [12]. There is no quick remedy for these local convergence issues. However, we note that, on several benchmark examples, the subspace framework indeed converges globally.

The reduced minimization problems require computing the ϵ-pseudospectral abscissa of rectangular pencils as in (2.1). We solve at the moment such pseudospectral abscissa problems using an extension of the criss-cross algorithm [11] to rectangular pencils [17, Section 5.2]. One major challenge in the rectangular setting is that one has to start from a point in the rightmost component of the associated rectangular ϵ-pseudospectrum. We have outlined some ideas to overcome this challenge in Section 5.1.3. Yet, locating the rightmost point in the ϵ-pseudospectrum, and computing the ϵ-pseudospectral abscissa for a rectangular pencil appear to be far from settled. This is certainly a problem worth investigating in detail.

References

[1] N. Aliyev, P. Benner, E. Mengi, and M. Voigt. A subspace framework for $\mathcal{H}_\infty$-norm minimization. *SIAM J. Matrix Anal. Appl.*, 41(2):928–956, 2020.

[2] P. Apkarian and D. Noll. Controller design via nonsmooth multidirectional search. *SIAM J. Control Optim.*, 44:1923–1949, 2006.

[3] P. Apkarian and D. Noll. Nonsmooth H_∞ synthesis. *IEEE Trans. Autom. Control*, 51:71–86, 2006.

[4] V. Blondel, M. Gevares, and A. Lindquist. Survey on the state of systems and control. *European J. Control*, 1:5–23, 1995.

[5] V. Blondel and J. N. Tsitsiklis. NP-hardness of some linear control design problems. *SIAM J. Control Optim.*, 35(6):2118–2127, 1997.

[6] J. V. Burke, A. S. Lewis, and M. L. Overton. Optimal stability and eigenvalue multiplicity. *Found. Comput. Math.*, 1:205–225, 2001.

[7] J. V. Burke, A. S. Lewis, and M. L. Overton. Optimizing matrix stability. *Proc. Am. Math. Soc.*, 129(6):1635–1642, 2001.

[8] J. V. Burke, A. S. Lewis, and M. L. Overton. Two numerical methods for optimizing matrix stability. *Linear Algebra Appl.*, 351-352(3):117–145, 2002.

[9] J. V. Burke, A. S. Lewis, and M. L. Overton. A nonsmooth, nonconvex optimization approach to robust stabilization by static output feedback and low-order controllers. In *Proc. 4th IFAC Syposium on Robust Control Design*, Milan, Italy, June 2003.

[10] J. V. Burke, A. S. Lewis, and M. L. Overton. Robust stability and a criss-cross algorithm for pseudospectra. *IMA J. Numer. Anal.*, 23(3):359–375, 2003.

[11] J.V. Burke, A. S. Lewis, and M. L. Overton. Optimization and pseudospectra, with applications to robust stability. *SIAM J. Matrix Anal. Appl.*, 25(1):80–104, 2003.

[12] F. E. Curtis, T. Mitchell, and M. L. Overton. A BFGS-SQP method for nonsmooth, nonconvex, constrained optimization and its evaluation using relative minimization profiles. *Optim. Method Softw.*, 32(1):148–181, 2017.

[13] N. Guglielmi and M. L. Overton. Fast algorithms for the approximation of the pseudospectral abscissa and pseudospectral radius of a matrix. *SIAM J. Matrix Anal. Appl.*, 32(4):1166–1192, 2011.

[14] M. Gurbuzbalaban and M. L. Overton. Some regularity results for the pseudospectral abscissa and pseudospectral radius of a matrix. *SIAM J. Optim.*, 22(2):281–285, 2012.

[15] F. Kangal, K. Meerbergen, E. Mengi, and W. Michiels. A subspace method for large scale eigenvalue optimization. *SIAM J. Matrix Anal. Appl.*, 39(1):48–82, 2017.

[16] H. O. Kreiss. Über die stabilitätsdefinition für differenzengleichungen die partielle differenzialgleichun- gen approximieren. *BIT*, 2:153–181, 2012.

[17] D. Kressner and B. Vandereycken. Subspace methods for computing the pseudospectral abscissa and stability radius. *SIAM J. Matrix Anal. Appl.*, 35(1):292–313, 2014.

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

[19] F. Leibfritz. *COMP*leib: *CO*nstraint *M*atrix-optimization *P*roblem *lib*rary – a collection of test examples for nonlinear semidefinite programs, control system

design and related problems. Technical report, Department of Mathematics, University of Trier, 2004.

[20] A. S. Lewis and J. P. Pang. Variational analysis of pseudospectra. *SIAM J. Optim.*, 19(3):1048–1072, 2008.

[21] E. Mengi. Large-scale and global maximization of the distance to instability. *SIAM J. Matrix Anal. Appl.*, 39(4):1776–1809, 2018.

[22] E. Mengi, E. A. Yildirim, and M. Kiliç. Numerical optimization of eigenvalues of Hermitian matrix functions. *SIAM J. Matrix Anal. Appl.*, 35(2):699–724, 2014.

[23] A. Nemirovskii. Several NP-hard problems arising in robust stability analysis. *Math. Control Signals Syst.*, 6:99–105, 1993.

[24] L. N. Trefethen. Computation of pseudospectra. *Acta Numer.*, 8:247–295, 1999.

[25] L. N. Trefethen and M. Embree. *Spectra and Pseudospectra: The Behavior of Nonnormal Matrices and Operators*. Princetion University Press, Princeton, NJ, 2005.

[26] T. G. Wright. Eigtool: a graphical tool for nonsymmetric eigenproblems, 2002. Oxford University Computing Laboratory. `http://www.comlab.ox.ac.uk/pseudospectra/eigtool/`.