

A SUBSPACE FRAMEWORK FOR \mathcal{L}_∞ MODEL REDUCTION

EMRE MENGI*

Abstract. We consider the problem of locating a nearest descriptor system of prescribed reduced order to a descriptor system with large order with respect to the \mathcal{L}_∞ norm. Widely employed approaches such as the balanced truncation and best Hankel norm approximation for this \mathcal{L}_∞ model reduction problem are usually expensive and yield solutions that are not optimal, not even locally. We propose approaches based on the minimization of the \mathcal{L}_∞ objective by means of smooth optimization techniques. As we illustrate, direct applications of smooth optimization techniques are not feasible, since the optimization techniques converge at best at a linear rate requiring too many evaluations of the costly \mathcal{L}_∞ -norm objective to be practical. We replace the original large-scale system with a system of smaller order that interpolates the original system at points on the imaginary axis, and minimize the \mathcal{L}_∞ objective after this replacement. The smaller system is refined by interpolating at additional imaginary points determined based on the local minimizer of the \mathcal{L}_∞ objective, and the optimization is repeated. We argue the framework converges at a quadratic rate under smoothness and nondegeneracy assumptions, and describe how asymptotic stability constraints on the reduced system sought can be incorporated into our approach. The numerical experiments on benchmark examples illustrate that the approach leads to locally optimal solutions to the \mathcal{L}_∞ model reduction problem, and the convergence occurs quickly for descriptors systems of order a few ten thousands.

Key words. \mathcal{H}_∞ model reduction, descriptor system, quasi-Newton methods, Petrov-Galerkin projection, Hermite interpolation

AMS subject classifications. 65D05, 65F15, 65L80, 90C53, 93A15, 93C05

1. Introduction. Various applications give rise to descriptor systems with large order. A model order reduction technique typically aims at approximating the large order system with a system of much smaller and prescribed order. There are several powerful numerical approaches for the model order reduction of descriptor systems at the moment. However, to our knowledge, there does not exist a work that addresses the determination of optimal reduced order systems with respect to the \mathcal{L}_∞ norm. Even finding a locally optimal solution for the \mathcal{L}_∞ -norm model reduction problem is not addressed thoroughly. The \mathcal{H}_∞ -norm model reduction problem is closely related with the system at hand asymptotically stable, and the reduced order system sought required to be asymptotically stable.

A descriptor system is often available in the state-space representation of the form

$$(1.1) \quad Ex'(t) = Ax(t) + Bu(t), \quad y(t) = Cx(t) + Du(t),$$

for given matrices $E, A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{p \times n}$, $D \in \mathbb{R}^{p \times m}$. The \mathcal{L}_∞ norm of the transfer function

$$(1.2) \quad H(s) = C(sE - A)^{-1}B + D$$

of the system in (1.1) is defined as

$$\|H\|_{\mathcal{L}_\infty} := \sup_{\omega \in \mathbb{R}} \sigma_{\max}(H(i\omega)) = \sup_{\omega \in \mathbb{R}, \omega \geq 0} \sigma_{\max}(H(i\omega)),$$

where $\sigma_{\max}(\cdot)$ denotes the largest singular value of its matrix argument, and the last equality holds as A, B, C, D, E are real matrices. Note that we customarily set $\|H\|_{\mathcal{L}_\infty} = \sup_{\omega \in \mathbb{R}} \sigma_{\max}(H(i\omega)) = \infty$ if H has a pole on the imaginary axis, or

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

the norm of its restriction to the imaginary axis is not bounded. If the descriptor system is asymptotically stable, then its \mathcal{L}_∞ norm reduces to the \mathcal{H}_∞ -norm. Formally, let us denote with L_2^k the space of functions $f : \mathbb{R} \rightarrow \mathbb{R}^k$ satisfying $\|f\|_{L_2^k} := \sqrt{\int_{-\infty}^{\infty} \|f(t)\|_2^2 dt} < \infty$ with $k = m$ or $k = p$. For simplicity, we omit the dependence of the space on the dimension k , and write L_2 as well as $\|f\|_{L_2}$, as which space (L_2^m or L_2^p) is referred to will be clear from the context. Moreover, suppose the system in (1.1) is asymptotically stable with poles in the open left half of the complex plane. Then the \mathcal{L}_∞ norm of H is the same as the \mathcal{H}_∞ norm of H defined as

$$\|H\|_{\mathcal{H}_\infty} := \sup_{s \in \mathbb{C}^+} \sigma_{\max}(H(s)),$$

which in turn is equal to induced norm of the operator $\phi : L_2 \rightarrow L_2$ associated with (1.1) in the time domain that maps u to y defined as

$$\|\phi\|_{L_2} := \max \{ \|\phi u\|_{L_2} \mid u \in L_2 \text{ s.t. } \|u\|_{L_2} = 1 \}.$$

Hence, under the asymptotic stability assumption on the descriptor system in (1.1), we have $\|H\|_{\mathcal{L}_\infty} = \|H\|_{\mathcal{H}_\infty} = \|\phi\|_{L_2}$.

The \mathcal{L}_∞ -norm model order reduction problem – or the \mathcal{L}_∞ model reduction problem in short – for a given descriptor system of order n and for a prescribed positive integer $r < n$ concerns finding a reduced descriptor system of order r that is closest to the given system of order n with respect to the \mathcal{L}_∞ norm. Formally, let $S^{\text{red}} = (A^{\text{red}}, E^{\text{red}}, B^{\text{red}}, C^{\text{red}}, D^{\text{red}})$ denote a system of order r with the state-space representation

$$(1.3) \quad E^{\text{red}} x'(t) = A^{\text{red}} x(t) + B^{\text{red}} u(t), \quad y(t) = C^{\text{red}} x(t) + D^{\text{red}} u(t)$$

described by the matrices $E^{\text{red}}, A^{\text{red}} \in \mathbb{R}^{r \times r}$, $B^{\text{red}} \in \mathbb{R}^{r \times m}$, $C^{\text{red}} \in \mathbb{R}^{p \times r}$, $D^{\text{red}} \in \mathbb{R}^{p \times m}$, and with the transfer function

$$(1.4) \quad H(s; S^{\text{red}}) = C^{\text{red}}(sE^{\text{red}} - A^{\text{red}})^{-1}B^{\text{red}} + D^{\text{red}}.$$

Furthermore, let $S = (A, E, B, C, D)$ be the given system of order n and with the transfer function as in (1.2). The \mathcal{L}_∞ model reduction problem involves finding a descriptor system S_*^{red} of order r that minimizes the objective

$$(1.5) \quad \begin{aligned} \|H - H(\cdot; S^{\text{red}})\|_{\mathcal{L}_\infty} &= \sup_{\omega \in \mathbb{R}} [\sigma(\omega; S^{\text{red}}) := \sigma_{\max}(H(i\omega) - H(i\omega; S^{\text{red}}))] \\ &= \sup_{\omega \in \mathbb{R}, \omega \geq 0} \sigma(\omega; S^{\text{red}}) \end{aligned}$$

over all descriptor systems S^{red} of order r . The problem at hand, in particular the objective in (1.5), is non-convex, and here we aim to determine a local minimizer of the objective in (1.5) numerically. The quality of the determined local minimizer also matters, however this issue is largely dependent upon with which reduced system of order r our approach is initialized.

Two important remarks are in order regarding the minimization of the objective in (1.5). First, in addition to non-convexity, an additional difficulty is the nonsmooth nature of the problem. The objective in (1.5) as a function of S^{red} is typically not differentiable when $\sigma(\omega; S^{\text{red}})$ has multiple global maximizers over $\omega \geq 0$. Secondly, under asymptotic stability assumptions on the original system and the reduced system, the error $\|H - H(\cdot; S^{\text{red}})\|_{\mathcal{L}_\infty}$ gives a uniform upper bound on how much the

outputs of the original and reduced systems can differ. To be precise, suppose that the system S of order n , and the reduced system S^{red} of order r are asymptotically stable. Furthermore, let us denote with ϕ and ϕ_r the operators in the time domain corresponding to the systems in (1.1) and (1.3), respectively. For every $u \in L_2$, we have

$$\|y - y_r\|_{L_2} \leq \|H - H(\cdot; S^{\text{red}})\|_{\mathcal{L}_\infty} \|u\|_{L_2}$$

where y, y_r are such that $y = \phi u$ and $y_r = \phi_r u$. This means that if a small error $\|H - H(\cdot; S^{\text{red}})\|_{\mathcal{L}_\infty}$ can be ensured by the minimization of (1.5), then the output $y_r = \phi_r u$ of the minimizing reduced system approximates the original output $y = \phi u$ well uniformly over every input u of prescribed norm.

1.1. Literature and Contributions. For an asymptotically stable descriptor system with the transfer function H , the H_∞ model reduction problem - that is, for a given small order r , finding an asymptotically stable system S^{red} of order r such that $\|H - H(\cdot; S^{\text{red}})\|_{\mathcal{H}_\infty}$ is small - has been under consideration for long time. One of the classical approaches for the \mathcal{H}_∞ model reduction problem is the balanced truncation, which determines a state transformation so that the observability and controllability grammians are the same diagonal matrix, and truncates the system matrices after applying this state transformation [12, 3, 24, 25]. The reduced system by the balanced truncation is typically not even a local minimizer of $\|H - H(\cdot; S^{\text{red}})\|_{\mathcal{H}_\infty}$ over S^{red} , though it usually is a good quality approximation of H with respect to the \mathcal{H}_∞ norm [15]. The major difficulty with the balanced truncation that limits its applicability to larger systems is that it requires the solution of two Lyapunov equations involving matrices of size equal to the order of the system. With the iterative approaches for the solution of the Lyapunov equations, such as the ADI method [17, 28, 26], the balanced truncation is applicable to systems with higher order, but still Lyapunov equations stand as a hurdle.

A classical alternative is finding a best approximation with respect to the Hankel norm (HNA) [14] rather than the \mathcal{H}_∞ norm. Approaches to compute a globally optimal solution to HNA in polynomial time are proposed [14]. However, the globally optimal solution to HNA is again usually not even a local minimizer of $\|H - H(\cdot; S^{\text{red}})\|_{\mathcal{H}_\infty}$. Furthermore, finding a globally optimal solution to HNA is even more costlier than the balanced truncation. Even with the efficient use of computational linear algebra tools [6], solving HNA for systems with high order is out of reach.

Here, we propose an approach to compute a local minimizer of $\|H - H(\cdot; S^{\text{red}})\|_{\mathcal{L}_\infty}$ over all systems S^{red} of order r . To our knowledge, the approach is the first attempt to find such a locally optimal solution. The approach uses smooth optimization techniques, which have been employed for solving nonsmooth optimization problems [19, 4] in the last fifteen years. Most often, they seem to be capable of locating locally optimal solutions, but slowly at best at a linear rate. Consequently, as we shall see below, a direct application of them to minimize $\|H - H(\cdot; S^{\text{red}})\|_{\mathcal{L}_\infty}$ is prohibitively expensive even for systems of medium order, as it requires the computation of the objective, that is the \mathcal{L}_∞ norm, too many times. Instead, we replace H with an approximation \tilde{H} of small order greater than r . Rather than $\|H - H(\cdot; S^{\text{red}})\|_{\mathcal{L}_\infty}$, we minimize $\|\tilde{H} - H(\cdot; S^{\text{red}})\|_{\mathcal{L}_\infty}$, then update \tilde{H} based on the minimizer, and repeat. The approximation \tilde{H} is built using the Petrov-Galerkin framework, and an update involves the expansion of the projection subspaces for the Petrov-Galerkin framework. We show that the proposed framework converges quadratically under simplicity assumptions. We also describe how the asymptotic stability constraints can be imposed on the variable S^{red} when minimizing $\|H - H(\cdot; S^{\text{red}})\|_{\mathcal{L}_\infty}$ in case the original system

in (1.1) is asymptotically stable. As the system corresponding to $H - H(\cdot; S^{\text{red}})$ is asymptotically stable when S and S^{red} is asymptotically stable, the incorporation of this constraint into our approach leads to a local minimization of $\|H - H(\cdot; S^{\text{red}})\|_{\mathcal{H}_\infty}$ over all asymptotically stable systems S^{red} of order r , i.e., locally optimal solution of the \mathcal{H}_∞ model reduction problem.

Iterative rational Krylov algorithm (IRKA) [16] is proposed to find a reduced order system of prescribed order that is locally optimal with respect to the \mathcal{H}_2 norm defined as $\|H\|_{\mathcal{H}_2} = \sqrt{\frac{1}{2\pi} \int_{-\infty}^{\infty} \text{trace}(H(i\omega)^* H(i\omega)) d\omega}$ for a system with the transfer function H . Formally, IRKA is an iterative interpolatory approach that finds a local minimizer of $\|H - H(\cdot; S^{\text{red}})\|_{\mathcal{H}_2}$ over all systems S^{red} of order r . In [13], starting from the reduced order system of order r generated by IRKA, an optimization based approach is proposed to find a locally optimal solution of $\|H - H(\cdot; S^{\text{red}})\|_{\mathcal{H}_\infty}$ for single-input-single-output (SISO) systems but with respect to particular rank-one modifications $\Delta_A = \varepsilon \mathbf{e} \mathbf{e}^T$, $\Delta_B = -\varepsilon \mathbf{e}$, $\Delta_C = -\varepsilon \mathbf{e}^T$, $\Delta_D = \varepsilon$ of the system matrices A^{red} , B^{red} , C^{red} , D^{red} generated by IRKA over the optimization parameter ε . In the reported results in [13], this optimization improves the accuracy of the reduced system returned by IRKA by a factor of 2-4 with respect to the \mathcal{H}_∞ norm. But again the eventual system is usually not a local minimizer of the objective $\|H - H(\cdot; S^{\text{red}})\|_{\mathcal{H}_\infty}$ over systems S^{red} of order r .

On a related note, our recent work [2] concerns the minimization of the \mathcal{H}_∞ norm of a descriptor system with large order dependent on parameters. At the center of that work is a subspace framework to cope with the large order of the system. It may seem plausible to look at the current work from that perspective. However, we have too many optimization parameters here. As a result applying the framework over there to attain quick convergence in the setting here is not feasible, as doing so yields projection subspaces growing rapidly (i.e., see Algorithm 2 in [2] to attain superlinear convergence). In the framework here, only $4m$ new directions, independent of r , are added into the subspaces at every iteration. Moreover, we observe quick convergence, so the subspaces remain small throughout.

1.2. Outline. We first consider the direct minimization of $\|H - H(\cdot; S^{\text{red}})\|_{\mathcal{L}_\infty}$ over systems S^{red} of order r by means of smooth optimization techniques in Section 2. In this section, we indicate the optimization variables, and spell out expressions for the first derivatives of the objective with respect to these variables. As we shall see, the direct optimization is too costly even for systems with moderate order, since smooth optimization techniques converge very slowly and require the evaluation of the \mathcal{L}_∞ objective too many times. Consequently, in Section 3, we replace the transfer function H with an approximating transfer function \tilde{H} of small order greater than r that Hermite interpolates H at several points on the imaginary axis. Then we minimize $\|\tilde{H} - H(\cdot; S^{\text{red}})\|_{\mathcal{L}_\infty}$ (by smooth optimization techniques), and refine \tilde{H} so that Hermite interpolation with H at another point on the imaginary axis is attained based on the computed minimizer of $\|\tilde{H} - H(\cdot; S^{\text{red}})\|_{\mathcal{L}_\infty}$. We introduce a refinement step on \tilde{H} so that interpolation properties can be attained between the full objective $\|H - H(\cdot; S^{\text{red}})\|_{\mathcal{L}_\infty}$ and the reduced objective $\|\tilde{H} - H(\cdot; S^{\text{red}})\|_{\mathcal{L}_\infty}$. Then the procedure is repeated with the refined \tilde{H} . In Section 4, we investigate the interpolation properties between full objective and the reduced objective. Based on these interpolation properties, we argue in Section 5 that the algorithm converges at a quadratic rate under smoothness and nondegeneracy assumptions. If the original descriptor system is asymptotically stable, it may be natural to minimize $\|\tilde{H} - H(\cdot; S^{\text{red}})\|_{\mathcal{L}_\infty}$

subject to the asymptotic stability constraints on the reduced system S^{red} . We discuss in Section 6 the incorporation of such asymptotic stability constraints on the reduced system into our approach. Section 7 is devoted to the details that need to be taken into account in a practical implementation of the proposed algorithm such as the initialization of the smooth optimization routines, and termination. A Matlab implementation of the algorithm is publicly available. In Section 8, we report numerical results obtained with this implementation. The numerical results indicate quick convergence to a locally optimal solution, and the capability to deal with systems with large order on the order of ten thousands.

2. Use of First and Second Order Derivative-Based Methods. First order methods such as the gradient descent algorithm, and second order methods such as quasi-Newton algorithms equipped with proper line-searches have been successfully applied to nonsmooth optimization problems in recent years. Here, if a curvature condition is employed in the line-search, this should take into consideration the fact that the directional derivatives do not have to converge to zero unlike the situation for smooth optimization problems, e.g., if Wolfe conditions are imposed in the line-search, weak Wolfe conditions should be used rather than strong Wolfe conditions. Also, for termination small gradient norms should not be required. Instead, for instance, a failure in sufficient decrease in the objective along the descent search direction may indicate convergence to a locally optimal solution.

The objective to be minimized in (1.5) for the \mathcal{L}_∞ -norm model reduction problem can be expressed as

$$\begin{aligned}
 \mathcal{F}(S^{\text{red}}) &= \sup_{\omega \geq 0} \sigma_{\max}(H(i\omega) - H(i\omega; S^{\text{red}})) \\
 (2.1) \quad &= \sup_{\omega \geq 0} \sigma_{\max}(\mathcal{H}(i\omega; S^{\text{red}})) = \|\mathcal{H}(\cdot; S^{\text{red}})\|_{\mathcal{L}_\infty}, \\
 \mathcal{H}(s; S^{\text{red}}) &:= [C \quad -C^{\text{red}}] \begin{bmatrix} sE - A & 0 \\ 0 & sE^{\text{red}} - A^{\text{red}} \end{bmatrix}^{-1} \begin{bmatrix} B \\ B^{\text{red}} \end{bmatrix} + (D - D^{\text{red}}),
 \end{aligned}$$

where $H(\cdot; S^{\text{red}})$ is as in (1.4). Assuming that the reduced system is at most index one and has semi-simple poles, by the Kronecker canonical form, there exist invertible $r \times r$ real matrices W, V such that $WE^{\text{red}}V$ is diagonal, and $WA^{\text{red}}V$ is block diagonal with 2×2 and 1×1 blocks along the diagonal. Consequently, the reduced system is equivalent to a system (with the same transfer function) for which $A^{\text{red}}, E^{\text{red}}$ are converted into tridiagonal and diagonal forms, respectively. Hence, under index one and semi-simple pole assumptions, we can perform the minimization over tridiagonal A^{red} and diagonal E^{red} . Recalling the dimensions of $A^{\text{red}}, B^{\text{red}}, C^{\text{red}}, D^{\text{red}}, E^{\text{red}}$, there are precisely $4r - 2 + rm + pr + pm$ optimization variables.

The gradient descent algorithm, as well as quasi-Newton algorithms to minimize \mathcal{F} require the gradients of \mathcal{F} . To this end, suppose there is a unique $\omega_* \geq 0$ satisfying

$$\sigma_{\max}(H(i\omega_*) - H(i\omega_*; S^{\text{red}})) = \mathcal{F}(S^{\text{red}}) = \sup_{\omega \geq 0} \sigma_{\max}(H(i\omega) - H(i\omega; S^{\text{red}})),$$

ensuring that \mathcal{F} is differentiable at S^{red} . Additionally, let u, v denote a consistent pair of unit left, right singular vectors corresponding to $\sigma_{\max}(H(i\omega_*) - H(i\omega_*; S^{\text{red}}))$, and let us introduce $\tilde{u} := u^* C^{\text{red}} (i\omega_* E^{\text{red}} - A^{\text{red}})^{-1}$, $\tilde{v} := (i\omega_* E^{\text{red}} - A^{\text{red}})^{-1} B^{\text{red}} v$. Then, by employing the analytical formulas for the derivatives of singular value functions

[18], [23, Section 3.3], the gradients of \mathcal{F} are given by

$$\begin{aligned}
 \nabla_{A^{\text{red}}}\mathcal{F}(S^{\text{red}}) &= -\text{diag}(\Re(\tilde{u}^T \odot \tilde{v})) - \text{diag}_{-1}(\Re(\tilde{u}(2:r)^T \odot \tilde{v}(1:r-1))) \\
 &\quad - \text{diag}_{+1}(\Re(\tilde{u}(1:r-1)^T \odot \tilde{v}(2:r))), \\
 (2.2) \quad \nabla_{E^{\text{red}}}\mathcal{F}(S^{\text{red}}) &= -\omega_* \cdot \text{diag}(\Im(\tilde{u}^T \odot \tilde{v})), \quad \nabla_{B^{\text{red}}}\mathcal{F}(S^{\text{red}}) = -\Re(\tilde{u}^T v^T), \\
 \nabla_{C^{\text{red}}}\mathcal{F}(S^{\text{red}}) &= -\Re(\bar{u} \tilde{v}^T), \quad \nabla_{D^{\text{red}}}\mathcal{F}(S^{\text{red}}) = -\Re(\bar{u} v^T),
 \end{aligned}$$

where \odot denotes the Hadamard product, \bar{u} denotes the complex conjugate of u , and the notation $\text{diag}(w)$ represents the square diagonal matrix whose diagonal entries are formed of the entries of the vector w . The notations $\text{diag}_{-1}(w)$ and $\text{diag}_{+1}(w)$ are similar to $\text{diag}(w)$ but with the difference that the subdiagonal and superdiagonal entries of the matrix are filled with the entries of w rather than the diagonal entries.

It is essential that a quasi-Newton method such as BFGS generates approximate Hessians that are positive definite. This is traditionally imposed by the line-searches. For instance, if BFGS is to be used to minimize \mathcal{F} , then a line-search ensuring the satisfaction of the weak Wolfe conditions may be adopted so that the approximate Hessians remain positive definite. On the other hand, for the gradient descent algorithm to minimize \mathcal{F} it is sufficient to adopt a simpler line-search that guarantees only sufficient reduction in the objective, e.g., an Armijo backtracking line-search.

One difficulty with using methods such as gradient descent and BFGS to minimize \mathcal{F} is that these algorithms converge rather slowly only at a linear rate at best. This may sound surprising especially for BFGS, which typically converges superlinearly for smooth problems. Slower convergence for BFGS is an artifact of nonsmoothness. As a result of linear convergence at best, the objective \mathcal{F} typically needs to be evaluated many times until reaching a prescribed accuracy. This may be prohibitively expensive, as it is apparent from (2.1) that evaluation of \mathcal{F} involves the computation of the \mathcal{L}_∞ norm of $\mathcal{H}(\cdot; S^{\text{red}})$, the transfer function for a large-scale system assuming the original system in (1.1) is large-scale.

To illustrate the slow convergence issues in the previous paragraph, and the computational difficulties that come with it, we apply the gradient descent algorithm to the `iss` example from the SLICOT collection. The system associated with this example has order $n = 270$, and $m = p = 3$. We attempt to solve the \mathcal{L}_∞ model reduction problem for $r = 12$ starting with the initial reduced order model generated by the balanced truncation approach. The errors (\mathcal{F}) and the 2-norms of the gradients of the errors ($\|\nabla\mathcal{F}\|_2$) of the iterates of the gradient descent algorithm are reported in Table 2.1. It takes 37 iterations until the errors in two consecutive iterations differ by no more than 10^{-6} in a relative sense. The initial \mathcal{L}_∞ -norm error 0.004470060020 (of the system obtained from the balanced truncation) is reduced to 0.002415438945 after 37 iterations. The eventual reduced model obtained appears to be a local minimizer of \mathcal{F} up to prescribed tolerances, as can be observed from the plots in Figure 2.1. Note however that according to the last columns in Table 2.1 the gradients of \mathcal{F} do not seem to be converging to zero, which indicates that the objective is not differentiable at the local minimizer to be converged. Meanwhile, the objective \mathcal{F} is evaluated 624 times, since the line-search at each iteration requires several objective function evaluations (i.e., to be precise 8-28 evaluations per iteration) until the satisfaction of the sufficient decrease condition. This results in a total runtime of about 500 seconds, costly for a system of relatively small order. To conclude, direct applications of the gradient descent and quasi-Newton algorithms do not seem viable for systems of even

TABLE 2.1

This concerns the \mathcal{L}_∞ model reduction of the *iss* example with $r = 12$. The objective $\mathcal{F}^{(k)} := \mathcal{F}(A^{(k)}, B^{(k)}, C^{(k)}, D^{(k)}, E^{(k)})$, and the 2-norm of $\nabla\mathcal{F}^{(k)} := \nabla\mathcal{F}(A^{(k)}, B^{(k)}, C^{(k)}, D^{(k)}, E^{(k)})$ for the iterate $(A^{(k)}, B^{(k)}, C^{(k)}, D^{(k)}, E^{(k)})$ by the gradient descent method at the k th iteration are listed.

k	$\mathcal{F}^{(k)}$	$\ \nabla\mathcal{F}^{(k)}\ _2$	k	$\mathcal{F}^{(k)}$	$\ \nabla\mathcal{F}^{(k)}\ _2$
0	0.004470060020	1.000093488	30	0.002415516341	0.803721909
1	0.004346739384	0.833556647	31	0.002415479783	1.000008471
2	0.003609940202	1.000097230	32	0.002415475189	0.803718441
3	0.003175718111	0.769359926	33	0.002415456030	1.000008467
4	0.002975716755	1.000095596	34	0.002415454613	0.803716708
5	0.002946113130	0.999918608	35	0.002415444154	1.000008465
6	0.002697635041	0.844275929	36	0.002415439844	0.803714645
7	0.002656707905	0.999952423	37	0.002415438945	1.000008462

modest order (e.g., a few thousands).

3. A Subspace Framework. The computational difficulty in minimizing the objective \mathcal{F} in (2.1) is due to the large order of the original system $S = (A, E, B, C, D)$. In this section, we propose to replace this system with a system of smaller order $S_r = (A_r, E_r, B_r, C_r, D_r)$ with the state-space representation

$$(3.1) \quad E_r x_r'(t) = A_r x_r(t) + B_r u(t), \quad y(t) = C_r x_r(t) + D u(t),$$

and solve the resulting \mathcal{L}_∞ model reduction problem, that is minimize

$$(3.2) \quad \mathcal{F}_r(S^{\text{red}}) = \sup_{\omega \geq 0} \sigma_{\max}(H_r(i\omega) - H(i\omega; S^{\text{red}})) = \sup_{\omega \geq 0} \sigma_{\max}(\mathcal{H}_r(i\omega; S^{\text{red}})),$$

where

$$H_r(s) := C_r(sE_r - A_r)^{-1}B_r + D, \quad \text{and}$$

$$\mathcal{H}_r(s; S^{\text{red}}) := [C_r \quad -C^{\text{red}}] \begin{bmatrix} sE_r - A_r & 0 \\ 0 & sE^{\text{red}} - A^{\text{red}} \end{bmatrix}^{-1} \begin{bmatrix} B_r \\ B^{\text{red}} \end{bmatrix} + (D - D^{\text{red}}).$$

The question that we need to address is how do we form a small system $S_r = (A_r, E_r, B_r, C_r, D)$ that is a good representative of the original system near a local minimizer of the original \mathcal{L}_∞ model reduction problem.

Recall how pure Newton's method operates to minimize a function $f: \mathbb{R}^q \rightarrow \mathbb{R}$. It approximates f with a quadratic model, and finds a local minimizer \tilde{x} of the quadratic model. Then, assuming f is twice differentiable at \tilde{x} , it refines the quadratic model so that the refined quadratic model q satisfies $f(\tilde{x}) = q(\tilde{x})$, $\nabla f(\tilde{x}) = \nabla q(\tilde{x})$ and $\nabla^2 f(\tilde{x}) = \nabla^2 q(\tilde{x})$. In the context of \mathcal{L}_∞ model reduction, we view \mathcal{F}_r as the model function for \mathcal{F} , even though \mathcal{F}_r is not quadratic. We minimize \mathcal{F}_r locally rather than \mathcal{F} , and refine the small system in (3.1) with the hope that the objective error function \mathcal{F}_{r+1} of the refined system interpolates \mathcal{F} and its first two derivatives at the computed minimizer of \mathcal{F}_r .

The small system in (3.1) is obtained from the original system by applying the Petrov-Galerkin framework; for given two subspaces $\mathcal{V}_r, \mathcal{W}_r$ of \mathbb{R}^n of equal dimension, the state space of the original system is restricted to \mathcal{V}_r and the differential part of the resulting system is imposed to be orthogonal to \mathcal{W}_r . Formally, denoting with V_r, W_r matrices whose columns form orthonormal bases for $\mathcal{V}_r, \mathcal{W}_r$ and with $x_r(t)$ the restricted state, the original system is approximated by

$$W_r^T(EV_r x_r'(t) - AV_r x_r(t) - Bu(t)) = 0, \quad y(t) = CV_r x_r(t) + Du(t),$$

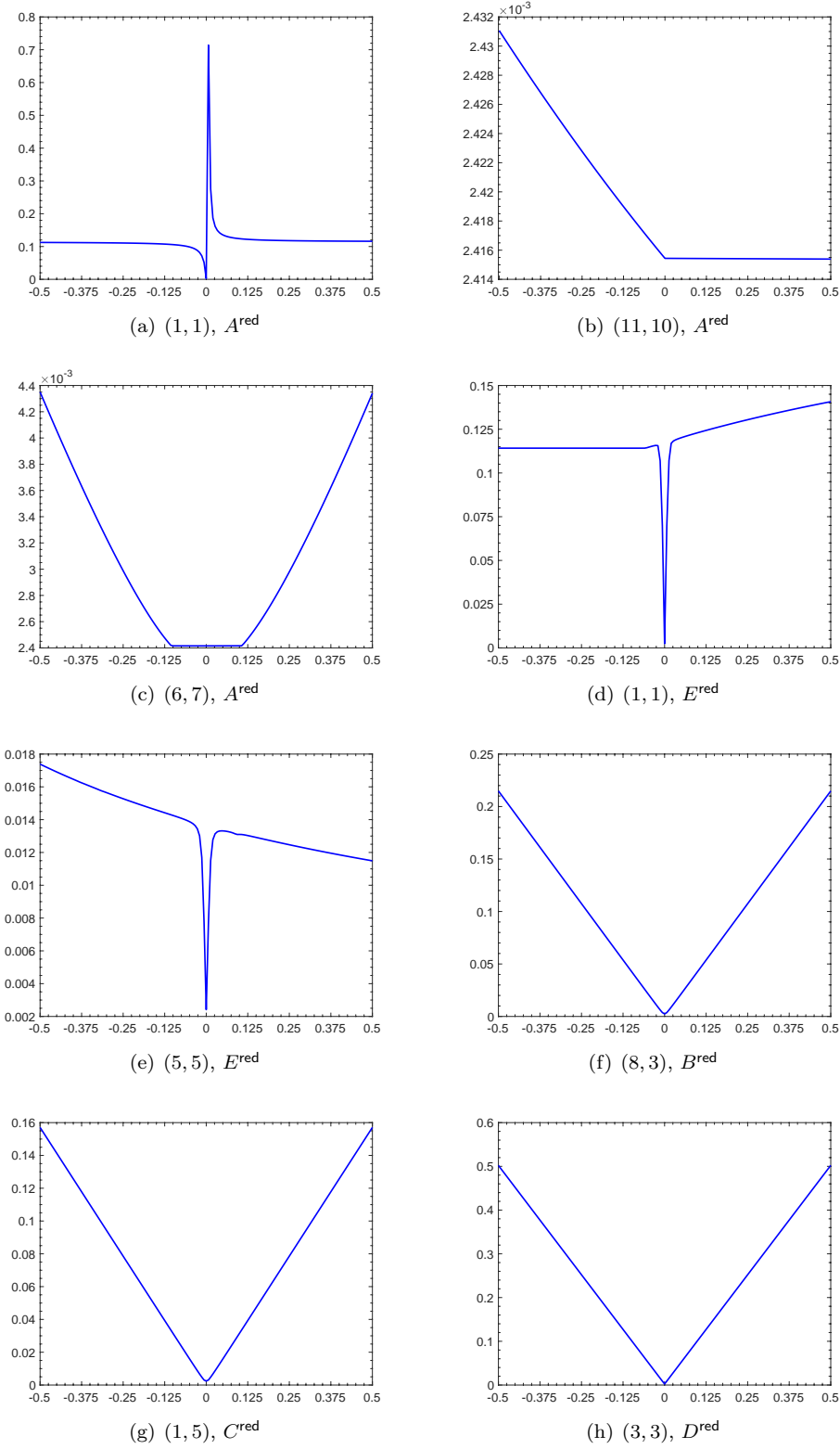


FIG. 2.1. The locally minimal reduced system generated by the gradient descent method for the *iss* example and $r = 12$ is varied, and the error \mathcal{F} is plotted as a function of the variation. In each one of the plots (a)–(h), only the indicated entry of one of the optimal coefficients $A^{\text{red}}, B^{\text{red}}, C^{\text{red}}, D^{\text{red}}, E^{\text{red}}$ is varied by amounts in $[-0.5, 0.5]$. Zero variation corresponds to the optimal reduced system.

giving rise to a system of the form (3.1) with

$$(3.3) \quad E_r = W_r^T E V_r, \quad A_r = W_r^T A V_r, \quad B_r = W_r^T B, \quad \text{and} \quad C_r = C V_r.$$

For the realization of the ideas in the previous paragraph, we need to be equipped with a tool that gives us the capability to interpolate $H(s)$ and its derivatives at a prescribed point in the complex plane with those of the transfer function for the small system. This tool is introduced in the next result, which follows from [5, Theorem 1].

THEOREM 3.1. *Let $\mu \in \mathbb{C}$ be such that $A - \mu E$ is invertible. Suppose*

$$\begin{aligned} \bigoplus_{j=0}^{\kappa} \Re \left[\{(A - \mu E)^{-1} E\}^j (A - \mu E)^{-1} B \right] &\subseteq \mathcal{V}_r, \\ \bigoplus_{j=0}^{\kappa} \Im \left[\{(A - \mu E)^{-1} E\}^j (A - \mu E)^{-1} B \right] &\subseteq \mathcal{V}_r, \\ \bigoplus_{j=0}^{\kappa} \Re \left[C(A - \mu E)^{-1} \{E(A - \mu E)^{-1}\}^j \right]^* &\subseteq \mathcal{W}_r, \quad \text{and} \\ \bigoplus_{j=0}^{\kappa} \Im \left[C(A - \mu E)^{-1} \{E(A - \mu E)^{-1}\}^j \right]^* &\subseteq \mathcal{W}_r. \end{aligned}$$

Then, with A_r, E_r, B_r, C_r defined as in (3.3), if $A_r - \mu E_r$ is invertible, we have

- (i) $H(\mu) = H_r(\mu)$ and $H(\bar{\mu}) = H_r(\bar{\mu})$,
- (ii) $H^{(j)}(\mu) = H_r^{(j)}(\mu)$ and $H^{(j)}(\bar{\mu}) = H_r^{(j)}(\bar{\mu})$ for $j = 1, \dots, 2\kappa + 1$.

Our proposed subspace framework at iteration r first finds a minimizer of $\mathcal{F}_r(S^{\text{red}})$, say $S_r^{\text{red}} = (A_r^{\text{red}}, B_r^{\text{red}}, C_r^{\text{red}}, D_r^{\text{red}}, E_r^{\text{red}})$. This is followed by the computation of an $\omega_r \in \mathbb{R}$, $\omega_r \geq 0$ such that

$$\mathcal{F}(S_r^{\text{red}}) = \sup_{\omega \geq 0} \sigma_{\max} (H(i\omega) - H(i\omega; S_r^{\text{red}})) = \sigma_{\max} (H(i\omega_r) - H(i\omega_r; S_r^{\text{red}})).$$

Computing such an ω_r requires the large-scale \mathcal{L}_{∞} -norm computation in (2.1) but by replacing $S^{\text{red}} = (A^{\text{red}}, B^{\text{red}}, C^{\text{red}}, D^{\text{red}}, E^{\text{red}})$ with $S_r^{\text{red}} = (A_r^{\text{red}}, B_r^{\text{red}}, C_r^{\text{red}}, D_r^{\text{red}}, E_r^{\text{red}})$. Then subspaces are expanded so that H and its first three derivatives are interpolated at $i\omega_r$ by those of the transfer function for the small system. A formal description of the framework is given in Algorithm 3.1 below. As the subspaces \mathcal{V}_r and \mathcal{W}_r are required to be of equal dimension, the description assumes that the number of inputs and the outputs are equal, i.e., $m = p$. Even if it is omitted here for simplicity, it is straightforward to modify the directions $\tilde{V}_{r+1}, \tilde{W}_{r+1}$ in lines 11-12 to be added to the subspaces $\mathcal{V}_r, \mathcal{W}_r$ in order to deal with the systems for which $m \neq p$. The final refinement step in line 15 aims at the satisfaction of the interpolation condition $\mathcal{F}(S_r^{\text{red}}) = \mathcal{F}_{r+1}(S_r^{\text{red}})$, as well as the interpolation conditions on the derivatives of $\mathcal{F}(S^{\text{red}})$ and $\mathcal{F}_{r+1}(S^{\text{red}})$ at S_r^{red} . This step is elaborated on in the next subsection.

3.1. Refinement Step. First we make a few observations regarding the relation between $\mathcal{F}(S_r^{\text{red}})$ and $\mathcal{F}_{r+1}(S_r^{\text{red}})$ at the r th subspace iteration in Algorithm 3.1 right before the refinement step.

At the r th iteration of Algorithm 3.1 right after line 14, by Theorem 3.1, we have

$$(3.4) \quad H(i\omega_r) = H_{r+1}(i\omega_r) \quad \text{and} \quad H^{(j)}(i\omega_r) = H_{r+1}^{(j)}(i\omega_r)$$

Algorithm 3.1 Subspace framework for \mathcal{L}_∞ model reduction

Input: System $S = (A, E, B, C, D)$ as in (1.1), the order $r \in \mathbb{Z}^+$ of the reduced system sought, and an initial estimate $S_0^{\text{red}} = (A_0^{\text{red}}, E_0^{\text{red}}, B_0^{\text{red}}, C_0^{\text{red}}, D_0^{\text{red}})$ of order r for a minimizer of \mathcal{F} as in (2.1).

Output: Estimate $S_\star^{\text{red}} = (A_\star^{\text{red}}, E_\star^{\text{red}}, B_\star^{\text{red}}, C_\star^{\text{red}}, D_\star^{\text{red}})$ for a minimizer of \mathcal{F} as in (2.1).

- 1: Choose the initial subspaces $\mathcal{V}_0, \mathcal{W}_0$ and orthonormal bases V_0, W_0 for them.
- 2: Form A_0, B_0, C_0, E_0 using (3.3), and let $S_0 = (A_0, E_0, B_0, C_0, D)$.

% main loop

- 3: **for** $r = 0, 1, \dots$ **do**
- 4: **if** $r \geq 1$ **then**
- 5: $S_r^{\text{red}} \leftarrow$ a minimizer of $\mathcal{F}_r(S^{\text{red}})$.
- 6: **end if**
- 7: $\omega_r \leftarrow$ a maximizer of $\sigma(\omega; S_r^{\text{red}}) = \sigma_{\max}(H(i\omega) - H(i\omega; S_r^{\text{red}}))$ over $\omega \geq 0$.
- 8: **if** $r \geq 1$ **then**
- 9: **Return** if convergence has occurred with $S_\star^{\text{red}} \leftarrow S_r^{\text{red}}$.
- 10: **end if**
- 11: **% expand the subspaces to interpolate at $i\omega_r$**
- 12: $\tilde{V}_{r+1} \leftarrow [\Re[(i\omega_r E - A)^{-1} B] \quad \Re[(i\omega_r E - A)^{-1} E(i\omega_r E - A)^{-1} B] \quad \Im[(i\omega_r E - A)^{-1} B] \quad \Im[(i\omega_r E - A)^{-1} E(i\omega_r E - A)^{-1} B]]$.
- 13: $\tilde{W}_{r+1} \leftarrow [\Re[(i\omega_r E - A)^{-*} C^*] \quad \Re[(i\omega_r E - A)^{-*} E(i\omega_r E - A)^{-*} C^*] \quad \Im[(i\omega_r E - A)^{-*} C^*] \quad \Im[(i\omega_r E - A)^{-*} E(i\omega_r E - A)^{-*} C^*]$.
- 14: $V_{r+1} \leftarrow \text{orth} \left(\begin{bmatrix} V_r & \tilde{V}_{r+1} \end{bmatrix} \right)$ and $W_{r+1} \leftarrow \text{orth} \left(\begin{bmatrix} W_r & \tilde{W}_{r+1} \end{bmatrix} \right)$.
- 15: **% update the small system**
- 16: Form $A_{r+1}, B_{r+1}, C_{r+1}, E_{r+1}$ using (3.3),
and let $S_{r+1} = (A_{r+1}, E_{r+1}, B_{r+1}, C_{r+1}, D)$.
- 17: **% refine the small system**
- 18: Refine V_{r+1}, W_{r+1} and S_{r+1} if necessary (using Algorithm 3.2).
- 19: **end for**

for $j = 1, 2, 3$ under the assumptions that $A - i\omega_r E$ and $A_{r+1} - i\omega_r E_{r+1}$ are invertible. Consequently, $H(i\omega_r) - H(i\omega_r; S_r^{\text{red}})$ and $H_{r+1}(i\omega_r) - H(i\omega_r; S_r^{\text{red}})$ are equal, and share the same set of left and right singular vectors. It immediately follows that setting

$$(3.5) \quad \sigma_{r+1}(\omega; S^{\text{red}}) := \sigma_{\max}(H_{r+1}(i\omega) - H(i\omega; S^{\text{red}})),$$

and recalling the definition of $\sigma(\omega; S^{\text{red}})$ in (1.5), we have

$$(3.6) \quad \sigma(\omega_r; S_r^{\text{red}}) = \sigma_{r+1}(\omega_r; S_r^{\text{red}}).$$

Indeed, as the singular values and vectors of $H(i\omega_r) - H(i\omega_r; S_r^{\text{red}})$ and $H_{r+1}(i\omega_r) - H(i\omega_r; S_r^{\text{red}})$ are the same, and the first two derivatives of $H(i\omega) - H(i\omega; S_r^{\text{red}})$ and

$H_{r+1}(i\omega) - H(i\omega; S_r^{\text{red}})$ at $\omega = \omega_r$ are equal due to (3.4), we also have

$$(3.7) \quad \frac{d^j \sigma}{d\omega^j}(\omega_r; S_r^{\text{red}}) = \frac{d^j \sigma_{r+1}}{d\omega^j}(\omega_r; S_r^{\text{red}})$$

for $j = 1, 2$. Now ω_r is a global maximizer of $\sigma(\omega; S_r^{\text{red}})$ over ω implying

$$\frac{d\sigma}{d\omega}(\omega_r; S_r^{\text{red}}) = 0 \quad \text{and} \quad \frac{d^2\sigma}{d\omega^2}(\omega_r; S_r^{\text{red}}) \leq 0.$$

Assuming that the last inequality on the second derivative above holds strictly, (3.7) implies ω_r is also a local maximizer of $\sigma_{r+1}(\omega; S_r^{\text{red}})$.

Regarding $\mathcal{F}(S_r^{\text{red}})$ and $\mathcal{F}_{r+1}(S_r^{\text{red}})$, the following relation always hold:

$$(3.8) \quad \begin{aligned} \mathcal{F}(S_r^{\text{red}}) &= \sup_{\omega \geq 0} \sigma(\omega; S_r^{\text{red}}) \\ &= \sigma(\omega_r; S_r^{\text{red}}) \\ &= \sigma_{r+1}(\omega_r; S_r^{\text{red}}) \\ &\leq \sup_{\omega \geq 0} \sigma_{r+1}(\omega; S_r^{\text{red}}) = \mathcal{F}_{r+1}(S_r^{\text{red}}), \end{aligned}$$

where the third equality is due to the interpolatory property in (3.6). As argued in the previous paragraph, the point ω_r is not only a global maximizer of $\sigma(\omega; S_r^{\text{red}})$, but also generically a local maximizer of $\sigma_{r+1}(\omega; S_r^{\text{red}})$. If it happens that ω_r is also a global maximizer of $\sigma_{r+1}(\omega; S_r^{\text{red}})$ beyond being a local maximizer, then the inequality in the equation above becomes an equality, and we have the interpolation property

$$(3.9) \quad \mathcal{F}(S_r^{\text{red}}) = \mathcal{F}_{r+1}(S_r^{\text{red}}).$$

In the refinement step, if it happens that ω_r is merely a local maximizer of $\sigma_{r+1}(\omega; S_r^{\text{red}})$, but not a global maximizer, then we find a global maximizer $\omega_r^{(0)}$ of $\sigma_{r+1}(\omega; S_r^{\text{red}})$ over $\omega \geq 0$ (equivalently compute the \mathcal{L}_∞ norm of $H_{r+1}(\cdot) - H(\cdot; S_r^{\text{red}})$). Observe that finding such a global maximizer has a small computational cost, as the orders of S_{r+1} and S_r^{red} are small. Then, by employing Theorem 3.1, we expand the subspaces \mathcal{V}_{r+1} , \mathcal{W}_{r+1} further so that the interpolatory properties are attained between $H_{r+1}(i\omega)$ after this refinement and $H(i\omega)$ at $\omega = \omega_r^{(0)}$, which in turn implies interpolatory properties between $\sigma_{r+1}(\omega; S_r^{\text{red}})$ and $\sigma(\omega; S_r^{\text{red}})$ at $\omega = \omega_r^{(0)}$. If ω_r after this refinement of S_{r+1} is still only a local maximizer of $\sigma_{r+1}(\omega; S_r^{\text{red}})$, but not a global maximizer, then we repeat this refinement procedure of S_{r+1} up until ω_r becomes a global maximizer of $\sigma_{r+1}(\omega; S_r^{\text{red}})$ (in practice up to prescribed tolerances). A formal description of the refinement step is given below in Algorithm 3.2. For simplicity, in line 3 of Algorithm 3.2 it is assumed that ω_r is the unique global maximizer of $\sigma(\omega; S_r^{\text{red}})$. More generally, all of the global maximizers of $\sigma(\omega; S_r^{\text{red}})$ can be returned in line 7 of Algorithm 3.1 (e.g., by employing the level-set methods to compute the \mathcal{L}_∞ norm), and whether $\omega_r^{(j)}$ is equal to any of these global maximizers can be checked in line 3 of Algorithm 3.2.

Assuming $\sigma(\omega; S_r^{\text{red}})$ is Lipschitz continuous, $\sigma_{r+1}(\omega; S_r^{\text{red}})$ is Lipschitz continuous with a uniform Lipschitz constant over the iterations of Algorithm 3.2, and the maximizer $\omega_r^{(j)}$ of $\sigma_{r+1}(\omega; S_r^{\text{red}})$ over $\omega \geq 0$ at every j is required to be in a prescribed bounded interval, the gap $|\omega_r^{(j)} - \omega_r|$ can be made less than any prescribed amount after finitely many iterations of Algorithm 3.2. At this point, the interpolation condition (3.9) is also met up to a multiple of the prescribed amount.

Algorithm 3.2 Refinement Step

```

1: for  $j = 0, 1, \dots$  do
2:    $\omega_r^{(j)} \leftarrow$  a maximizer of  $\sigma_{r+1}(\omega; S_r^{\text{red}}) = \sigma_{\max}(H_{r+1}(i\omega) - H(i\omega; S_r^{\text{red}}))$ 
      over  $\omega \geq 0$ .
3:   if  $\omega_r^{(j)} = \omega_r$  (up to prescribed tolerances) then
4:     Terminate with  $V_{r+1}$ ,  $W_{r+1}$  and  $S_{r+1}$ .
5:   end if
6:   % expand the subspaces to interpolate at  $i\omega_r^{(j)}$ 
       $\tilde{V}_{r+1} \leftarrow \begin{bmatrix} \Re[(i\omega_r^{(j)}E - A)^{-1}B] & \Re[(i\omega_r^{(j)}E - A)^{-1}E(i\omega_r E - A)^{-1}B] \\ \Im[(i\omega_r^{(j)}E - A)^{-1}B] & \Im[(i\omega_r^{(j)}E - A)^{-1}E(i\omega_r^{(j+1)}E - A)^{-1}B] \end{bmatrix}$ .
7:    $\tilde{W}_{r+1} \leftarrow \begin{bmatrix} \Re[(i\omega_r^{(j)}E - A)^{-*}C^*] & \Re[(i\omega_r^{(j)}E - A)^{-*}E(i\omega_r^{(j)}E - A)^{-*}C^*] \\ \Im[(i\omega_r^{(j)}E - A)^{-*}C^*] & \Im[(i\omega_r^{(j)}E - A)^{-*}E(i\omega_r^{(j)}E - A)^{-*}C^*] \end{bmatrix}$ .
8:    $V_{r+1} \leftarrow \text{orth}\left(\begin{bmatrix} V_{r+1} & \tilde{V}_{r+1} \end{bmatrix}\right)$  and  $W_{r+1} \leftarrow \text{orth}\left(\begin{bmatrix} W_{r+1} & \tilde{W}_{r+1} \end{bmatrix}\right)$ .
9:   % update the small system
      Form  $A_{r+1}, B_{r+1}, C_{r+1}, E_{r+1}$  using (3.3),
      and let  $S_{r+1} = (A_{r+1}, E_{r+1}, B_{r+1}, C_{r+1}, D)$ .
10: end for

```

4. Interpolation Properties of the Subspace Framework. Suppose that ω_r is a global maximizer of $\sigma_{r+1}(\omega; S_r^{\text{red}})$ by the termination of the refinement step, in which case the interpolation condition (3.9) holds due to (3.8). It can be shown that, assuming \mathcal{F} and \mathcal{F}_{r+1} are twice differentiable at S_r^{red} , indeed all of the first two derivatives of \mathcal{F} and \mathcal{F}_{r+1} are equal at S_r^{red} as well. To this end, let x_1, x_2 be any two entries of the matrix variables $A^{\text{red}}, B^{\text{red}}, C^{\text{red}}, D^{\text{red}}, E^{\text{red}}$ of \mathcal{F} and \mathcal{F}_{r+1} . Recalling

$$\begin{aligned} \mathcal{H}(i\omega; S^{\text{red}}) &= H(i\omega) - H(i\omega; S^{\text{red}}), \\ \mathcal{H}_{r+1}(i\omega; S^{\text{red}}) &= H_{r+1}(i\omega) - H(i\omega; S^{\text{red}}), \end{aligned}$$

and by employing (3.4), it is apparent that

$$(4.1) \quad \mathcal{H}(i\omega_r; S_r^{\text{red}}) = \mathcal{H}_{r+1}(i\omega_r; S_r^{\text{red}}),$$

$$(4.2) \quad \frac{\partial \mathcal{H}}{\partial y}(i\omega_r; S_r^{\text{red}}) = \frac{\partial \mathcal{H}_{r+1}}{\partial y}(i\omega_r; S_r^{\text{red}}),$$

$$(4.3) \quad \frac{\partial^2 \mathcal{H}}{\partial y \partial z}(i\omega_r; S_r^{\text{red}}) = \frac{\partial^2 \mathcal{H}_{r+1}}{\partial y \partial z}(i\omega_r; S_r^{\text{red}}),$$

for all $y, z \in \{\omega, x_1, x_2\}$. By exploiting

$$\mathcal{F}(S^{\text{red}}) = \sup_{\omega \geq 0} \sigma_{\max}(\mathcal{H}(i\omega; S^{\text{red}})), \quad \mathcal{F}_{r+1}(S^{\text{red}}) = \sup_{\omega \geq 0} \sigma_{\max}(\mathcal{H}_{r+1}(i\omega; S^{\text{red}})),$$

and using implicit differentiation

$$(4.4) \quad \begin{aligned} \frac{\partial \mathcal{F}}{\partial y}(S_r^{\text{red}}) &= \frac{\partial(\sigma_{\max} \circ \mathcal{H})}{\partial y}(i\omega_r; S_r^{\text{red}}) \\ &= \frac{\partial(\sigma_{\max} \circ \mathcal{H}_{r+1})}{\partial y}(i\omega_r; S_r^{\text{red}}) = \frac{\partial \mathcal{F}_{r+1}}{\partial y}(S_r^{\text{red}}) \end{aligned}$$

for $y \in \{x_1, x_2\}$, where the second equality is due to (4.2), as well as (4.1) implying the fact that $\mathcal{H}(i\omega_r; S_r^{\text{red}})$ and $\mathcal{H}_{r+1}(i\omega_r; S_r^{\text{red}})$ have the same left and right singular vectors. We remark that, for the first and third equalities above, we use the fact ω_r is a global maximizer of $\sigma(\omega; S_r^{\text{red}}) = \sigma_{\max}(\mathcal{H}(i\omega; S_r^{\text{red}}))$ and $\sigma_{r+1}(\omega; S_r^{\text{red}}) = \sigma_{\max}(\mathcal{H}_{r+1}(i\omega; S_r^{\text{red}}))$, respectively.

Moreover, for any $y, z \in \{\omega, x_1, x_2\}$, we have

$$\frac{\partial^2(\sigma_{\max} \circ \mathcal{H})}{\partial y \partial z}(i\omega_r; S_r^{\text{red}}) = \frac{\partial^2(\sigma_{\max} \circ \mathcal{H}_{r+1})}{\partial y \partial z}(i\omega_r; S_r^{\text{red}})$$

due to (4.2) and (4.3) combined with the fact that $\mathcal{H}(i\omega_r; S_r^{\text{red}})$, $\mathcal{H}_{r+1}(i\omega_r; S_r^{\text{red}})$ have the same singular values and vectors due to (4.1). Consequently,

$$(4.5) \quad \begin{aligned} \frac{\partial^2 \mathcal{F}}{\partial y \partial z}(S_r^{\text{red}}) &= \frac{\partial^2(\sigma_{\max} \circ \mathcal{H})}{\partial y \partial z}(i\omega_r; S_r^{\text{red}}) + \frac{\partial^2(\sigma_{\max} \circ \mathcal{H})}{\partial y \partial \omega}(i\omega_r; S_r^{\text{red}}) \times \\ &\quad \left\{ -\frac{\partial^2(\sigma_{\max} \circ \mathcal{H})}{\partial \omega \partial z}(i\omega_r; S_r^{\text{red}}) \Big/ \frac{\partial^2(\sigma_{\max} \circ \mathcal{H})}{\partial^2 \omega}(i\omega_r; S_r^{\text{red}}) \right\} \\ &= \frac{\partial^2(\sigma_{\max} \circ \mathcal{H}_{r+1})}{\partial y \partial z}(i\omega_r; S_r^{\text{red}}) + \frac{\partial^2(\sigma_{\max} \circ \mathcal{H}_{r+1})}{\partial y \partial \omega}(i\omega_r; S_r^{\text{red}}) \times \\ &\quad \left\{ -\frac{\partial^2(\sigma_{\max} \circ \mathcal{H}_{r+1})}{\partial \omega \partial z}(i\omega_r; S_r^{\text{red}}) \Big/ \frac{\partial^2(\sigma_{\max} \circ \mathcal{H}_{r+1})}{\partial^2 \omega}(i\omega_r; S_r^{\text{red}}) \right\} \\ &= \frac{\partial^2 \mathcal{F}_{r+1}}{\partial y \partial z}(S_r^{\text{red}}) \end{aligned}$$

for any $y, z \in \{x_1, x_2\}$.

5. A Quadratic Convergence Result Regarding Algorithm 3.1. In this section, we establish a result that indicates a quadratic convergence regarding the iterates of Algorithm 3.1 under a few assumptions, especially smoothness assumptions.

In this section and the next section, we denote with $\mathcal{D}^{r,m,p}$ the set of consisting of every descriptor system S^{red} of order r and index at most one with semi-simple poles, m inputs, p outputs. Throughout this section, we make use of the vectorization $\mathcal{V}(S^{\text{red}})$ of the system $S^{\text{red}} = (A^{\text{red}}, E^{\text{red}}, B^{\text{red}}, C^{\text{red}}, D^{\text{red}})$ defined as

$$(5.1) \quad \mathcal{V}(S^{\text{red}}) := \left[\text{vec}(A^{\text{red}})^T \text{vec}(E^{\text{red}})^T \text{vec}(B^{\text{red}})^T \text{vec}(C^{\text{red}})^T \text{vec}(D^{\text{red}})^T \right]^T,$$

where $\text{vec}(M)$ denotes the vector obtained by stacking up the columns of matrix M . The gradients $\nabla \mathcal{F}(S^{\text{red}})$ and $\nabla \mathcal{F}_{r+1}(S^{\text{red}})$ are vectors formed of the first partial derivatives of $\mathcal{F}(S^{\text{red}})$ and $\mathcal{F}_{r+1}(S^{\text{red}})$ based on the ordering of the variables, i.e., the entries of $A^{\text{red}}, E^{\text{red}}, B^{\text{red}}, C^{\text{red}}, D^{\text{red}}$, in the vectorization in (5.1). Similarly, $\nabla^2 \mathcal{F}(S^{\text{red}})$ and $\nabla^2 \mathcal{F}_{r+1}(S^{\text{red}})$ denote the Hessians of $\mathcal{F}(S^{\text{red}})$ and $\mathcal{F}_{r+1}(S^{\text{red}})$ based on the ordering of the variables according to (5.1).

We assume that there are two consecutive iterates S_r^{red} and S_{r+1}^{red} of Algorithm 3.1 that are sufficiently close to a local maximizer S_*^{red} of $\mathcal{F}(S^{\text{red}})$. Moreover, we silently assume throughout that the interpolation properties in (4.4) and (4.5) hold at S_r^{red} . We also keep the assumption stated below that guarantees that $\mathcal{F}(S^{\text{red}})$ is real analytic at S_*^{red} .

ASSUMPTION 5.1. *The maximum of $\sigma(\omega; S_*^{\text{red}})$ over all $\omega \geq 0$ is attained at a unique ω_* . Furthermore, $\sigma(\omega_*; S_*^{\text{red}}) = \sigma_{\max}(\mathcal{H}(i\omega_*; S_*^{\text{red}}))$ is a simple singular value of $\mathcal{H}(i\omega_*; S_*^{\text{red}})$.*

An assumption regarding the smoothness of $\mathcal{F}_{r+1}(S^{\text{red}})$ that we rely on is given next. Recalling $\|v\|_2$ for a vector v denotes the 2-norm of v , we make use of the distance $\|\tilde{S}^{\text{red}} - \hat{S}^{\text{red}}\| := \|\mathcal{V}(\tilde{S}^{\text{red}}) - \mathcal{V}(\hat{S}^{\text{red}})\|$ for systems $\tilde{S}^{\text{red}}, \hat{S}^{\text{red}} \in \mathcal{D}^{r,m,p}$, and the ball $B(\hat{S}^{\text{red}}, \delta) := \{\tilde{S}^{\text{red}} \in \mathcal{D}^{r,m,p} \mid \|\tilde{S}^{\text{red}} - \hat{S}^{\text{red}}\| < \delta\}$ for a system $\hat{S}^{\text{red}} \in \mathcal{D}^{r,m,p}$ and positive real number δ .

ASSUMPTION 5.2.

- (i) *For every $S^{\text{red}} \in B(S_r^{\text{red}}, \delta_r)$ with $\delta_r := \|S_{r+1}^{\text{red}} - S_r^{\text{red}}\|$ the following conditions hold:*
 - *The maximum of $\sigma_{r+1}(\omega; S^{\text{red}})$ over all $\omega \geq 0$ is attained at a unique $\bar{\omega}$.*
 - *The singular value $\sigma(\bar{\omega}; S^{\text{red}}) = \sigma_{\max}(\mathcal{H}_{r+1}(i\bar{\omega}; S^{\text{red}}))$ of $\mathcal{H}_{r+1}(i\bar{\omega}; S^{\text{red}})$ is simple.*
- (ii) *Moreover, all of the third derivatives of $\mathcal{F}_{r+1}(S^{\text{red}})$ can be bounded by quantities independent of r at all $S^{\text{red}} \in B(S_r^{\text{red}}, \delta_r)$.*

We remark that part (i) of Assumption 5.2 guarantees that $\mathcal{F}_{r+1}(S^{\text{red}})$ is real-analytic in the ball $B(S_r^{\text{red}}, \delta_r)$, and so three times differentiable in this ball, which we depend on in part (ii) of Assumption 5.2.

We state and prove the main result that relates $\|S_r^{\text{red}} - S_*^{\text{red}}\|$ and $\|S_{r+1}^{\text{red}} - S_*^{\text{red}}\|$ below.

THEOREM 5.3. *Suppose that two consecutive iterates S_r^{red} and S_{r+1}^{red} of Algorithm 3.1 are sufficiently close to a local maximizer S_*^{red} of $\mathcal{F}(S^{\text{red}})$. Moreover, suppose Assumptions 5.1, 5.2 hold, and $\nabla^2 \mathcal{F}(S_*^{\text{red}})$ is invertible. Then there is a constant C independent of r such that*

$$\|S_{r+1}^{\text{red}} - S_*^{\text{red}}\| \leq C \cdot \|S_r^{\text{red}} - S_*^{\text{red}}\|^2.$$

Proof. By continuity $\sigma(\omega; S^{\text{red}}) = \sigma_{\max}(\mathcal{H}(i\omega; S^{\text{red}}))$ remains simple at all ω and $S^{\text{red}} \in \mathcal{D}^{r,m,p}$ such that ω is sufficiently close to ω_* and S^{red} is sufficiently close to S_*^{red} , where ω_* is as in Assumption 5.1. Thus, by the analytic implicit function theorem, there is $\tilde{\delta} > 0$ such that $\mathcal{F}(S^{\text{red}})$ is real analytic at all $S^{\text{red}} \in B(S_*^{\text{red}}, \tilde{\delta})$ (see, e.g., [21, Lemma 16] for the details in the analogous context of the distance instability). By the assumption that $\nabla^2 \mathcal{F}(S_*^{\text{red}})$ is invertible, and continuity of the second partial derivatives of $\mathcal{F}(S^{\text{red}})$ in $B(S_*^{\text{red}}, \tilde{\delta})$, the Hessian $\nabla^2 \mathcal{F}(S^{\text{red}})$ remains invertible in a ball $B(S_*^{\text{red}}, \delta)$ for some $\delta < \tilde{\delta}$. Furthermore, without loss of generality, we assume $S_r^{\text{red}}, S_{r+1}^{\text{red}}$ are close enough to S_*^{red} so that $S_r^{\text{red}}, S_{r+1}^{\text{red}} \in B(S_*^{\text{red}}, \delta)$, and the ball $B(S_r^{\text{red}}, \delta_r)$ in Assumption 5.2 is contained in $B(S_*^{\text{red}}, \delta)$. We let $\beta := \min_{S^{\text{red}} \in B(S_*^{\text{red}}, \delta)} \sigma_{\min}(\nabla^2 \mathcal{F}(S^{\text{red}})) > 0$, and note that $\nabla^2 \mathcal{F}(S^{\text{red}})$ is Lipschitz continuous in $B(S_*^{\text{red}}, \delta)$, say with the Lipschitz constant γ .

By an application of Taylor's theorem with integral remainder, we have

$$\begin{aligned}
(5.2) \quad 0 &= \nabla \mathcal{F}(S_*^{\text{red}}) \\
&= \nabla \mathcal{F}(S_r^{\text{red}}) + \int_0^1 \nabla^2 \mathcal{F}(S_r^{\text{red}} + t(S_*^{\text{red}} - S_r^{\text{red}})) (\mathcal{V}(S_*^{\text{red}}) - \mathcal{V}(S_r^{\text{red}})) dt \\
&= \nabla \mathcal{F}(S_r^{\text{red}}) + \nabla^2 \mathcal{F}(S_r^{\text{red}}) (\mathcal{V}(S_*^{\text{red}}) - \mathcal{V}(S_r^{\text{red}})) + \mathcal{O}(\|S_*^{\text{red}} - S_r^{\text{red}}\|^2),
\end{aligned}$$

where, for the third equality, we have used the Lipschitz continuity of $\nabla^2 \mathcal{F}(S^{\text{red}})$ in $B(S_*^{\text{red}}, \delta)$. Additionally, by Taylor's theorem with second order Lagrange remainder,

$$\begin{aligned}
(5.3) \quad 0 &= \nabla \mathcal{F}_{r+1}(S_{r+1}^{\text{red}}) \\
&= \nabla \mathcal{F}_{r+1}(S_r^{\text{red}}) + \nabla^2 \mathcal{F}_{r+1}(S_r^{\text{red}}) (\mathcal{V}(S_{r+1}^{\text{red}}) - \mathcal{V}(S_r^{\text{red}})) + \mathcal{O}(\|S_{r+1}^{\text{red}} - S_r^{\text{red}}\|^2) \\
&= \nabla \mathcal{F}(S_r^{\text{red}}) + \nabla^2 \mathcal{F}(S_r^{\text{red}}) (\mathcal{V}(S_{r+1}^{\text{red}}) - \mathcal{V}(S_r^{\text{red}})) + \mathcal{O}(\|S_{r+1}^{\text{red}} - S_r^{\text{red}}\|^2),
\end{aligned}$$

where the third equality is due to $\nabla \mathcal{F}_{r+1}(S_r^{\text{red}}) = \nabla \mathcal{F}(S_r^{\text{red}})$ and $\nabla^2 \mathcal{F}_{r+1}(S_r^{\text{red}}) = \nabla^2 \mathcal{F}(S_r^{\text{red}})$, that are consequences of (4.4) and (4.5).

By employing (5.3) in (5.2), we deduce

$$\nabla^2 \mathcal{F}(S_r^{\text{red}}) (\mathcal{V}(S_*^{\text{red}}) - \mathcal{V}(S_{r+1}^{\text{red}})) = \mathcal{O}(\|S_{r+1}^{\text{red}} - S_r^{\text{red}}\|^2) + \mathcal{O}(\|S_*^{\text{red}} - S_r^{\text{red}}\|^2).$$

Finally, noting $\|\nabla^2 \mathcal{F}(S_r^{\text{red}}) (\mathcal{V}(S_*^{\text{red}}) - \mathcal{V}(S_{r+1}^{\text{red}}))\| \geq \beta \|S_*^{\text{red}} - S_{r+1}^{\text{red}}\|$, from the last equality we obtain

$$\|S_{r+1}^{\text{red}} - S_*^{\text{red}}\| \leq \mathcal{O}(\|S_r^{\text{red}} - S_*^{\text{red}}\|^2)$$

as desired. \square

6. Dealing with Asymptotic Stability Constraints. In many applications, the reduced order system sought $S^{\text{red}} = (A^{\text{red}}, E^{\text{red}}, B^{\text{red}}, C^{\text{red}}, D^{\text{red}})$ of order r not only is close with respect to the \mathcal{L}_∞ norm, but may also be required to be asymptotically stable. As we search through reduced order systems of index at most one, the asymptotic stability requirement is equivalent to the condition $\alpha(A^{\text{red}}, E^{\text{red}}) < 0$, where $\alpha(A^{\text{red}}, E^{\text{red}})$ is the spectral abscissa of the pencil $L(s) = A^{\text{red}} - sE^{\text{red}}$ defined by

$$\alpha(A^{\text{red}}, E^{\text{red}}) := \max \{ \text{Re}(z) \mid z \in \mathbb{C} \text{ s.t. } \det(A - zE) = 0 \}.$$

In this setting, with $\mathcal{F}(S^{\text{red}})$ defined as in (2.1), rather than the unconstrained minimization of $\mathcal{F}(S^{\text{red}})$ over all descriptor systems $S^{\text{red}} \in \mathcal{D}^{r,m,p}$, it may be desirable to solve the constrained minimization problem

$$(6.1) \quad \min \{ \mathcal{F}(S^{\text{red}}) : S^{\text{red}} \in \mathcal{D}^{r,m,p} \text{ s.t. } \alpha(A^{\text{red}}, E^{\text{red}}) \leq \beta \}$$

for a prescribed negative real number β , where $\mathcal{D}^{r,m,p}$ denotes the set of all descriptor systems of order r and index at most one with semi-simple poles, m inputs, p outputs.

Extension of the proposed subspace framework, that is Algorithm 3.1, to deal with the constrained minimization problem in (6.1) rather than the unconstrained minimization of $\mathcal{F}(S^{\text{red}})$ is straightforward. The only difference in Algorithm 3.1 is in line 5, where S_r^{red} is no longer a minimizer of $\mathcal{F}_r(S^{\text{red}})$, but instead a minimizer of the constrained problem

$$(6.2) \quad \min \{ \mathcal{F}_r(S^{\text{red}}) : S^{\text{red}} \in \mathcal{D}^{r,m,p} \text{ s.t. } \alpha(A^{\text{red}}, E^{\text{red}}) \leq \beta \}$$

for the reduced function $\mathcal{F}_r(S^{\text{red}})$ as in (3.2). The problem in (6.2) involves only the small systems S_r as well as S^{red} , and is solvable by means of Newton-method based approaches. Such a Newton-method based approach makes use of the gradient of the constraint function $\mathcal{C}(S^{\text{red}}) := \alpha(A^{\text{red}}, E^{\text{red}}) - \beta$, in addition to the gradient of the objective $\mathcal{F}_r(S^{\text{red}})$. Let λ be the rightmost eigenvalue of the pencil $L(s) = A^{\text{red}} - sE^{\text{red}}$ with u and v denoting a pair of corresponding left and right eigenvectors normalized so that $u^* E^{\text{red}} v = 1$, and assume λ is a simple eigenvalue and the unique rightmost eigenvalue of $L(s)$, which ensures the differentiability of $\mathcal{C}(S^{\text{red}})$. Then, by differentiating the equation $A^{\text{red}} v = \lambda E^{\text{red}} v$ with respect to the entries of A^{red} and E^{red} and multiplying with u^* from left, the partial derivatives of $\mathcal{C}(S^{\text{red}})$ are given by

$$\frac{\partial \mathcal{C}}{A_{ij}^{\text{red}}}(S^{\text{red}}) = \Re(\bar{u}_i v_j), \quad \frac{\partial \mathcal{C}}{E_{jj}^{\text{red}}}(S^{\text{red}}) = -\Re(\lambda \bar{u}_j v_j),$$

where A_{ij}^{red} is the subdiagonal, superdiagonal or diagonal entry of the matrix variable A^{red} at position (i, j) , and E_{jj}^{red} is the diagonal entry of E^{red} at position (j, j) .

We remark that, assuming ω_r is again a global minimizer of $\sigma_{r+1}(\omega; S_r^{\text{red}})$ after the refinement step, the interpolation properties

$$\mathcal{F}(S_r^{\text{red}}) = \mathcal{F}_{r+1}(S_r^{\text{red}}), \quad \nabla \mathcal{F}(S_r^{\text{red}}) = \nabla \mathcal{F}_{r+1}(S_r^{\text{red}}), \quad \text{and} \quad \nabla^2 \mathcal{F}(S_r^{\text{red}}) = \nabla^2 \mathcal{F}_{r+1}(S_r^{\text{red}})$$

still hold. Moreover, if a logarithmic barrier approach is adopted for the solution of the constrained problems, then, in essence, constrained problems are turned into unconstrained problems by lifting the constraints to the objective via the logarithmic barrier functions

$$\begin{aligned} L^\mu(S^{\text{red}}) &= \mathcal{F}(S^{\text{red}}) - \mu \cdot \log(\beta - \alpha(A^{\text{red}}, E^{\text{red}})), \\ L_r^\mu(S^{\text{red}}) &= \mathcal{F}_r(S^{\text{red}}) - \mu \cdot \log(\beta - \alpha(A^{\text{red}}, E^{\text{red}})) \end{aligned}$$

associated with problems (6.1), (6.2), respectively, where $\log(\cdot)$ denotes the natural logarithm of its parameter, μ is a positive real number and represent the barrier parameter. In this case, the interpolation properties extend to the logarithmic barrier functions as well. In particular, we have

$$L^\mu(S_r^{\text{red}}) = L_{r+1}^\mu(S_r^{\text{red}}), \quad \nabla L^\mu(S_r^{\text{red}}) = \nabla L_{r+1}^\mu(S_r^{\text{red}}), \quad \text{and} \quad \nabla^2 L^\mu(S_r^{\text{red}}) = \nabla^2 L_{r+1}^\mu(S_r^{\text{red}})$$

for every positive real number μ .

7. Practical Issues. Here we spell out a few practical issues regarding Algorithm 3.1 such as how we form the initial systems S_0, S_0^{red} , when we terminate, the details of bases for projection subspaces, solutions of reduced \mathcal{L}_∞ -norm minimization problems, and \mathcal{L}_∞ -norm computations.

7.1. Initialization. The initial subspaces $\mathcal{V}_0, \mathcal{W}_0$ (in line 1 of Algorithm 3.1) are chosen so that H_0 , the transfer function of S_0 , interpolates H at the imaginary parts of a prescribed number of dominant poles of H . Formally, for a prescribed ℓ , let $s_1, \dots, s_\ell \in \mathbb{C}$ be the most dominant ℓ poles of H with nonnegative imaginary parts (i.e., only the dominant poles with nonnegative imaginary parts are taken into consideration, as the poles of H appear in complex conjugate pairs such that any two

complex conjugate poles have the same dominance metric), we set

$$\begin{aligned}\mathcal{V}_0 &= \bigoplus_{k=1}^{\ell} \bigoplus_{j=0}^1 \left\{ \Re \left[\{(A - i\Im s_k E)^{-1} E\}^j (A - i\Im s_k E)^{-1} B \right] \right. \\ &\quad \left. \bigoplus \Im \left[\{(A - i\Im s_k E)^{-1} E\}^j (A - i\Im s_k E)^{-1} B \right] \right\}, \\ \mathcal{W}_0 &= \bigoplus_{k=1}^{\ell} \bigoplus_{j=0}^1 \left\{ \Re \left[C(A - i\Im s_k E)^{-1} \{E(A - i\Im s_k E)^{-1}\}^j \right]^* \right. \\ &\quad \left. \bigoplus \Im \left[C(A - i\Im s_k E)^{-1} \{E(A - i\Im s_k E)^{-1}\}^j \right]^* \right\}.\end{aligned}$$

Theorem 3.1 ensures that

$$\begin{aligned}H(i\Im s_k) &= H_0(i\Im s_k), & H^{(j)}(i\Im s_k) &= H_0^{(j)}(i\Im s_k) \\ H(-i\Im s_k) &= H_0(-i\Im s_k), & H^{(j)}(-i\Im s_k) &= H_0^{(j)}(-i\Im s_k)\end{aligned}$$

for $j = 1, 2, 3$ and $k = 1, \dots, \ell$.

Additionally, at every subspace iteration with $r > 0$, an initial point is needed for the solution of the minimization problem in line 5 of Algorithm 3.1 regardless of how it is solved, e.g., via gradient descent or BFGS. This initialization carries significance, as it affects which local minimizer of \mathcal{F}_r is to be converged. At a subspace iteration with $r > 0$, the minimizer is initialized with the optimal reduced system from the previous iteration, that is with S_{r-1}^{red} . Initial S_0^{red} of order r must be supplied to Algorithm 3.1. We set S_0^{red} as either

- the model of order r obtained from an application of the balanced truncation approach, or
- the model of order r whose transfer function interpolates H at the imaginary parts of a prescribed number of dominant poles of H .

For the latter choice, we remark that the number of dominant poles used to form S_0 is strictly greater than the number of dominant poles used to form this initial model S_0^{red} for the minimizer. For either choice, we make sure $\dim \mathcal{V}_0 = \dim \mathcal{W}_0 > r$ by using sufficiently many dominant poles of H when forming $S_0 = (A_0, E_0, B_0, C_0, D)$.

An issue that requires attention is that $S_0^{\text{red}} = (A_0^{\text{red}}, E_0^{\text{red}}, B_0^{\text{red}}, C_0^{\text{red}}, D_0^{\text{red}})$ must be such that A_0^{red} is tridiagonal and E_0^{red} is diagonal, whereas the balanced truncation or the interpolatory approach yields the system $(\widehat{A}, \widehat{E}, \widehat{B}, \widehat{C}, \widehat{D})$ of order r such that \widehat{A} and \widehat{E} do not necessarily have these structures. Let us suppose \widehat{E} is invertible. Then we can first compute the eigenvalues of the $r \times r$ pencil $\widehat{L}(s) = \widehat{A} - s\widehat{E}$, and form a block diagonal real matrix T_2 with 2×2 and 1×1 blocks along its diagonal that have the same eigenvalues as \widehat{L} . The 2×2 and 1×1 blocks of T_2 on its diagonal correspond to a conjugate pair of complex eigenvalues and real eigenvalues of \widehat{L} , respectively. Here we remark that T_2 is an $r \times r$ matrix. Hence, we can compute its eigenvalue decomposition

$$T_2 = V_2 \Lambda V_2^{-1}$$

for a diagonal matrix Λ and invertible V_2 at ease. We also have the eigenvalue decomposition

$$\widehat{E}^{-1} \widehat{A} = V \Lambda V^{-1}$$

at hand. Note that the middle factors in eigenvalue decompositions above are the same, as T_2 has the same eigenvalues as the pencil \widehat{L} , which in turn has the same

eigenvalues as $\widehat{E}^{-1}\widehat{A}$. But then

$$\begin{aligned}\widehat{H}(s) &:= \widehat{C}(s\widehat{E} - \widehat{A})^{-1}\widehat{B} + \widehat{D} = \widehat{C}(sI - \widehat{E}^{-1}\widehat{A})^{-1}\widehat{E}^{-1}\widehat{B} + \widehat{D} \\ &= \widehat{C}(sI - V\Lambda V^{-1})^{-1}\widehat{E}^{-1}\widehat{B} + \widehat{D} \\ &= (\widehat{C}V)(sI - \Lambda)^{-1}(V^{-1}\widehat{E}^{-1}\widehat{B}) + \widehat{D} \\ &= (\widehat{C}V)(sI - V_2^{-1}T_2V_2)^{-1}(V^{-1}\widehat{E}^{-1}\widehat{B}) + \widehat{D} \\ &= (\widehat{C}VV_2^{-1})(sI - T_2)^{-1}(V_2V^{-1}\widehat{E}^{-1}\widehat{B}) + \widehat{D}.\end{aligned}$$

Hence, we can use

$$A_0^{\text{red}} := T_2, \quad E_0^{\text{red}} := I, \quad B_0^{\text{red}} := V_2V^{-1}\widehat{E}^{-1}\widehat{B}, \quad C_0^{\text{red}} := \widehat{C}VV_2^{-1}, \quad D_0^{\text{red}} := \widehat{D}_r$$

as the matrices of the initial system S_0^{red} .

7.2. Termination. The termination in line 9 of Algorithm 3.1 is determined based on the values of $\|H(\cdot) - H(\cdot; S_r^{\text{red}})\|_{\mathcal{L}_\infty}$ at two consecutive subspace iterations. The error $\|H(\cdot) - H(\cdot; S_r^{\text{red}})\|_{\mathcal{L}_\infty}$ is readily available at the r th subspace iteration after line 7, as

$$\|H(\cdot) - H(\cdot; S_r^{\text{red}})\|_{\mathcal{L}_\infty} = \sigma_{\max}(H(i\omega_r) - H(i\omega_r; S_r^{\text{red}})).$$

To be precise, we terminate at the r th subspace iteration in line 9 if $r \geq 1$ and

$$(7.1) \quad \left| \|H(\cdot) - H(\cdot; S_r^{\text{red}})\|_{\mathcal{L}_\infty} - \|H(\cdot) - H(\cdot; S_{r-1}^{\text{red}})\|_{\mathcal{L}_\infty} \right| \leq \text{tol} \|H(\cdot) - H(\cdot; S_r^{\text{red}})\|_{\mathcal{L}_\infty}$$

for a prescribed tolerance tol .

The termination condition for the minimizer to solve the minimization problem in line 5 of Algorithm 3.1 also requires some care. Recall that the objective \mathcal{F}_r here is nonsmooth, and, as a result, the norms of the gradients of \mathcal{F}_r at the iterates generated by the minimizer do not have to converge to zero. Instead, the minimizer is terminated if the line-search fails (to return a point that causes sufficient decrease), or the decrease in \mathcal{F}_r at two consecutive iterates is less than $\varepsilon \cdot \text{tol}$ in a relative sense, where tol is as in (7.1) and ε is a real number in $(0, 0.5)$.

As for the termination condition of the refinement step (i.e., the condition in line 3 of Algorithm 3.2) employed in practice, we rely on

$$|\omega_r^{(j)} - \omega_r| \leq \text{tol} |\omega_r|$$

where tol is again the tolerance in (7.1).

7.3. Orthonormalization of the Bases for the Subspaces. Keeping the bases for the subspaces $\mathcal{V}_r, \mathcal{W}_r$ (i.e., the columns of V_r, W_r) orthonormal improves the robustness of the algorithm against the rounding errors. For instance, then the system matrices A_r, B_r, C_r, E_r can be formed more accurately in the presence of rounding errors.

This orthonormality property of the bases is attained in line 13 of Algorithm 3.1, as well as line 8 of Algorithm 3.2. In line 13 of Algorithm 3.1, V_r and W_r are already orthonormal bases for \mathcal{V}_r and \mathcal{W}_r . The expansion directions $\widetilde{V}_{r+1}, \widetilde{W}_{r+1}$ to be included in the subspaces to obtain the expanded subspaces $\mathcal{V}_{r+1}, \mathcal{W}_{r+1}$ has to be orthonormalized with respect to the existing orthonormal bases V_r, W_r . This is achieved in practice by executing

$$(7.2) \quad \widetilde{V}_{r+1} \leftarrow \widetilde{V}_{r+1} - V_r(V_r^T \widetilde{V}_{r+1}) \quad \text{and} \quad \widetilde{W}_{r+1} \leftarrow \widetilde{W}_{r+1} - W_r(W_r^T \widetilde{W}_{r+1}).$$

Near convergence the interpolation points $i\omega_r$ start not changing by much in consecutive iterations. This results in the new expansion directions $\tilde{V}_{r+1}, \tilde{W}_{r+1}$ that are nearly contained in the existing subspaces $\mathcal{V}_r, \mathcal{W}_r$. In this case, the orthonormalization in (7.2) of $\tilde{V}_{r+1}, \tilde{W}_{r+1}$ with respect to existing V_r, W_r suffers from cancellation type rounding errors. Applying the orthonormalization in (7.2) several times improves the accuracy, and result in directions $\tilde{\tilde{V}}_{r+1}, \tilde{\tilde{W}}_{r+1}$ that are better orthonormalized against V_r, W_r . In practice, we apply (7.2) a few times, e.g., 3-4 times, then orthonormalize the resulting $\tilde{\tilde{V}}_{r+1}, \tilde{\tilde{W}}_{r+1}$ via the Gram-Schmidt procedure, and take $V_{r+1} = [V_r \ \tilde{\tilde{V}}_{r+1}]$, $W_{r+1} = [W_r \ \tilde{\tilde{W}}_{r+1}]$ as the matrices whose columns form orthonormal bases for $\mathcal{V}_{r+1}, \mathcal{W}_{r+1}$. In line 8 of Algorithm 3.2, the columns of V_{r+1} and W_{r+1} are similarly orthonormalized. We ultimately use V_{r+1}, W_{r+1} when forming the system S_{r+1} in line 14 of Algorithm 3.1.

7.4. Solution of the Reduced \mathcal{L}_∞ -Norm Minimization Problem. We use BFGS to minimize the reduced \mathcal{L}_∞ objective $\mathcal{F}_r(S^{\text{red}})$ in line 5 of Algorithm 3.1. To be precise, we have explored two alternatives here; a small variation of a Matlab implementation of a line-search BFGS due to Michael L. Overton making use of weak Wolfe conditions, and GRANSO [11]. The former is only meant for unconstrained problems when we do not impose the asymptotic stability constraints described in Section 7, whereas the asymptotic stability constraints in Section 7 are also incorporated into this optimization when we use GRANSO.

7.5. Computation of the \mathcal{L}_∞ Norm. Algorithm 3.1 in line 7 requires the computation of the \mathcal{L}_∞ norm of a system whose order is the sum of the order of the original system S and r . If the original system does not have large order, we use the built-in `norm` command in Matlab for these \mathcal{L}_∞ -norm computations. Otherwise, if the original system has large order, we use the subspace framework introduced in [1] for the large-scale \mathcal{L}_∞ -norm computations. Additionally, the minimization of the reduced \mathcal{L}_∞ objective in line 5 via BFGS requires small-scale \mathcal{L}_∞ -norm computations, which we carry out using the `norm` command in Matlab.

8. Numerical Results. In this section, we report the results of numerical experiments performed with a Matlab implementation of Algorithm 3.1 taking also the practical issues indicated in the previous section into account. The first two subsections §8.1 and §8.2 below concern experiments on rather smaller order systems, §8.3 concerns experiments on a system of medium order, while the results of experiments on several large-order systems are reported in §8.4. All of the experiments are conducted in Matlab 2020b on an an iMac with Mac OS 12.1 operating system, Intel[®] Core[™] i5-9600K CPU and 32GB RAM.

The numerical experiments are performed using the variation of the Matlab implementation of BFGS due to Michael L. Overton for the solution of the reduced \mathcal{L}_∞ -norm minimization problems. Hence, the asymptotic stability constraints are not imposed. The original systems in all of the experiments in §8.1-8.3 concerning small- to medium-order systems are asymptotically stable, and the computed optimal reduced systems in these examples also turn out to be asymptotically stable. The tolerance `tol` for termination (discussed in §7.2) is set equal to 10^{-8} in §8.1-8.3, and 10^{-6} in §8.4.

For comparison or initialization purposes, some of the numerical experiments involve the application of the balanced truncation for which we use the Matlab toolbox MORLAB [7], in particular the routine `m1_ct_dss_bt` or `m1_ct_ss_bt` depending on whether the system at hand is a descriptor system or more specifically a linear

time-invariant system. Moreover, the Hankel singular values computed for smaller systems for comparison purposes are retrieved by calling the built-in routine `hankelsv` in Matlab. As the first three subsections concern the model reduction of relatively smaller systems, the built-in routine `norm` is employed in line 7 of Algorithm 3.1 for \mathcal{L}_∞ -norm computations, while the subspace framework in [1] is employed for this purpose in §8.4 that concerns the model reduction of descriptor systems with large order.

8.1. ISS Example. We start with the `iss` example of order $n = 270$ that is also considered when optimizing the objective \mathcal{F} directly in Section 2. As before, we seek the nearest reduced descriptor system of order $r = 12$ with respect to the \mathcal{L}_∞ norm. An application of Algorithm 3.1 with the initial estimate S_0^{red} produced by the balanced truncation terminates when $r = 6$. The error $\mathcal{F}(S_\star^{\text{red}}) = 0.0022516$ of the estimate S_\star^{red} returned is nearly half of the error $\mathcal{F}(S_0^{\text{red}}) = 0.0044701$ of the initial estimate S_0^{red} . The optimal reduced system S_\star^{red} is indeed slightly better than the estimate returned by the direct optimization at which the objective \mathcal{F} takes the value 0.0024154. Yet the total runtime is about 66 seconds, much shorter than 500 seconds, roughly the time required by the direct optimization. The local optimality of $S_\star^{\text{red}} = (A_\star^{\text{red}}, B_\star^{\text{red}}, C_\star^{\text{red}}, D_\star^{\text{red}}, E_\star^{\text{red}})$ is apparent from Figure 8.1, which indicates an increase in the objective \mathcal{F} if one of the entries of one of $A_\star^{\text{red}}, B_\star^{\text{red}}, C_\star^{\text{red}}, D_\star^{\text{red}}, E_\star^{\text{red}}$ is modified. Moreover, the Hankel singular value σ_{r+1} for this example, a lower bound for the minimal error possible for any system of order r , is 0.0022353 smaller than $\mathcal{F}(S_\star^{\text{red}}) = 0.0022516$ only by a slim margin, so S_\star^{red} must be nearly optimal globally as well.

The largest singular values of the errors $\sigma_{\max}(H(i\omega) - H(i\omega; S_0^{\text{red}}))$ and $\sigma_{\max}(H(i\omega) - H(i\omega; S_\star^{\text{red}}))$ of the initial estimate S_0^{red} and the optimal estimate S_\star^{red} are plotted as functions of ω in Figure 8.2. The singular value error function $\sigma_{\max}(H(i\omega) - H(i\omega; S_\star^{\text{red}}))$ for the optimal S_\star^{red} is extremely flat, as indeed $\sigma_{\max}(H(i\omega) - H(i\omega; S_\star^{\text{red}})) \in [2.02 \cdot 10^{-3}, 2.26 \cdot 10^{-3}]$ at all ω . Furthermore, the error $\sigma_{\max}(H(i\omega) - H(i\omega; S_\star^{\text{red}}))$ is maximized at four distinct points marked by the circles on the right-hand plot. This indicates that the objective \mathcal{F} is not differentiable at the computed optimizer S_\star^{red} .

Information about the progress of Algorithm 3.1 is given in Table 8.1. We start with the reduced system S_0 of order 36 that interpolates the original system S of order 270 at three points on the imaginary axis, namely the imaginary parts of the most dominant three poles of S . At every iteration, if no refinement step is performed, the order of the reduced system S_r increases by $4m = 12$. Additionally, each refinement step results in an increase of $4m = 12$ in the order of S_r . We observe in the first column that the error $\mathcal{F}(S_r^{\text{red}})$ at the minimizer S_r^{red} of the reduced objective \mathcal{F}_r decays rapidly with respect to r . Total number of objective function evaluations is 492 (i.e., the sum of the function evaluations in the fifth columns), however the \mathcal{L}_∞ objective to be minimized involves the reduced system S_r rather than the full system S . For instance, the number of \mathcal{L}_∞ -norm computations performed are 105, 155, 123 at iterations $r = 1, 2, 3$. Yet, these \mathcal{L}_∞ -norm computations involve the reduced system S_r of order 72, 84, 120 for $r = 1, 2, 3$. Observe that the number of bfgs iterations eventually decrease at the later iterations, as the computed optimal S_r^{red} used as the initial estimate when minimizing \mathcal{F}_{r+1} becomes stationary, i.e., as the computed minimizer S_r^{red} of \mathcal{F}_r is also close to a minimizer of \mathcal{F}_{r+1} . Refinement steps are needed only at the initial iteration when $r = 0$ and when $r = 2$. No refinement step turns out to be necessary at the later iterations. This is a generic pattern which we observe in vast majority of examples we have experimented on.

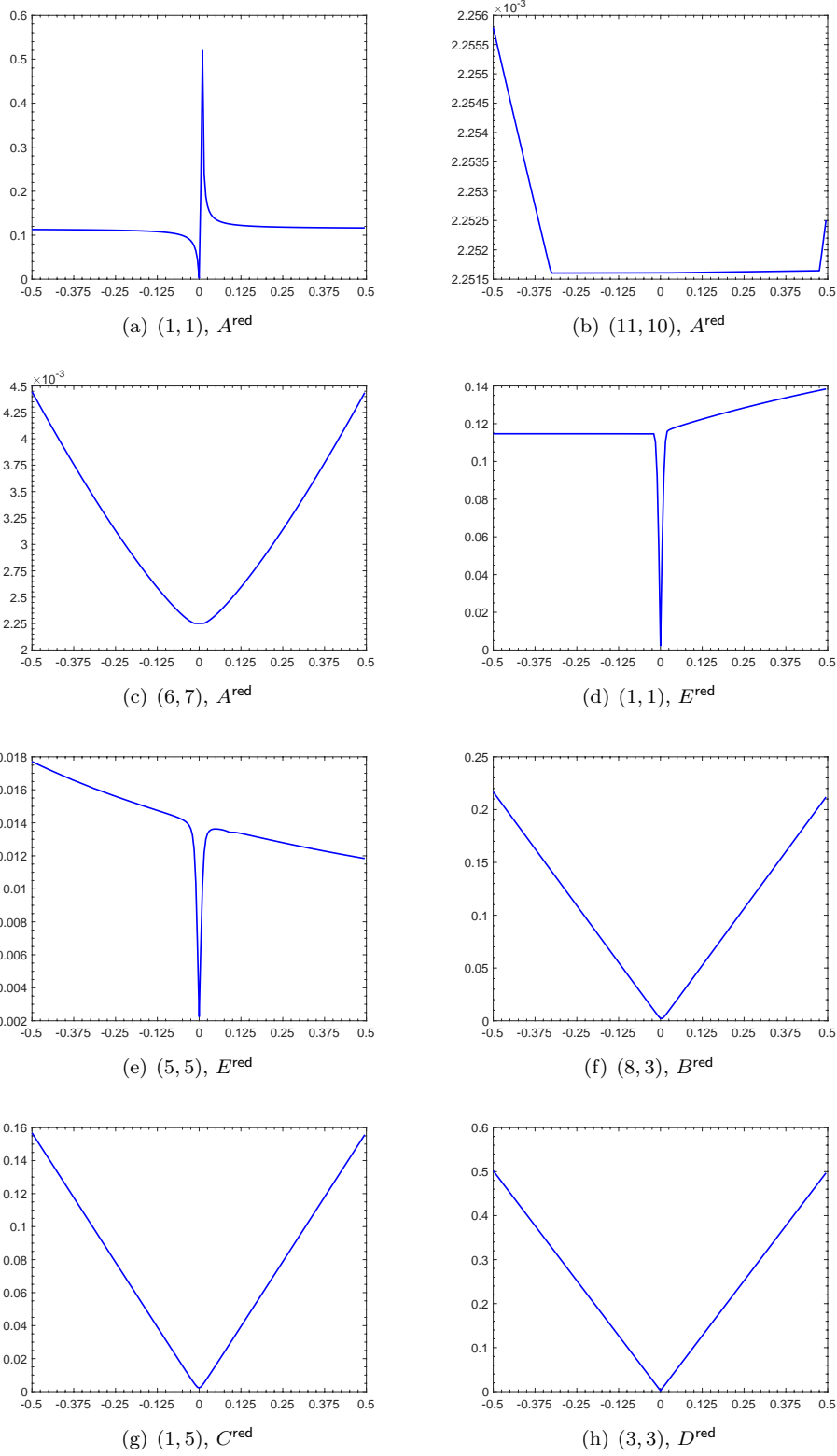


FIG. 8.1. The figure is similar to Figure 2.1 and concerns the *iss* example with $r = 12$. Only now the minimization of \mathcal{F} is performed using the subspace framework outlined in Algorithm 3.1. Specifically, each plot depicts \mathcal{F} as a function of the variation of one of the entries of one of A^{red} , B^{red} , C^{red} , D^{red} , E^{red} . Zero variation corresponds to the optimal reduced system by Algorithm 3.1.

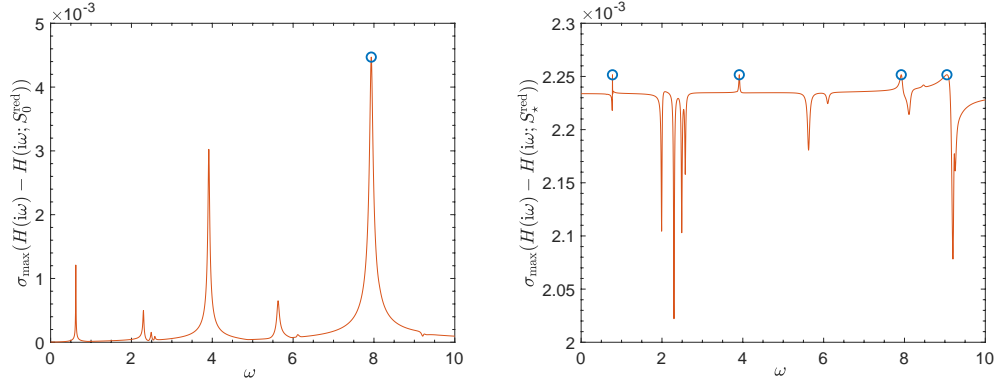


FIG. 8.2. The plots of $\sigma_{\max}(H(i\omega) - H(i\omega; S_0^{\text{red}}))$ (left) and $\sigma_{\max}(H(i\omega) - H(i\omega; S_\star^{\text{red}}))$ (right) as functions of ω for the *iss* example with $r = 12$, where S_0^{red} is the initial estimate, and S_\star^{red} is the optimal estimate computed by Algorithm 3.1. In each plot, the circles mark the points where the largest singular value function attains the largest value.

r	$\mathcal{F}(S_r^{\text{red}})$	red order	# bfgs iter	# fun evals	# refine
0	0.004470060020	36	—	—	2
1	0.003517059977	72	38	105	0
2	0.002259657400	84	45	155	2
3	0.002252138011	120	35	123	0
4	0.002251613679	132	11	48	0
5	0.002251609387	144	2	29	0
6	0.002251607779	156	2	32	—

TABLE 8.1

The iterates and information about the progress of Algorithm 3.1 on the *iss* example with $r = 12$. The columns of red order, # bfgs iter, # fun evals, and # refine list the order of the system S_r , number of bfgs iterations, number of objective function evaluations performed by bfgs, and number of refinement steps performed at the r th iteration.

8.2. CD Player Model. Our next example is the CD player model which is available in the SLICOT library. The model is a linear-time invariant system of order $n = 120$ and with $m = 2$ inputs and $p = 2$ outputs. The details of the model can be found in [10], and the references therein. Our primary purpose here is to compare on this example Algorithm 3.1 with the approach in [13] for \mathcal{H}_∞ model reduction based on rank-one modifications of the system matrices. As the approach in [13] is for SISO systems, the results are reported over there for this example but with only the second input and the first output. We follow the same practice here when applying our approach. The initial estimate S_0^{red} for a minimizer for Algorithm 3.1 is constructed using the balanced truncation. Moreover, the initial reduced system S_0 is of order 12, and is constructed so that it interpolates the full system S at the imaginary parts of its most dominant three poles.

The reduced systems S_\star^{red} of order $r = 2, 4, 6, 8, 10$ are computed using Algorithm 3.1. Table 8.2 lists the relative errors $\|H - H(\cdot; S_\star^{\text{red}})\|_{\mathcal{L}_\infty} / \|H\|_{\mathcal{L}_\infty}$ for the reduced system S_\star^{red} computed by various approaches. In particular, the columns of IHA, MBT, HNA are the reported results in [13, Table 4] by using the approach over there initialized with the model returned by IRKA, initialized with the model returned by

r	Alg. 3.1	IHA	MBT	HNA	BT	Lower Bnd
2	3.12×10^{-1}	3.66×10^{-1}	3.68×10^{-1}	3.35×10^{-1}	3.69×10^{-1}	1.95×10^{-1}
4	1.82×10^{-2}	2.14×10^{-2}	2.25×10^{-2}	2.00×10^{-2}	2.25×10^{-2}	1.13×10^{-2}
6	9.44×10^{-3}	1.04×10^{-2}	1.19×10^{-2}	1.23×10^{-2}	1.23×10^{-2}	6.79×10^{-3}
8	4.18×10^{-3}	4.85×10^{-3}	6.40×10^{-3}	5.99×10^{-3}	6.41×10^{-3}	3.20×10^{-3}
10	7.45×10^{-4}	8.99×10^{-4}	1.24×10^{-3}	1.08×10^{-3}	1.32×10^{-3}	5.86×10^{-4}

TABLE 8.2

This table concerns the “cd player model”. Relative errors $\|H - H(\cdot; S_{\star}^{\text{red}})\|_{\mathcal{L}_{\infty}}/\|H\|_{\mathcal{L}_{\infty}}$ are listed for the optimal estimate S_{\star}^{red} computed by various methods for finding reduced systems of order $r = 2, 4, 6, 8, 10$, as well as the lower bound $\sigma_{r+1}/\|H\|_{\mathcal{L}_{\infty}}$.

r	$\mathcal{F}(S_r^{\text{red}})$	red order	# bfgs iter	# fun evals	# refine
0	0.439972058849	12	—	—	1
1	0.291281639337	20	565	1479	0
2	0.287107598817	24	134	387	0
3	0.287107598817	28	1	32	—

TABLE 8.3

The iterates and information about the progress of Algorithm 3.1 on the “cd player model” for finding a reduced system of order $r = 8$. The columns represent quantities as in Table 8.1.

the balanced truncation, and the best Hankel norm approximation. Moreover, the columns of BT and Lower Bnd correspond to the relative error of the reduced model by the balanced truncation, and the theoretical lower bound $\sigma_{r+1}/\|H\|_{\mathcal{L}_{\infty}}$ for any reduced system of order r for the relative error, where σ_{r+1} is the $r + 1$ th largest Hankel singular value of the system. As can be seen in Table 8.2, our approach produces reduced systems with smaller errors compared to those produced by other approaches in all cases. The reduced systems produced by Algorithm 3.1 does not seem far away from global optimality either, as their errors are slightly greater than the theoretical lower bounds in terms of the Hankel singular values in the last column.

We give some details of Algorithm 3.1 applied to find a reduced system of order $r = 8$ in Figures 8.3 and 8.4, as well as in Table 8.3. In particular, Figure 8.3 confirms that the reduced system S_{\star}^{red} by Algorithm 3.1 is locally optimal, i.e., small variations in the entries of the system matrices cause increase in the \mathcal{L}_{∞} error objective. Figure 8.4 displays the error $\sigma_{\max}(H(i\omega) - H(i\omega; S_0^{\text{red}}))$ of the initial model, and the error $\sigma_{\max}(H(i\omega) - H(i\omega; S_{\star}^{\text{red}}))$ of the model by Algorithm 3.1 as functions of ω . Once again the error function for the optimal model S_{\star}^{red} is flatter, even if it is not as pronounced as for the `iss` example, compared to that for the initial model S_0^{red} . The error function $\sigma_{\max}(H(i\omega) - H(i\omega; S_{\star}^{\text{red}}))$ for the optimal model attains its maximum at five different ω values, which implies that the objective \mathcal{F} is not smooth at S_{\star}^{red} . As displayed in Table 8.3, the convergence occurs again quite rapidly; indeed four iterations are sufficient to reach prescribed accuracy and terminate. At each iteration, the order of the reduced system increases by $4m = 4$. Additionally, the refinement step performed in the initial iteration causes also an increase of $4m = 4$ in the order of the reduced system. Larger number of bfgs iterations are needed at iterations with $r = 1, 2$, when the objective involves reduced systems of order 20, 24, respectively. The total runtime is around 15 seconds, and the relative error at termination is $(\mathcal{F}(S_{\star}^{\text{red}}) := \|H - H(\cdot; S_{\star}^{\text{red}})\|_{\mathcal{L}_{\infty}})/\|H\|_{\mathcal{L}_{\infty}} = (2.87 \times 10^{-1})/(6.87 \times 10^1) = 4.18 \times 10^{-3}$.

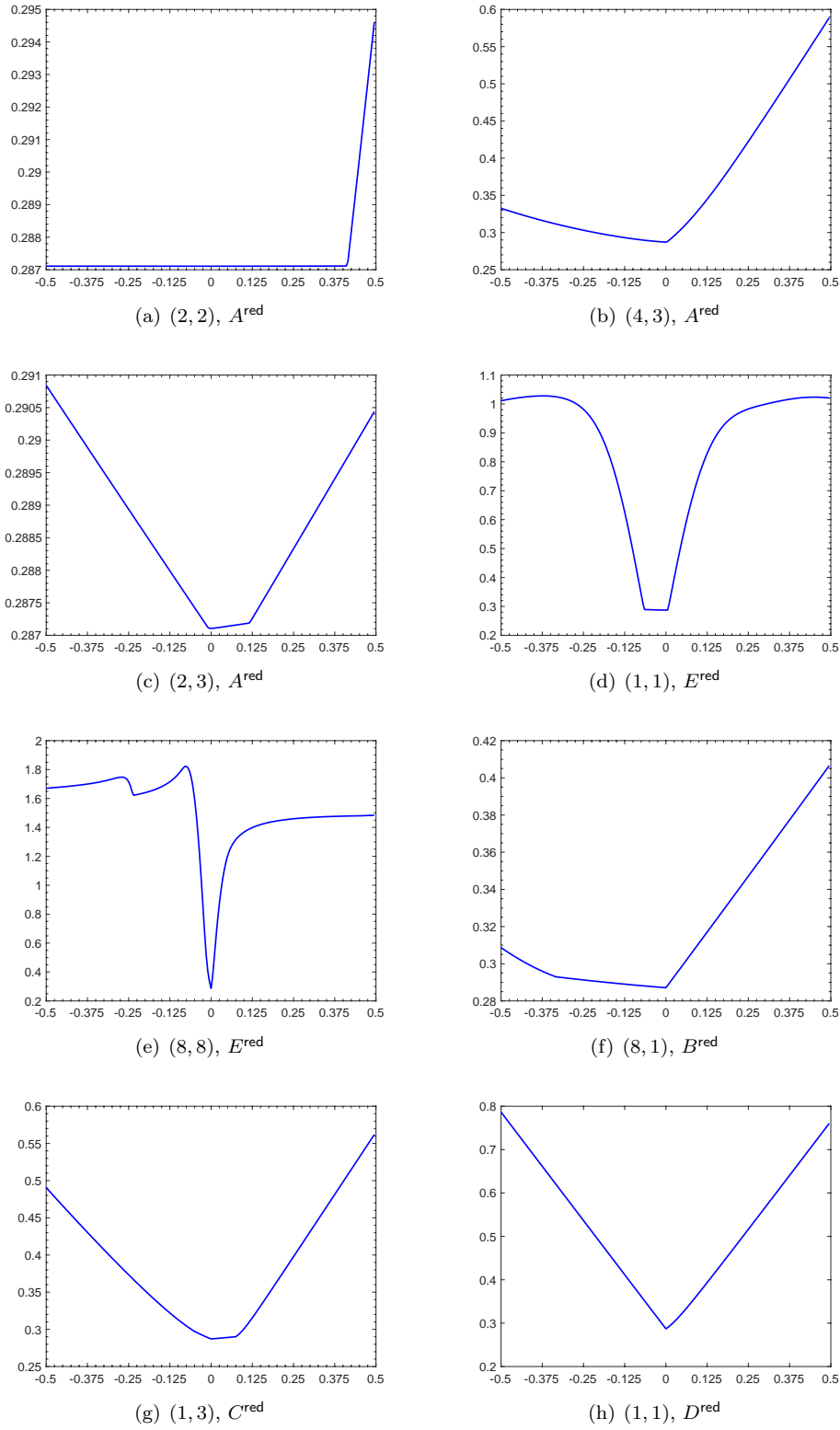


FIG. 8.3. The figure is analogous to Figure 2.1, but concerns the “cd player model” with $r = 8$. Each plot depicts \mathcal{F} as a function of the variation of one of the entries of one of A^{red} , B^{red} , C^{red} , D^{red} , E^{red} . Zero variation corresponds to the optimal reduced system by Algorithm 3.1.

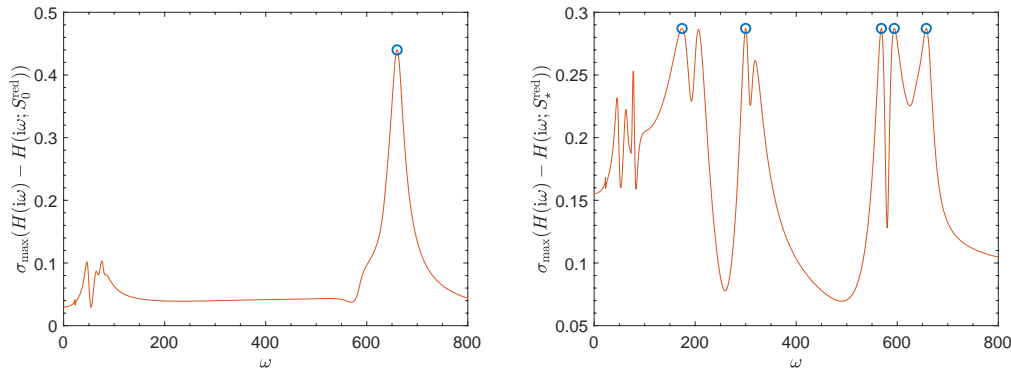


FIG. 8.4. The plots illustrate the errors of the initial, optimal models by Algorithm 3.1 for the “cd player model” with $r = 8$, and are analogous to those in Figure 8.2.

8.3. FOM Model. We next report numerical results on the FOM example available in the SLICOT library. The FOM example is a linear time-invariant system of order $n = 1006$, and with $m = p = 1$. The details are given in [27, Example 3]. Here, we are mainly interested in investigating the quality of the estimates for optimal reduced systems produced by Algorithm 3.1. To this end, we compare the errors of the reduced systems by Algorithm 3.1 with those of the balanced truncation, as well as the theoretical lower bounds for the errors in terms of Hankel singular values for varying choices of prescribed order r of the reduced system sought. As in §8.1 and §8.2, we set the initial estimate S_0^{red} for a minimizer as the system produced by the balanced truncation, and the initial reduced system S^0 is always of order 12 and interpolates the full system S at the imaginary parts of its most dominant three poles.

In Figure 8.5, the \mathcal{L}_∞ error $\|H - H(\cdot; S_\star^{\text{red}})\|_{\mathcal{L}_\infty}$ of the optimal reduced system S_\star^{red} by Algorithm 3.1 and the balanced truncation are plotted as functions of the prescribed order r of the reduced system sought. Included in the figure is also the plot of the Hankel singular value σ_{r+1} , a theoretical lower bound for the \mathcal{L}_∞ error $\|H - H(\cdot; S^{\text{red}})\|_{\mathcal{L}_\infty}$ of any system S^{red} of order r . Especially when $r \in [2, 6]$, the errors of the reduced systems by Algorithm 3.1 are quite close to the theoretical lower bound. Indeed, the errors of the reduced systems by Algorithm 3.1 usually differ by the theoretical lower bound by a factor of two at most. Moreover, in most of cases the errors of reduced systems by Algorithm 3.1 is significantly less than the error of the reduced system by the balanced truncation.

8.4. Systems with Large Order. Finally, we report results on systems with large order arising from modeling of power plants due to Rommes and his colleagues. All of these large-scale examples are available on the website of Rommes¹.

Due to the large order of the systems, the publicly available implementations of the balanced truncation are usually not applicable, and even when they are applicable, they require substantial amount of computation time. Hence, unlike the previous three subsections, we form the initial estimate for the minimizer S_0^{red} using the dominant poles of the system quite efficiently. For each system, we first compute the ten most dominant poles of the system using the approach in [22], in particular its implementation publicly available at <https://zenodo.org/record/5103430>. Then

¹<http://sites.google.com/site/rommes/software>

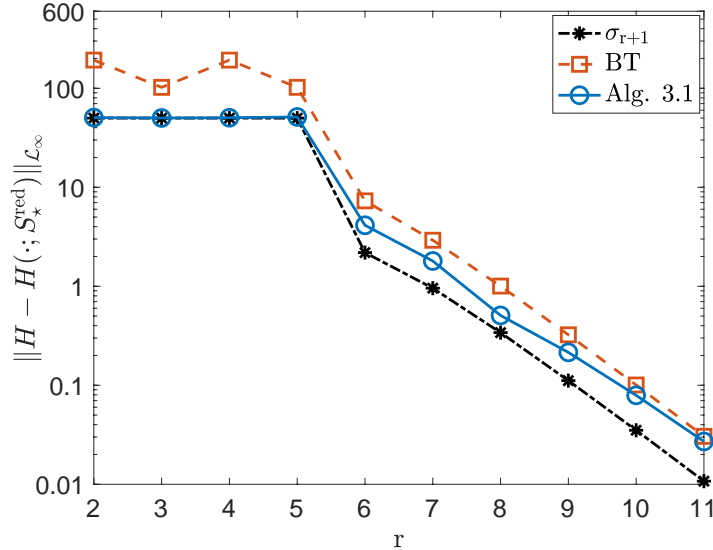


FIG. 8.5. Errors of the reduced systems of order $r \in [2, 11]$ produced by Algorithm 3.1 and the balanced truncation (BT), as well as the $(r+1)$ th largest Hankel singular value σ_{r+1} for the FOM example.

S_0^{red} of order r is constructed so as to interpolate the full system S at the imaginary parts of its $r/(4m)$ most dominant poles. Similarly, the initial reduced system S_0 is constructed such that it interpolates S at the imaginary parts of its ℓ most dominant poles, where $\ell = 7$ if the system is single-input-single-output (with $m = 1$), and $\ell = 3$ if the system is multiple-input-multiple-output (with $m > 1$). The order of the resulting reduced system S_0 is $4m\ell$. In all of the examples, the prescribed order r is such that $r < 4m\ell$, that is the order of S_0 is greater than the prescribed order r .

Even Algorithm 3.1 requires the computation of the \mathcal{L}_∞ norm of systems of order $n+r$ a few times (usually not more than 5-6 times in our experiments) in line 7, where n is the large order of the system. The classical level-set approaches [8, 9] for \mathcal{L}_∞ -norm computation and their implementations in Matlab are usually no more applicable, or when they are applicable, they take excessive amount of time. Instead, we employ the interpolatory subspace framework in [1] for these large-scale \mathcal{L}_∞ -norm computations, that is for maximizing $\sigma_{\max}(H(i\omega) - H(i\omega; S_r^{\text{red}}))$ over ω at the r th iteration. As the approach in [1] is locally convergent, whether the initial interpolation points are sufficiently close to global maximizers of $\sigma_{\max}(H(i\omega) - H(i\omega; S_r^{\text{red}}))$ plays a large role in converging to a global maximizer. We choose the initial interpolation points as the union of the imaginary parts of the ten most dominant poles, and 15 equally-spaced points on the interval $[-0.1, 2\mathcal{M}]$ with \mathcal{M} denoting the largest of the absolute values of the imaginary part of the ten most dominant poles.

The absolute and relative errors of the computed reduced systems of order r along with the total runtimes are reported in Table 8.4. For systems S20PI_n, S40PI_n, M40PI_n of order $n = 1182$ or $n = 2182$, we have also computed reduced systems of order r by means of the balanced truncation. In these examples, the errors of the reduced systems by Algorithm 3.1 are significantly smaller than those of the reduced systems by the balanced truncation. Moreover, Algorithm 3.1 on these examples

Example	$n, m = p$	r	approach	error	rel error	time
S20PI_n	1182, 1	12	Alg. 3.1	7.67×10^{-1}	2.23×10^{-1}	19.8
S20PI_n	1182, 1	16	Alg. 3.1	7.66×10^{-1}	2.22×10^{-1}	36.2
S20PI_n	1182, 1	12	BT	1.76×10^0	5.11×10^{-1}	45.1
S20PI_n	1182, 1	16	BT	1.32×10^0	3.84×10^{-1}	44.2
S40PI_n	2182, 1	12	Alg. 3.1	9.30×10^{-1}	2.78×10^{-1}	48.1
S40PI_n	2182, 1	16	Alg. 3.1	6.71×10^{-1}	2.00×10^{-1}	38.1
S40PI_n	2182, 1	32	BT	1.75×10^0	5.23×10^{-1}	410.5
M40PI_n	2182, 3	12	Alg. 3.1	1.99×10^0	5.22×10^{-1}	52.2
M40PI_n	2182, 3	24	Alg. 3.1	1.70×10^0	4.45×10^{-1}	117.1
M40PI_n	2182, 3	36	BT	3.07×10^0	8.03×10^{-1}	401.9
ww_vref_6405	13251, 1	12	Alg. 3.1	5.80×10^{-4}	2.04×10^{-1}	9.2
ww_vref_6405	13251, 1	16	Alg. 3.1	4.19×10^{-4}	1.48×10^{-1}	15.1
xingo_afonso	13250, 1	12	Alg. 3.1	3.55×10^{-2}	8.74×10^{-3}	14.4
xingo_afonso	13250, 1	16	Alg. 3.1	3.56×10^{-2}	8.77×10^{-3}	14.0
xingo_afonso	13250, 1	20	Alg. 3.1	1.13×10^{-2}	2.79×10^{-3}	26.2
bips07_1998	15066, 4	16	Alg. 3.1	1.24×10^1	6.30×10^{-2}	127.6
bips07_1998	15066, 4	32	Alg. 3.1	9.67×10^0	4.91×10^{-2}	219.8
bips07_3078	21228, 4	16	Alg. 3.1	1.27×10^1	6.06×10^{-2}	200.8
bips07_3078	21228, 4	32	Alg. 3.1	1.00×10^1	4.78×10^{-2}	274.1

TABLE 8.4

The absolute errors $\|H - H(\cdot; S_*^{\text{red}})\|_{\mathcal{L}_\infty}$ and relative errors $\|H - H(\cdot; S_*^{\text{red}})\|_{\mathcal{L}_\infty}/\|H\|_{\mathcal{L}_\infty}$ for systems of large order, where S_*^{red} is the optimal reduced system by either Algorithm 3.1 or the balanced truncation (BT). Total runtimes in seconds are also listed in the last column.

require less computation time compared to the balanced truncation. For systems with larger order, the implementation of the balanced truncation that we rely on does not seem suitable; as this implementation is based on dense linear algebra routines, it cannot cope with such systems. On the other hand, as evident from Table 8.4, Algorithm 3.1 is also able to deal with such systems of order ten thousands in a couple of minutes in the worst case. Most of the runtime of Algorithm 3.1 is usually taken by BFGS for solving reduced \mathcal{L}_∞ -norm minimization problems in line 5 involving small systems. In the end, rather than performing quite a few large-scale \mathcal{L}_∞ -norm computations, we end up performing quite a few small-scale \mathcal{L}_∞ -norm computations, and only a few large-scale \mathcal{L}_∞ -norm computations. This results in an approach that is not only computationally feasible but also more reliable, as small-scale \mathcal{L}_∞ norm computations can be fulfilled accurately, efficiently and reliably without worrying about local convergence thanks to the level-set methods [8, 9].

9. Software. A Matlab implementation of Algorithm 3.1 is publicly available at <https://zenodo.org/record/8344591>. The numerical results reported in the previous section are obtained with this implementation. Scripts are included to reproduce the results for the CD player model in §8.2, and the `xingo_afonso`, `bips07_1998` examples in §8.4. The results for other benchmark examples can be obtained similarly.

10. Conclusion. We have proposed an approach to find a locally optimal solution of the \mathcal{L}_∞ -norm model reduction problem. To our knowledge, this is the first work on the subject. Our approach is based on the usage of smooth optimization techniques such as the gradient descent method and BFGS. A direct application of such smooth optimization techniques for the \mathcal{L}_∞ -norm model reduction problem does not

seem suitable even for systems with modest order, as smooth optimization techniques converge very slowly and require the evaluation of the costly \mathcal{L}_∞ -norm objective too many times. Hence, our approach replaces the original system of modest or large order with a system of small order, and solves the resulting reduced \mathcal{L}_∞ -norm minimization problem by means of the smooth optimization. Then it refines and increases slightly the order of the reduced system based on the minimizer of this reduced optimization problem. This refinement is performed with an eye on interpolation between the full and reduced \mathcal{L}_∞ objectives. Under smoothness assumptions, admittedly strong in this context, we have given formal arguments for the quick convergence of the approach. We have also described how asymptotic stability constraints on the small system of prescribed order sought can be incorporated into the approach. The numerical experiments on a variety of real benchmark examples indicate that our approach retrieves indeed a locally optimal solution of the \mathcal{L}_∞ -norm model reduction problem in practice. Moreover, on some small benchmark examples, we have obtained reduced systems not far away from being optimal globally according to the theoretical lower bounds in terms of Hankel singular values. Experiments on large benchmark examples illustrate that the approach is usually suitable for systems of order a few ten thousands.

The quality of the converged locally optimal solution depends on the initial guess for the optimal reduced system. To generate the initial guess, we have employed two different strategies based on the balanced truncation and dominant poles. The first of these strategies may not be computationally feasible if the original system has large order, whereas the second strategy seems suitable even for large systems. However, a strategy generating a good initial guess is certainly worth further research. The proposed approach typically requires a few large-scale \mathcal{L}_∞ -norm computations. Performing these \mathcal{L}_∞ -norm computations accurately, especially without getting stagnated at a local maximizer that is not optimal globally, is crucial for the reliability of the proposed approach. We have employed the interpolatory subspace framework in [1] with the initial interpolation points chosen based on the dominant poles for these large-scale \mathcal{L}_∞ -norm computations. This approach usually seems to work well in practice for large-scale \mathcal{L}_∞ -norm computations. Still, we hope to explore further a good initial interpolation selection strategy for [1] so that it converges globally, leading to the correct \mathcal{L}_∞ norm with very high probability. Other efficient and accurate candidates for large-scale \mathcal{L}_∞ -norm computation are worth studying. In [13], the original system is replaced by a smaller order system obtained from the Loewner framework [20] to reduce the burden of large-scale \mathcal{L}_∞ -norm computations. We have not attempted here to incorporate the Loewner framework into our approach. As a future work, our approach can possibly benefit from the Loewner framework; for instance, the initial reduced system replacing the full system can perhaps be obtained using the Loewner framework. Our quick convergence result for the proposed approach is under strong smoothness assumptions. Investigating the order of convergence of the approach in the likely nonsmooth setting (i.e., when the \mathcal{L}_∞ objective at the converged minimizer is nonsmooth) is a possible direction for future research. Last but not the least, the convergence of smooth optimization techniques such as BFGS is more of an empirical phenomenon in the current-state-of-art with some intuition as to why. Analyzing the convergence of smooth optimization techniques in the presence of nonsmoothness at the optimizers is an important open problem.

REFERENCES

- [1] A. ALIYEV, P. BENNER, E. MENGI, P. SCHWERDTNER, AND M. VOIGT, *Large-scale compu-*

- tation of \mathcal{L}_∞ -norms by a greedy subspace method, *SIAM J. Matrix Anal. Appl.*, 38 (2017), pp. 1496–1516.
- [2] A. ALIYEV, P. BENNER, E. MENGI, AND M. VOIGT, *A subspace framework for H -infinity norm minimization*, *SIAM J. Matrix Anal. Appl.*, 41 (2020), pp. 928–956.
 - [3] A. C. ANTOULAS, *Approximation of Large-Scale Dynamical Systems*, vol. 6 of *Adv. Des. Control*, SIAM Publications, Philadelphia, PA, 2005.
 - [4] A. ASL AND M. L. OVERTON, *Analysis of the gradient method with an armijo–wolfe line search on a class of non-smooth convex functions*, *Optim. Method Softw.*, 35 (2020), pp. 223–242.
 - [5] C. BEATTIE AND S. GUGERCIN, *Interpolatory projection methods for structure-preserving model reduction*, *Systems Control Lett.*, 58 (2009), pp. 225–232.
 - [6] P. BENNER, E. QUINTANA-ORTI, AND G. QUINTANA-ORTI, *Computing optimal hankel norm approximations of large-scale systems*, in *Proceedings of 43rd IEEE Conference on Decision and Control*, vol. 3, 2004, pp. 3078–3083.
 - [7] P. BENNER AND S. W. R. WERNER, *MORLAB – Model Order Reduction LABORatory (version 5.0)*, Aug. 2019, <https://doi.org/10.5281/zenodo.3332716>. see also: <http://www.mpi-magdeburg.mpg.de/projects/morlab>.
 - [8] S. BOYD AND V. BALAKRISHNAN, *A regularity result for the singular values of a transfer matrix and a quadratically convergent algorithm for computing its L_∞ -norm*, *Systems Control Lett.*, 15 (1990), pp. 1–7.
 - [9] N. A. BRUINSMA AND M. STEINBUCH, *A fast algorithm to compute the H_∞ -norm of a transfer function matrix*, *Systems Control Lett.*, 14 (1990), pp. 287–293.
 - [10] Y. CHAHLAOUI AND P. VAN DOOREN, *Benchmark examples for model reduction of linear time-invariant dynamical systems*, in *Dimension Reduction of Large-Scale Systems*, vol. 45, 2005, pp. 381–395.
 - [11] 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 (2017), pp. 148–181.
 - [12] G. E. DULLERUD AND F. PAGANINI, *A Course in Robust Control: A Convex Approach*, Springer-Verlag, New York, NY, 2nd ed., 2010.
 - [13] G. FLAGG, C. A. BEATTIE, AND S. GUGERCIN, *Interpolatory H_∞ model reduction*, *Syst. Control Lett.*, 62 (2013), pp. 567–574.
 - [14] K. GLOVER, *All optimal hankel-norm approximations of linear multivariable systems and their \mathcal{L}_∞ -error bounds*, *Int. J. Control*, 39 (1984), pp. 1115–1193.
 - [15] S. GUGERCIN AND A. C. ANTOULAS, *A survey of model reduction by balanced truncation and some new results*, *Int. J. Control*, 77 (2004), pp. 748–766.
 - [16] S. GUGERCIN, A. C. ANTOULAS, AND C. BEATTIE, *\mathcal{H}_2 model reduction for large-scale linear dynamical systems*, *SIAM J. Matrix Anal. Appl.*, 30 (2008), pp. 609–638.
 - [17] S. GUGERCIN, D. C. SORENSEN, AND A. C. ANTOULAS, *A modified low-rank Smith method for large-scale Lyapunov equations*, *Numerical Algorithms*, 32 (2003), pp. 27–55.
 - [18] P. LANCASTER, *On eigenvalues of matrices dependent on a parameter*, *Numer. Math.*, 6 (1964), pp. 377–387.
 - [19] A. LEWIS AND M. L. OVERTON, *Nonsmooth optimization via quasi-Newton methods*, *Math. Program.*, 141 (2013), pp. 135–163.
 - [20] A. J. MAYO AND A. C. ANTOULAS, *A framework for the solution of the generalized realization problem*, *Linear Algebra Appl.*, 427 (2007), pp. 634–662.
 - [21] E. MENGI, *Large-scale and global maximization of the distance to instability.*, *SIAM J. Matrix Anal. Appl.*, 39 (2018), pp. 1776–1809.
 - [22] E. MENGI, *Large-scale estimation of dominant poles of a transfer function by an interpolatory framework*, *SIAM J. Sci. Comput.*, 44 (2022), pp. A2412–A2438.
 - [23] E. MENGI, E. A. YILDIRIM, AND M. KILIÇ, *Numerical optimization of eigenvalues of Hermitian matrix functions*, *SIAM J. Matrix Anal. Appl.*, 35 (2014), pp. 699–724, <https://doi.org/10.1137/130933472>, <http://dx.doi.org/10.1137/130933472>.
 - [24] B. MOORE, *Principal component analysis in linear systems: controllability, observability, and model reduction*, *IEEE Trans. Autom. Control*, 26 (1981), pp. 17–32.
 - [25] C. MULLIS AND R. ROBERTS, *Synthesis of minimum roundoff noise fixed point digital filters*, *IEEE Trans. Circuits Syst.*, 23 (1976), pp. 551–562.
 - [26] T. PENZL, *A cyclic low-rank Smith method for large sparse Lyapunov equations*, *SIAM J. Sci. Comput.*, 21 (1999), pp. 1401–1418.
 - [27] T. PENZL, *Algorithms for model reduction of large dynamical systems*, *Linear Algebra Appl.*, 415 (2006), pp. 322–343.
 - [28] T. STYKEL, *Gramian-based model reduction for descriptor systems*, *Math. Contol Signals Syst.*, 16 (2004), pp. 297–319.