Computation of the $\mathcal{L}_\infty$-Norm Using Rational Interpolation ⋆

P. Schwerdtner * **M. Voigt** *

* *Technische Universität Berlin, Institut für Mathematik, Straße des 17. Juni 136, 10623 Berlin, Germany*

Abstract: We propose a greedy interpolation approach to compute the $\mathcal{L}_\infty$-norm of a possibly irrational $\mathcal{L}_\infty$-function. We approximate this function by a sequence of rational $\mathcal{L}_\infty$-functions. For this we use interpolation employing the Loewner matrix framework. In each iteration, the $\mathcal{L}_\infty$-norm of the rational approximation is computed using established methods. Then a new interpolation point is added where the $\mathcal{L}_\infty$-norm of the function approximation is attained. This way, the $\mathcal{L}_\infty$-norm of the approximation converges superlinearly to the $\mathcal{L}_\infty$-norm of the original function. We illustrate the efficiency of the resulting algorithm for various numerical examples and compare it to state-of-the-art methods.

Keywords: Computer-aided control systems design, descriptor systems, eigenvalue problems, $\mathcal{H}_\infty$ control, linear control systems, norms, numerical algorithms, robust control

1. INTRODUCTION

We address the computation of the $\mathcal{L}_\infty$-norm of a matrix-valued function

$$H : \Omega \to \mathbb{C}^{p\times m} \tag{1}$$

where the domain Ω is assumed to be an open subset of the complex plane enclosing the imaginary axis $\mathrm{i}\mathbb{R}$. Such functions typically appear in the frequency domain analysis of linear dynamical systems

$$\begin{aligned} \tfrac{\mathrm{d}}{\mathrm{d}t}Ex(t) &= Ax(t) + Bu(t),\\ y(t) &= Cx(t), \end{aligned} \tag{2}$$

where $x : \mathbb{R} \to \mathbb{R}^n$ is the state, $u : \mathbb{R} \to \mathbb{R}^m$ is the control input, $y : \mathbb{R} \to \mathbb{R}^p$ is the measured output, and $E, A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$. If $sE - A$ is a regular pencil, that is $\det(sE - A) \not\equiv 0$, then the transfer function of (2) is given by

$$H(s) = C(sE - A)^{-1}B.$$

Note that in this case, H is a rational function. On the other hand, the transfer function of the delay differential-algebraic system

$$\begin{aligned} \tfrac{\mathrm{d}}{\mathrm{d}t}Ex(t) &= A_0x(t) + \sum_{j=1}^{q} A_jx(t-\tau_j) + Bu(t),\\ y(t) &= Cx(t), \end{aligned} \tag{3}$$

with E, B, C having the same dimensions as in (2) and $A_j \in \mathbb{R}^{n\times n}$, $j = 0, 1, \ldots, q$, is a irrational function given by

$$H(s) = C\left(sE - A_0 - \sum_{j=1}^{q} \mathrm{e}^{-s\tau_j}A_j\right)^{-1}B. \tag{4}$$

⋆ This work was supported by the German Research Foundation (DFG) within the priority program 1897: "Calm, Smooth, Smart – Novel Approaches for Influencing Vibrations by Means of Deliberately Introduced Dissipation".

Both of these transfer functions are encompassed by the setting given in (1), however our framework is not limited to these cases.

We define the spaces

$$\mathcal{L}_\infty^{p\times m} := \left\{ H|_{\mathrm{i}\mathbb{R}} \;\middle|\; H : \Omega \to \mathbb{C}^{p\times m} \text{ is analytic for an open } \Omega \subseteq \mathbb{C} \text{ with } \mathrm{i}\mathbb{R} \subset \Omega \text{ and } \sup_{\omega\in\mathbb{R}} \|H(\mathrm{i}\omega)\|_2 < \infty \right\},$$

$$\mathcal{H}_\infty^{p\times m} := \left\{ H : \mathbb{C}^+ \to \mathbb{C}^{p\times m} \;\middle|\; H \text{ is analytic and } \sup_{s\in\mathbb{C}^+} \|H(s)\|_2 < \infty \right\},$$

where $\mathbb{C}^+ := \{s \in \mathbb{C} \mid \operatorname{Re}(s) > 0\}$. The function H (more precisely, its restriction to the imaginary axis) is assumed to be in the space $\mathcal{L}_\infty^{p\times m}$. For ease of notation, we write $H \in \mathcal{L}_\infty^{p\times m}$ instead of $H|_{\mathrm{i}\mathbb{R}} \in \mathcal{L}_\infty^{p\times m}$. For such, the $\mathcal{L}_\infty$-norm is given by

$$\|H\|_{\mathcal{L}_\infty} := \sup_{\omega\in\mathbb{R}} \|H(\mathrm{i}\omega)\|_2 = \sup_{\omega\in\mathbb{R}} \sigma_{\max}(H(\mathrm{i}\omega)), \tag{5}$$

where $\sigma_{\max}(\cdot)$ denotes the largest singular value of its matrix argument. Since in most applications, functions in $\mathcal{H}_\infty^{p\times m}$ are rather investigated, it is important to note that for such functions, the $\mathcal{H}_\infty$-norm and the $\mathcal{L}_\infty$-norm are equal, i. e.,

$$\|H\|_{\mathcal{H}_\infty} := \sup_{s\in\mathbb{C}^+} \|H(s)\|_2 = \sup_{\omega\in\mathbb{R}} \sigma_{\max}(H(\mathrm{i}\omega)) = \|H\|_{\mathcal{L}_\infty}.$$

1.1 Literature

Established algorithms for computing the $\mathcal{L}_\infty$-norm of rational functions are usually based on the state-space realizations determined by the matrices E, A, B, C in (2). Usually, these are only suitable for small and dense problems, where n is a few thousand at most. The algorithms are outlined in Boyd and Balakrishnan (1990)

and Bruinsma and Steinbuch (1990) for $E = I_n$ and in Benner et al. (2012) for general E. These methods use level-set based optimization approaches which require the repeated solution of structured eigenvalue problems. An extension to the $\mathcal{L}_\infty$-norm computation for delay-systems is covered by Gumussoy and Michiels (2015, 2009). However, as for the rational case, these algorithms are also limited to systems of small order. Furthermore, eigenvalue optimization techniques as in Mengi et al. (2014) can be used to compute the $\mathcal{L}_\infty$-norm.

The problem of computing the $\mathcal{H}_\infty$-norm of rational functions, where the matrices E and A in its state-space realization are of large dimension has only been considered recently. For example, there are a couple of works exploiting the relationship between the $\mathcal{H}_\infty$-norm and spectral value sets (Guglielmi et al. (2013); Benner and Voigt (2014); Mitchell and Overton (2016)). Another algorithm using the implicit determinant method and with the potential of being applied to large-scale systems is discussed in Freitag et al. (2014).

In Aliyev et al. (2017), the problem of computing the $\mathcal{L}_\infty$-norm for (possibly irrational) functions H of the form

$$H(s) := C(s)D(s)^{-1}B(s) \tag{6}$$

is considered with meromorphic functions $B : \Omega \to \mathbb{C}^{n\times m}$, $C : \Omega \to \mathbb{C}^{p\times n}$, $D : \Omega \to \mathbb{C}^{n\times n}$ that satisfy

$$\begin{aligned} B(s) &:= f_1(s)B_1 + \ldots + f_{\kappa_B}(s)B_{\kappa_B}, \\ C(s) &:= g_1(s)C_1 + \ldots + g_{\kappa_C}(s)C_{\kappa_C}, \\ D(s) &:= h_1(s)D_1 + \ldots + h_{\kappa_D}(s)D_{\kappa_D}. \end{aligned}$$

Here, $B_1, \ldots, B_{\kappa_B} \in \mathbb{C}^{n\times m}$, $C_1, \ldots, C_{\kappa_C} \in \mathbb{C}^{p\times n}$, $D_1, \ldots, D_{\kappa_D} \in \mathbb{C}^{n\times n}$ are given matrices and $f_1, \ldots, f_{\kappa_B}$, $g_1, \ldots, g_{\kappa_C}$, $h_1, \ldots, h_{\kappa_D} : \Omega \to \mathbb{C}$ are functions meromorphic in Ω. Therein, the typical setting in applications where n is large and further $n \gg m, p$ is exploited to compute the $\mathcal{L}_\infty$-norm at a lower cost using subspace projection. More precisely, matrices $V, W \in \mathbb{C}^{n\times k}$ with $\widetilde{B}(s) := W^{\mathsf{H}}B(s)$, $\widetilde{C}(s) := C(s)V$, and $\widetilde{D}(s) := W^{\mathsf{H}}D(s)V$ and $n \gg k$ are computed. Then, the $\mathcal{L}_\infty$-norm of the reduced function $\widetilde{H}(s) := \widetilde{C}(s)\widetilde{D}(s)^{-1}\widetilde{B}(s)$ is computed and the projection subspaces are expanded using the information of $\widetilde{H}(\mathrm{i}\omega_*)$ where $\mathrm{i}\omega_*$ is the point on the imaginary axis, where the $\mathcal{L}_\infty$-norm value of the reduced function $\widetilde{H}$ is attained. This leads to an iterative algorithm which is shown to have a local superlinear rate of convergence. This approach works well for descriptor systems, i. e., $D(s) = sE - A$, since it can rely on well-established methods to compute the $\mathcal{L}_\infty$-norm of the reduced functions. However, in the irrational case, it makes use of the `eigopt`-algorithm (Mengi et al. (2014)) for the maximization of $\sigma_{\max}(H(\mathrm{i}\cdot))$. This algorithm requires a priori knowledge about the function H such as the frequency range in which the global maximizer of $\sigma_{\max}(H(\mathrm{i}\cdot))$ is located as well as a global lower bound on the second derivative of $-\sigma_{\max}(H(\mathrm{i}\cdot))$. This a priori knowledge is often difficult to obtain and a large frequency range and a low bound on the second derivative of $-\sigma_{\max}(H(\mathrm{i}\cdot))$ can lead to high computational costs.

1.2 Contributions and Outline

The motivation of our article is to replace the need for obtaining the above mentioned a priori knowledge of the function under consideration. The idea of our method is similar to what was done in Aliyev et al. (2017) - we iteratively compute the $\mathcal{L}_\infty$-norm of a reduced function and thereafter expand this function in a way such that the $\mathcal{L}_\infty$-norm of the reduced functions converges to the $\mathcal{L}_\infty$-norm of the original one. However, in contrast to Aliyev et al. (2017), the reduced functions are not created via projection of the matrix-valued functions $B(\cdot)$, $C(\cdot)$, and $D(\cdot)$. Instead we employ the Loewner matrix framework from Mayo and Antoulas (2007) to construct a sequence of rational approximations of H whose $\mathcal{L}_\infty$-norm converges to the $\mathcal{L}_\infty$-norm of H. These rational approximations can be realized by linear state-space systems, and therefore, well-established methods for computing the $\mathcal{L}_\infty$-norm can be used. This avoids the use of `eigopt` (and thus the need for some hard-to-obtain a priori knowledge of the function) and this can also highly accelerate the computations in the irrational case.

In fact, using this approach the structural constraints that were imposed on H in Aliyev et al. (2017) are not necessary, since only evaluations of H are needed to construct the rational approximations. This work is inspired by Beattie and Gugercin (2012) in which an iterative algorithm for rational approximation that only uses transfer function evaluations without the knowledge of a realization was devised. In the next section, we show how to construct rational approximations using the Loewner framework. Thereafter, we summarize our approach for computing the $\mathcal{L}_\infty$-norm in Algorithm 1. The local superlinear convergence property can be transferred from Aliyev et al. (2017). Implementation subtleties of the proposed method and conducted numerical experiments are discussed in Section 4.

2. PRELIMINARIES

In Aliyev et al. (2017) the function H is interpolated at a set of interpolation points $\mathbb{W} = \{\mathrm{i}\omega_1, \ldots, \mathrm{i}\omega_r\} \subset \mathrm{i}\mathbb{R}$. Under the assumption that $p = m$ (see Aliyev et al. (2017) for the general case), the two projection spaces

$$\begin{aligned} \mathcal{V} &= \operatorname{span}\left[D(\mathrm{i}\omega_1)^{-1}B(\mathrm{i}\omega_1) \;\ldots\; D(\mathrm{i}\omega_r)^{-1}B(\mathrm{i}\omega_r)\right], \\ \mathcal{W} &= \operatorname{span}\left[D(\mathrm{i}\omega_1)^{-\mathsf{H}}C(\mathrm{i}\omega_1)^{\mathsf{H}} \;\ldots\; D(\mathrm{i}\omega_r)^{-\mathsf{H}}C(\mathrm{i}\omega_r)^{\mathsf{H}}\right] \end{aligned}$$

are constructed. With $V, W \in \mathbb{C}^{n\times k}$ being matrices whose columns form an orthonormal basis of $\mathcal{V}$ and $\mathcal{W}$, respectively, the reduced function is given by $\widetilde{H}(s) := \widetilde{C}(s)\widetilde{D}(s)^{-1}\widetilde{B}(s)$ with $\widetilde{B}(s) := W^{\mathsf{H}}B(s)$, $\widetilde{C}(s) := C(s)V$, and $\widetilde{D}(s) := W^{\mathsf{H}}D(s)V$. This construction ensures the Hermite interpolation properties

$$H(\mathrm{i}\omega_j) = \widetilde{H}(\mathrm{i}\omega_j), \quad H'(\mathrm{i}\omega_j) = \widetilde{H}'(\mathrm{i}\omega_j) \quad \text{for } j = 1, \ldots, r. \tag{7}$$

Moreover, if the maximum singular values of $H(\mathrm{i}\omega_j)$ for $j = 1, \ldots, r$ are simple, then $\sigma_{\max}(H(\mathrm{i}\cdot))$ and $\sigma_{\max}\big(\widetilde{H}(\mathrm{i}\cdot)\big)$ are differentiable, and this gives the Hermite interpolation properties

$$\begin{aligned} &\sigma_{\max}(H(\mathrm{i}\omega_j)) = \sigma_{\max}\big(\widetilde{H}(\mathrm{i}\omega_j)\big), \\ &\sigma'_{\max}(H(\mathrm{i}\omega_j)) = \sigma'_{\max}\big(\widetilde{H}(\mathrm{i}\omega_j)\big) \quad \text{for } j = 1, \ldots, r. \end{aligned} \tag{8}$$

Now we want to construct a *realization of a rational function* that matches the function H and its derivative at the interpolation points given in $\mathbb{W} \subset \mathrm{i}\mathbb{R}$, i. e., it fulfills

the Hermite interpolation property (7). So we want to construct a linear system

$$\begin{aligned}\tfrac{\mathrm{d}}{\mathrm{d}t}\widetilde{E}x(t) &= \widetilde{A}x(t) + \widetilde{B}u(t),\\ y(t) &= \widetilde{C}x(t),\end{aligned}$$

which, for brevity, we denote by $\left[\widetilde{E}, \widetilde{A}, \widetilde{B}, \widetilde{C}\right]$ such that its transfer function $\widetilde{H}(s) := \widetilde{C}(s\widetilde{E} - \widetilde{A})^{-1}\widetilde{B}$ is a Hermite interpolant of H. In Mayo and Antoulas (2007) a framework for the solution of this problem is given by incorporating so-called Loewner matrices. Usually, Loewner matrices are employed for tangential interpolation. By choosing the tangential directions as the unit vectors, we obtain the *block Loewner matrix* $\mathbf{E} = [\mathbf{E}_{ij}] \in \mathbb{C}^{pr\times mr}$ and the *shifted block Loewner matrix* $\mathbf{A} = [\mathbf{A}_{ij}] \in \mathbb{C}^{pr\times mr}$ with

$$\mathbf{E}_{ij} = \begin{cases} \dfrac{1}{\mathrm{i}\omega_i - \mathrm{i}\omega_j}(H(\mathrm{i}\omega_i) - H(\mathrm{i}\omega_j)), & \text{if } i \neq j,\\ H'(\mathrm{i}\omega_i), & \text{else,}\end{cases}$$

$$\mathbf{A}_{ij} = \begin{cases} \dfrac{1}{\omega_i - \omega_j}(\omega_i H(\mathrm{i}\omega_i) - \omega_j H(\mathrm{i}\omega_j)), & \text{if } i \neq j,\\ H(\mathrm{i}\omega_i) + \mathrm{i}\omega_i H'(\mathrm{i}\omega_i), & \text{else,}\end{cases} \tag{9}$$

and moreover,

$$\mathbf{B} = \begin{bmatrix} H(\mathrm{i}\omega_1)\\ \vdots \\ H(\mathrm{i}\omega_r)\end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} H(\mathrm{i}\omega_1) & \dots & H(\mathrm{i}\omega_r)\end{bmatrix}.$$

Note that the matrices $\mathbf{E}$ and $\mathbf{A}$ are not square in general. To obtain a regular linear system, a regularization procedure may be necessary. To this end, if for all $\mu \in \{\mathrm{i}\omega_1, \dots, \mathrm{i}\omega_r\}$ we have that

$$\operatorname{rank}(\mu\mathbf{E} - \mathbf{A}) = \operatorname{rank}\begin{bmatrix}\mathbf{E} & \mathbf{A}\end{bmatrix} = \operatorname{rank}\begin{bmatrix}\mathbf{E}\\ \mathbf{A}\end{bmatrix}, \tag{10}$$

then the linear system $\left[\widetilde{E}, \widetilde{A}, \widetilde{B}, \widetilde{C}\right]$ is obtained after applying economic singular value decompositions to vertical and horizontal concatenations of $\mathbf{E}$ and $\mathbf{A}$. This means that we compute

$$\begin{bmatrix}\mathbf{E} & \mathbf{A}\end{bmatrix} = Y\Sigma_{\mathrm{l}}\widetilde{X}^{\mathsf{H}}, \quad \begin{bmatrix}\mathbf{E}\\ \mathbf{A}\end{bmatrix} = \widetilde{Y}\Sigma_{\mathrm{r}}X^{\mathsf{H}} \tag{11}$$

where we have $\Sigma_{\mathrm{l}} = \operatorname{diag}(\sigma_{\mathrm{l},1}, \dots, \sigma_{\mathrm{l},k}) \in \mathbb{R}^{k\times k}$, $\Sigma_{\mathrm{r}} = \operatorname{diag}(\sigma_{\mathrm{r},1}, \dots, \sigma_{\mathrm{r},k}) \in \mathbb{R}^{k\times k}$ with $\sigma_{\mathrm{l},1} \geq \dots \geq \sigma_{\mathrm{l},k} > 0$, $\sigma_{\mathrm{r},1} \geq \dots \geq \sigma_{\mathrm{r},k} > 0$, and $\widetilde{Y}$, Y, $\widetilde{X}$, $X \in \mathbb{C}^{n\times k}$ have orthonormal columns. The system matrices are then given by

$$\widetilde{E} = -Y^{\mathsf{H}}\mathbf{E}X, \quad \widetilde{A} = -Y^{\mathsf{H}}\mathbf{A}X, \quad \widetilde{B} = Y^{\mathsf{H}}\mathbf{B}, \quad \widetilde{C} = \mathbf{C}X. \tag{12}$$

Remark 1. Condition (10) can be related to the Kronecker structure of the pencil $s\mathbf{E} - \mathbf{A}$, see (Gantmacher, 1959, Chap. VII, §3). Namely, if (10) is fulfilled, then the pencil $s\mathbf{E}-\mathbf{A}$ is either regular or its singular Kronecker blocks are of size at most 1×0 or 0×1. The treatment of higher-index singular Kronecker blocks is more involved.

Remark 2. The matrices $\mathbf{E}$, $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ are generally complex. However, if the set of interpolation points that are used for their construction is closed under complex conjugation, then these matrices can be chosen to be real by an appropriate transformation, see Antoulas et al. (2017). For this purpose, in our implementation we generate additional interpolation data for the complex conjugates of $\mathbb{W}$ such that the set of interpolation points $\mathbb{W}\cup\overline{\mathbb{W}}$ is closed under complex conjugation. This step is necessary in order to use the implementation of the Boyd-Balakrishnan algorithm from Benner et al. (2012) which is only suitable for real data.

Algorithm 1 Computation of the $\mathcal{L}_\infty$-norm using rational approximation

Input: A function $H \in \mathcal{L}_\infty^{p\times m}$, initial interpolation points $\mathbb{W} \subset \mathrm{i}\mathbb{R}$.
Output: The $\mathcal{L}_\infty$-norm of H and the maximizer ω_* with $f_* := \|H\|_{\mathcal{L}_\infty} = \|H(\mathrm{i}\omega_*)\|_2$.

```
while not converged do
    Construct [Ẽ, Ã, B̃, C̃] as in (9), (11), and (12).
    Define H̃(s) := C̃(sẼ − Ã)^{-1}B̃.
    Compute ω_* := arg max_{ω∈ℝ∪{∞}} σ_max(H̃(iω)).
    Set 𝕎 := 𝕎 ∪ {iω_*}.
end while
Set f_* := ‖H(iω_*)‖_2.
```

3. OUR METHOD

Being provided the tools to construct a low-order system $\left[\widetilde{E}, \widetilde{A}, \widetilde{B}, \widetilde{C}\right]$ such that its transfer function interpolates a given function H at the interpolation points $\mathbb{W}$, an algorithm for computing the $\mathcal{L}_\infty$-norm is derived. Algorithm 1 is an iterative interpolation method where the set $\mathbb{W}$ which contains the interpolation points is altered after each iteration. In each iteration, the maximizer $\arg\max_{\omega\in\mathbb{R}\cup\{\infty\}} \sigma_{\max}\big(\widetilde{H}(\mathrm{i}\omega)\big)$ with $\widetilde{H}$ being the transfer function of the current rational approximation is added to $\mathbb{W}$, and then a new rational approximation is computed based on the updated interpolation points. The complete method is summarized in Algorithm 1.

The most expensive part is the evaluation of the given function H that requires the solution of possibly large linear systems with many right-hand sides. Thereafter, only rather small matrices must be constructed and the $\mathcal{L}_\infty$-norm of the rational approximations can be computed efficiently by using well-established methods such as Benner et al. (2012). Note that due to the Hermite interpolation properties (7) and (8), the same convergence theory as in Aliyev et al. (2017) applies. This means that we have a local superlinear rate of convergence.

4. IMPLEMENTATION DETAILS AND NUMERICAL RESULTS

In this section we describe a few details on our implementation of Algorithm 1, which is available online [1] . As already explained in Section 2, by construction, the pencil $s\mathbf{E} - \mathbf{A}$ in (9) may be singular or close to singular. In particular, this is the case, if $m \neq p$, or if during the iteration of Algorithm 1, two interpolation points get very close. A regularization of the matrix pencil can be performed using the singular value decomposition that is shown in equation (11). Let $k_{\max}$ be the largest index k such that

$$\frac{\sigma_{\mathrm{l},k}}{\sigma_{\mathrm{l},1}} > \varepsilon_{\mathrm{reg}} \quad \text{and} \quad \frac{\sigma_{\mathrm{r},k}}{\sigma_{\mathrm{r},1}} > \varepsilon_{\mathrm{reg}},$$

[1] implemented in `linorm_subsp v1.2` and downloadable from `http://www.math.tu-berlin.de/index.php?id=186267&L=1`.

where ε_{reg} is a predefined truncation tolerance. Then the matrices X and Y in (11) are truncated to size $n \times k_{\max}$ and the projection is applied as in (12). In our numerical experiments, the default value for ε_{reg} was set to 10^{-12}.

Algorithm 1 is terminated if one of the following conditions is satisfied:

a) The relative distance of the maximizers ω_r, ω_{r-1} of $\sigma_{\max}\big(\widetilde{H}(\mathrm{i}\cdot)\big)$ between two consecutive iterations is less or equal than a prescribed tolerance ε_ω, i. e.,

$$|\omega_r - \omega_{r-1}| \leq \frac{1}{2}\varepsilon_\omega \, |\omega_r + \omega_{r-1}| \, .$$

b) The relative difference of two consecutively computed $\mathcal{L}_\infty$-norms f_r, f_{r-1} of the reduced functions is less or equal than a prescribed tolerance ε_f provided that ω_r and ω_{r-1} are sufficiently close. Formally, this is the case when

$$|f_r - f_{r-1}| \leq \frac{1}{2}\varepsilon_f \, |f_r + f_{r-1}| \quad \text{and}$$
$$|\omega_r - \omega_{r-1}| \leq \delta \cdot \frac{1}{2}\varepsilon_\omega \, |\omega_r + \omega_{r-1}| \, .$$

c) The maximum number of iterations is exceeded, i. e.,

$$r > r_{\max}.$$

Criterion b) is needed for the case that the function $\sigma_{\max}\big(H(\mathrm{i}\cdot)\big)$ is "flat" near its global maximizer. In this case, a large interval of points ω gives almost the same value of $\sigma_{\max}\big(H(\mathrm{i}\cdot)\big)$. Due to numerical errors, the maximizers for the final iterations may still violate the convergence condition a), while the actual norm values have already converged. For our numerical experiments we choose the default values $\varepsilon_\omega = \varepsilon_f = 10^{-6}$, $\delta = 10^3$, and $r_{\max} = 30$.

Now we present some numerical results of our method. Our experiments have been performed on a machine with an 4 Intel® Core™ 3.30GHz i5-4590 CPUs and 16GB RAM in MATLAB 9.4.0.813654 (R2018a) running on Linux version 4.4.104-39-default. We test our algorithm on 24 irrational functions. The first example is the transfer function of a delay model from Beattie and Gugercin (2009), examples 2–12 are constructed from the nonlinear eigenvalue problems in Betcke et al. (2013), and examples 13–24 are taken from Michiels and Gummussoy (2010). On the website of our code we also provide the data files for all of these examples.

To illustrate the behavior of our method, we depict the maximum singular value functions for the original function H and its rational approximations in each iteration in Figure 1. Even if the rational functions do not approximate the system well on the entire imaginary axis, the approximation quality is very good locally close to the maximizer of $\sigma_{\max}(H(\mathrm{i}\cdot))$.

In Table 1 we compare Algorithm 1 with the method from Mengi et al. (2014) implemented in `eigopt`. Here we use an absolute accuracy of $10^{-6} \cdot \|H\|_{\mathcal{L}_\infty}$ as `eigopt`'s termination criterion, where $\|H\|_{\mathcal{L}_\infty}$ is computed by our method. With this we achieve a small relative distance of the norm values of 10^{-5} or better. Also the locations of the maximizers match up to numerical errors – in fact, the correct values are found for each example. Moreover, the new method is faster for many examples with a massive acceleration for some of the examples.

To demonstrate that our method also works efficiently on transfer functions of large-scale systems, we show a comparison of Algorithm 1 with the method of Aliyev et al. (2017) on the transfer function of delay system from Beattie and Gugercin (2009) which has variable state-space dimension. The results are in Table 2. Note that for large values of n, the evaluation of the transfer function becomes the most expensive part of both methods. However, in our new method also the derivatives of the transfer function have to be evaluated to construct the block Loewner matrices. In this case, the gain of not using `eigopt` is lost by the additional evaluation of the derivatives.

Table 3 contains a comparison of the runtime of Algorithm 1 with the method described in Michiels and Gummussoy (2010) and implemented in the software package `hinfn`. We use `hinfn` with default options except that we set the relative tolerance for the $\mathcal{L}_\infty$-norm to 10^{-6} and we choose the initial interpolation points of Algorithm 1 as test frequencies for `hinfn` to obtain a fair comparison. The investigated functions are transfer functions of delay systems as in (3). The state-space dimensions of examples 1–7, 11, and 12 do not exceed 4. Note, that for these examples, the runtime ratios are not high, since the involved delay eigenvalue problems can still be solved efficiently by `hinfn`. However, Algorithm 1 is still more efficient for these small-scale examples. A larger performance gain can be observed in examples 8–10. Here, the state-space dimension is larger (up to 40). Then the solution of the delay eigenvalue problems becomes the bottleneck in `hinfn`. On the other hand, the state-space dimension of the reduced systems in the Loewner framework only depends on the number of interpolation points and not on the state-space dimension of the original system. Thus, as expected, Algorithm 1 drastically outperforms `hinfn` for higher-dimensional examples. Note that `hinfn` fails for the last example, but if we enlarge the number of discretization points for the delay eigenvalue problem to 20, we obtain the correct result in 0.194s.

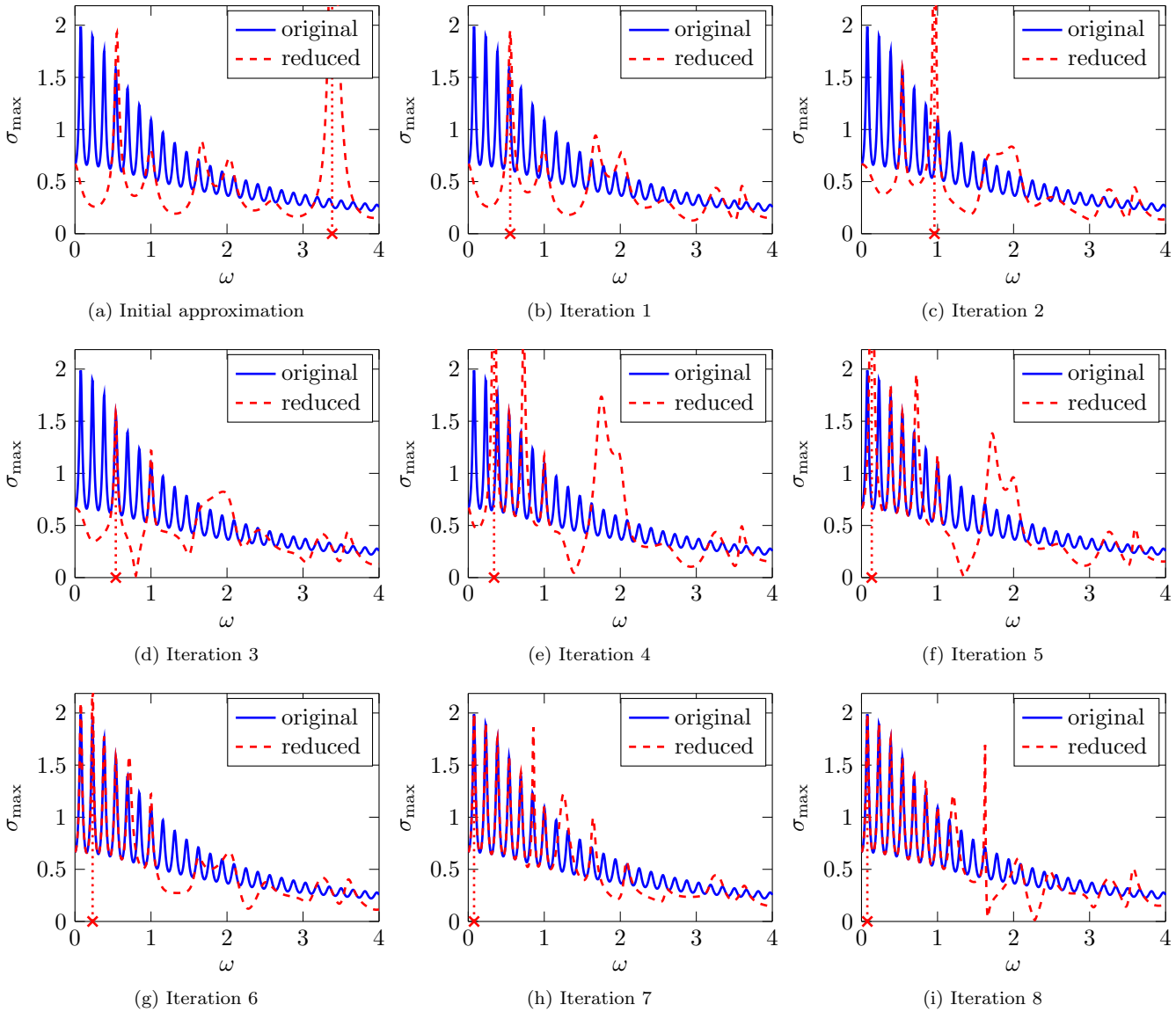
5. CONCLUSIONS AND OUTLOOK

We have introduced an approach for the computation of the $\mathcal{L}_\infty$-norm of an $\mathcal{L}_\infty$-function H by using the Loewner framework. This approach only uses function evaluations to create an interpolating function $\widetilde{H}$ of H. No information on the structure of H has to be exploited or even known. This allows the use of established methods for rational problems and, as our numerical examples have shown, may provide large gains in computational efficiency.

Since the $\mathcal{L}_\infty$-norm is as a robustness measure for stability of dynamical systems, this method has the potential to become very beneficial in the realm of $\mathcal{H}_\infty$-control. This is currently under investigation for the case of delay systems.

ACKNOWLEDGMENT

We thank Emre Mengi for useful comments on an earlier version of this manuscript.



(a) Initial approximation (b) Iteration 1 (c) Iteration 2

(d) Iteration 3 (e) Iteration 4 (f) Iteration 5

(g) Iteration 6 (h) Iteration 7 (i) Iteration 8

Figure 1. Intermediate rational approximations that are obtained by Algorithm 1 for the `hinfn_ex4` example. The original function is depicted in blue while the approximations are represented by the red dashed lines. The red crosses on the base line indicate the location of the maximizers for the $\mathcal{L}_\infty$-norm of the reduced functions.

REFERENCES

Aliyev, N., Benner, P., Mengi, E., Schwerdtner, P., and Voigt, M. (2017). Large-scale computation of $\mathcal{L}_\infty$-norms by a greedy subspace method. *SIAM J. Matrix Anal. Appl.*, 38(4), 1496–1516.

Antoulas, A.C., Lefteriu, S., and Ionita, A.C. (2017). A tutorial introduction to the Loewner framework for model reduction. In P. Benner, A. Cohen, M. Ohlberger, and K. Willcox (eds.), *Model Reduction and Approximation: Theory and Algorithms*, chapter 8, 335–376. SIAM, Philadelphia, PA, USA.

Beattie, C. and Gugercin, S. (2009). Interpolatory projection methods for structure-preserving model reduction. *Systems Control Lett.*, 58(3), 225–232.

Beattie, C. and Gugercin, S. (2012). Realization-independent $\mathcal{H}_2$ approximation. In *Proc. 51st IEEE Conference on Decision and Control*. Maui, HI, USA.

Benner, P., Sima, V., and Voigt, M. (2012). $\mathcal{L}_\infty$-norm computation for continuous-time descriptor systems using structured matrix pencils. *IEEE Trans. Automat. Control*, 57(1), 233–238.

Benner, P. and Voigt, M. (2014). A structured pseudospectral method for $\mathcal{H}_\infty$-norm computation of large-scale descriptor systems. *Math. Control Signals Systems*, 26(2), 303–338.

Betcke, T., Higham, N.J., Mehrmann, V., Schröder, C., and Tisseur, F. (2013). NLEVP: A collection of nonlinear eigenvalue problems. *ACM Trans. Math. Software*, 39(2), 7:1–7:28.

Boyd, S. and Balakrishnan, V. (1990). 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(1), 1–7.

Bruinsma, N.A. and Steinbuch, M. (1990). A fast algorithm to compute the H_∞-norm of a transfer function

Table 1. Comparison of Algorithm 1 with the method from Mengi et al. (2014). Columns 6 and 7 contain the parameters used for `eigopt`: a lower bound γ for the second derivative of $-\sigma_{\max}(H(\mathrm{i}\cdot))$ and the frequency interval $[\omega_{\min}, \omega_{\max}]$ in which the $\mathcal{L}_\infty$-norm is sought. The number of iterations n_{iter} of Algorithm 1 is given in the eighth column. The relative differences of the computed norm values and the maximizers between the two methods in columns 9 and 10, and the runtimes and their ratios are given in the last three columns.

										time in seconds		
#	example	n	m	p	γ	bounds	n_{iter}	diff. f	diff. ω	Algor. 1	`eigopt`	ratio
1	`delay_model`	500	1	1	-8.35e+01	$[0, 50]$	4	9.59e-08	3.69e-06	0.028	0.821	29.51
2	`butterfly`	64	2	1	-1.61e+01	$[0, 10]$	3	2.82e-15	0.00e+00	0.034	0.017	0.51
3	`dirac`	80	2	1	-2.07e+02	$[0, 10]$	2	8.64e-16	0.00e+00	0.033	0.225	6.75
4	`gen_hyper2`	15	2	1	-4.78e+00	$[0, 100]$	3	3.83e-08	4.29e-04	0.025	0.112	4.52
5	`gen_tantipal2`	16	2	1	-3.41e+02	$[0, 10]$	4	1.36e-08	6.42e-05	0.024	0.032	1.33
6	`gen_tpal2`	16	2	1	-6.67e+02	$[0, 10]$	3	1.71e-07	9.26e-05	0.026	0.036	1.40
7	`hadeler`	8	2	1	-4.41e-01	$[0, 10]$	2	5.16e-16	0.00e+00	0.011	0.023	2.05
8	`loaded_string`	20	2	1	-1.93e+01	$[0, 100]$	2	1.19e-15	0.00e+00	0.011	0.023	2.03
9	`sleeper`	10	2	1	-1.78e+01	$[0, 10]$	2	3.78e-09	7.53e-05	0.018	0.026	1.43
10	`spring`	5	2	1	-7.42e-01	$[0, 10]$	2	4.35e-16	0.00e+00	0.012	0.012	0.95
11	`spring_dashpot`	10	2	1	-1.21e+11	$[0, 0.01]$	3	3.81e-05	1.35e-04	0.020	0.149	7.36
12	`wiresaw2`	10	2	1	-4.36e-01	$[0, 100]$	3	1.15e-07	1.53e-04	0.025	0.048	1.97
13	`hinfn_ex1`	3	1	3	-5.39e+02	$[0, 1]$	2	5.32e-16	0.00e+00	0.009	0.011	1.26
14	`hinfn_ex2`	3	1	3	-2.38e+02	$[0, 1]$	2	1.86e-07	2.27e-04	0.012	0.017	1.38
15	`hinfn_ex3`	1	1	1	-2.16e-01	$[0, 10]$	2	7.96e-10	5.09e-05	0.012	0.013	1.00
16	`hinfn_ex4`	1	1	1	-1.77e+03	$[0, 10]$	9	1.35e-07	1.17e-04	0.157	0.128	0.82
17	`hinfn_ex5`	3	1	3	-2.10e+00	$[0, 100]$	4	1.59e-08	5.13e-05	0.036	0.263	7.36
18	`hinfn_ex6`	3	1	3	-4.46e+01	$[0, 100]$	5	1.37e-10	1.15e-07	0.050	1.743	35.10
19	`hinfn_ex7`	4	2	2	-1.11e+01	$[0, 100]$	8	2.67e-09	1.52e-05	0.151	0.287	1.90
20	`hinfn_ex8`	10	2	4	-4.73e+00	$[0, 100]$	7	1.19e-08	1.39e-04	0.449	0.607	1.35
21	`hinfn_ex9`	20	1	1	-8.32e+01	$[0, 100]$	5	5.54e-10	4.54e-06	0.132	2.310	17.53
22	`hinfn_ex10`	40	2	3	-4.10e+04	$[0, 10]$	4	6.90e-09	1.19e-05	0.070	0.138	1.98
23	`hinfn_ex11`	3	2	2	-1.57e-01	$[0, 100]$	4	1.79e-09	1.15e-05	0.055	0.034	0.61
24	`hinfn_ex12`	3	2	2	-2.79e-01	$[0, 100]$	4	3.15e-08	4.82e-05	0.048	0.040	0.84

Table 2. Comparison of Algorithm 1 (`v1.2`) with the method from Aliyev et al. (2017) (`v1.0`) on the transfer function of the delay model from Beattie and Gugercin (2009).

	time in seconds		
n	`v1.2`	`v1.0`	ratio
100	0.023	1.289	56.208
300	0.025	1.280	51.812
1000	0.029	1.293	43.867
3000	0.046	1.283	28.128
10000	0.093	1.305	13.997
30000	0.262	1.433	5.470
100000	0.872	1.866	2.138
300000	2.509	3.096	1.234
1000000	8.606	7.511	0.873

Table 3. Comparison of the runtimes of Algorithm 1 and `hinfn` from Michiels and Gumussoy (2010).

		time in seconds		
#	example	Algor. 1	`hinfn`	ratio
1	`hinfn_ex1`	0.024	0.063	2.645
2	`hinfn_ex2`	0.015	0.121	8.284
3	`hinfn_ex3`	0.011	0.029	2.665
4	`hinfn_ex4`	0.147	0.045	0.309
5	`hinfn_ex5`	0.039	0.306	7.895
6	`hinfn_ex6`	0.048	0.229	4.721
7	`hinfn_ex7`	0.143	0.230	1.610
8	`hinfn_ex8`	0.462	1.975	4.271
9	`hinfn_ex9`	0.098	40.219	408.498
10	`hinfn_ex10`	0.064	233.989	3657.742
11	`hinfn_ex11`	0.051	0.128	2.483
12	`hinfn_ex12`	0.047	–	–

matrix. *Systems Control Lett.*, 14(4), 287–293.

Freitag, M.A., Spence, A., and Van Dooren, P. (2014). Calculating the H_∞-norm using the implicit determinant method. *SIAM J. Matrix Anal. Appl.*, 35(2), 619–635.

Gantmacher, F.R. (1959). *The Theory of Matrices*, volume 2. AMS Chelsea Publishing, Providence, RI, USA.

Guglielmi, N., Gürbüzbalaban, M., and Overton, M.L. (2013). Fast approximation of the H_∞-norm via optimization over spectral value sets. *SIAM J. Matrix Anal. Appl.*, 34(2), 709–737.

Gumussoy, S. and Michiels, W. (2009). Computing $\mathcal{H}_\infty$ norms of time-delay systems. In *Proc. Joint 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference*, 263–268. Shanghai, China.

Gumussoy, S. and Michiels, W. (2015). Computation of extremum singular values and the strong H-infinity norm of SISO time-delay systems. *Automatica*, 54, 266–271.

Mayo, A.J. and Antoulas, A.C. (2007). A framework for the solution of the generalized realization problem. *Linear Algebra Appl.*, 425(4), 634–662.

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

Michiels, W. and Gummussoy, S. (2010). Characterization and computation of $\mathcal{H}_\infty$ norms for time-delay systems. *SIAM J. Matrix Anal. Appl.*, 31(4), 2093–2115.

Mitchell, T. and Overton, M.L. (2016). Hybrid expansion-contraction: a robust scaleable method for approximating the H_∞ norm. *IMA J. Numer. Anal.*, 36(3), 985–1014.