POLITECNICO DI TORINO
Repository ISTITUZIONALE

Constrained rational fitting for D-stable model order reduction

Original
Constrained rational fitting for D-stable model order reduction / Bradde, T., Grivet-Talocia, S.. - In: ADVANCES IN COMPUTATIONAL MATHEMATICS. - ISSN 1019-7168. - ELETTRONICO. - 52:2(2026). [10.1007/s10444-026-10297-5]

Availability:
This version is available at: 11583/3009960 since: 2026-04-16T11:55:44Z

Publisher:
Springer

Published
DOI:10.1007/s10444-026-10297-5

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

28 November 2026

Advances in Computational Mathematics (2026) 52:23
https://doi.org/10.1007/s10444-026-10297-5

Constrained rational fitting for $\mathcal{D}$-stable model order reduction

Tommaso Bradde[1] · **Stefano Grivet-Talocia**[1]

Received: 22 July 2025 / Accepted: 27 February 2026
© The Author(s) 2026

Abstract
A classical problem in data-driven model order reduction (MOR) of linear time-invariant (LTI) systems is the preservation of structural properties of the underlying large-scale dynamics. When dealing with MOR based on transfer function measurements, one relevant problem is how to force the reduced-order model (ROM) to inherit the asymptotic stability of the reference system, i.e., to enforce the poles of the ROM transfer function to have strictly negative real part. In this work, we tackle the more general problem of placing such poles in arbitrary linear matrix inequality (LMI) regions of the complex plane, which include a rich class of convex sets symmetric with respect to the real axis. LTI systems with poles constrained to this kind of regions are called $\mathcal{D}$-stable. Combining well-established results from control theory and recent developments in asymptotically stable rational approximation algorithms, we show that the problem can be solved efficiently via standard convex optimization routines. Several numerical testbenches of engineering interest confirm the effectiveness of the proposed methodology in practical applications.

Keywords Reduced-order modeling · Data-driven method · Rational approximation · Frequency-domain data · Stability enforcement · D-stability · Barycentric forms · Vector fitting

Mathematics Subject Classification (2010) 41A20 · 93D20 · 74H55 · 93C05 · 93C80

Communicated by: Mario Ohlberger

✉ Tommaso Bradde
tommaso.bradde@polito.it

Stefano Grivet-Talocia
stefano.givet@polito.it

[1] Department of Electronics and Telecommunications, Politecnico di Torino, 10129 Torino, Italy

Published online: 24 March 2026

1 Introduction

In virtually all areas of science and engineering, large-scale dynamical models are essential for obtaining qualitative and quantitative predictions of physical phenomena. For such models, predictive accuracy typically increases with complexity to an extent that most engineering application scenarios may require excessive computing resources or runtime. Consequently, depending on the application, practitioners must balance accuracy against computational cost in order to provide useful results with minimal effort.

Model order reduction (MOR) is a widespread approach to address this challenge [6]. The objective of MOR is to simplify an original full-order model (FOM) by reducing its complexity, yet preserving the level of accuracy that is required for the specific application on some output quantities of interest. Depending on the available information on the FOM, different classes of MOR algorithms have been developed over the years. When a first-principle model is available, typically in the form of discretized partial differential equations (PDE) or large-scale state-space or descriptor forms (e.g., in electronic circuit design flows), classical MOR approaches perform projection or truncation of the original large-scale system. These approaches have been studied and developed extensively and are not considered in this work. A recent review can be found in the three-volume book [6, 58, 60]

A closed-form FOM representation is not always available, and the original system may be characterized or observed only through its input–output behavior. This behavior can be in the form of sampled time-domain responses to a given set of input stimuli, or alternatively in the form of samples of the transfer function at prescribed frequency points; the latter case applies only to Linear systems. Building a ROM starting from input–output data is often referred to as Data-Driven MOR (DDMOR) [3, 6, 58]. This task can be achieved via numerical optimization of the ROM coefficients based on a suitable parameterization [7, 10, 34, 53], via direct realization of the system from the available data [42], by exploiting some physical insight [28], or by enforcing optimality conditions under some prescribed measure [33, 35, 43]. To improve performance and efficiency, iterative greedy schemes have also been proposed [18, 44, 48].

This paper focuses on DDMOR of LTI systems based on frequency samples of the FOM transfer function. Most commonly, such samples are obtained from direct measurements, when hardware is available, or from virtual measurements. In the latter case, the frequency responses are typically produced from closed-box commercial solvers, which do not provide access to the FOM equations but only to their evaluation results. Since the transfer functions of LTI systems are known to admit approximations in terms of finite-order rational functions, DDMOR can be performed by finding the coefficients of a candidate reduced-order rational function that accurately fits the available data. This rational approximation problem has been tackled extensively in the literature, following two main approaches. Interpolation-based methods [3, 18, 28, 42, 44, 48] aim at matching the ROM response exactly at the available data points. The most prominent approach in this class is the so-called AAA algorithm [44]. Conversely, optimization-based approaches [7, 10, 34, 35, 43, 53] allow for some approximation error at the data points, in the attempt to increase robustness to noise and/or enforce some additional model structure. The Vector Fitting (VF) algorithm [31, 34, 54] is

regarded as the most representative scheme in this class, also based on its ubiquitous availability in commercial tools for engineering applications [62–64].

The above DDMOR approaches have been originally developed to derive a compact model in case the FOM is not accessible through its internal variables or equations. However, DDMOR techniques have been proven so effective and successful in applications that several approaches have been presented to extend their application scope also to solve the more "classical" MOR problem, as applied to reduce the complexity of a fully-detailed FOM description. In this scenario, the frequency samples can be evaluated during the modeling process, allowing for additional adaptive and/or greedy strategies to optimize performance and accuracy. The very successful algorithm denoted as Iterative Rational Krylov Approximation (IRKA) [33], that is able to attain $\mathcal{H}_2$ optimality, has also been adapted to the problem of DDMOR based on frequency-domain data [61]. The algorithm that is presented in this work assumes the availability of a fixed set of frequency-domain measurements, is agnostic to the particular procedure used to generate them, and is intended to be a DDMOR algorithm of general applicability. The novel contribution that is proposed herein is related to additional model properties that are enforced during its construction from this data.

When ROMs are required to perform time-domain simulations, they should inherit the causality, stability, and passivity properties of the FOM [12, 55]. Whereas causality is easily enforced and passivity is crucial only when ROMs are inserted in larger system interconnections, the preservation of stability is a general and essential prerequisite. Since in any numerical simulation, spurious numerical instabilities are to be avoided, only rational fitting procedures that are endowed with stability guarantees can be considered effective. This requirement made algorithms like VF [14–16, 21, 24, 34, 40] a preferred choice in professional DDMOR tools used in engineering, paving the way to several investigations in the last two decades. In particular, the problem of stability enforcement has been tackled via constrained optimization or perturbation strategies, typically applied in a post-processing step after instability has been detected. A representative though not exhaustive list of such approaches includes [29, 30, 35, 36, 43, 47, 50, 51, 53, 55]. For interpolatory approaches such as the AAA algorithm [44], stability has been classically enforced via unstable pole flipping or truncation, a combination of both [5, 17, 23, 38, 56], or by a search for the closest stable approximation to the originally unstable interpolant [26, 27]. More recently, a very effective approach based on semidefinite programming has been proposed [11].

In this article, we generalize the problem of ROM stability enforcement, i.e., we propose a fitting approach able to constrain the poles of the resulting rational model not only over the left-half complex plane, but over an arbitrary *linear matrix inequality (LMI) region* of such plane. Theoretically, this class of sets includes arbitrarily accurate approximations for generic convex sets that are symmetric with respect to the real axis [19]. LTI systems having poles in such regions are called $\mathcal{D}$-stable [19]. Inspired by the recent developments regarding asymptotic stability in the interpolatory AAA framework [11], our approach consists of augmenting the optimization stage of a rational fitting algorithm (VF) with a set of constraints that explicitly enforce the desired pole locations. Applying an approach similar to the one of [11], the resulting optimization algorithm is relaxed to a convex semidefinite program. The relaxation

is enabled by weighting the cost function that is iteratively minimized by the rational fitting process.

Our work fits and is somehow complementary with recent developments by Borghi et al. [8, 9], that allow to perform the reduction process while preserving the location of the ROMs poles over general domains. Unlike these approaches, the method presented here is fully data-driven and immediately transferable to scenarios in which the input–output behavior of the FOM is characterized in the time domain [10]. Our derivations will be limited to the case of scalar rational approximation via constrained VF. However, we remark that:

1. The extension to constrained approximation of rational matrix functions is straightforward, since in both cases the VF optimization routine solves the same least-squares problem; in both scalar and matrix cases, this problem is augmented by the constraints that are introduced in this work.
2. We chose to embed the proposed constraints into the VF iteration, that we consider here as the reference algorithm. However, the results and the derivations can be transferred to other rational approximation algorithms that employ a similar model structure with virtually no modifications (e.g., in the AAA framework, with the model structure introduced in [37]).

The proposed approach is applied to three different test problems of engineering interest. We show how properly choosing a $\mathcal{D}$-stability region can improve the performance of the VF algorithm under certain modeling requirements such as limiting the maximum bandwidth of the desired ROM; also, we show how the method can be applied to fit transfer functions corrupted by significant amount of noise, a situation that is commonly encountered in data-driven modeling scenarios.

2 Notation and problem statement

In the following, $s = \sigma + j\omega \in \mathbb{C}$ and $j = \sqrt{-1}$ denote respectively the Laplace variable and the imaginary unit. Scalar numbers will be denoted with lowercase italic letters, while uppercase italic fonts will be reserved for scalar functions of the variable s (i.e., $z = H(s) : \mathbb{C} \to \mathbb{C}$). Given $z \in \mathbb{C}$, z^* represents its conjugate. The operators $\mathfrak{Re}\{\cdot\}$ and $\mathfrak{Im}\{\cdot\}$ yield the real and imaginary parts of their arguments. Bold letters will be used for vectors and matrices, whose size will be always made explicit (e.g., $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{X} \in \mathbb{R}^{n\times n}$). The matrix $\mathbf{A}^T$ is the transpose of $\mathbf{A}$. The expression $\mathbf{A} \otimes \mathbf{B}$ denotes the Kronecker product between $\mathbf{A}$ and $\mathbf{B}$. The matrix $\mathbf{I}_n$ is the identity matrix of size n. The expression $\mathbf{J} \succ \mathbf{K}$ ($\mathbf{J} \succeq \mathbf{K}$) means that the matrix $\mathbf{J} - \mathbf{K}$ is positive (semi)definite. A sequence of numbers indexed by a subscript k, e.g., x_k with $1 \leq k \leq K$, will be shortly denoted as $\{x_k\}$. The length of the sequence will be assumed equal to the uppercased subscript letter (e.g., $\{x_k\}$ has cardinality K).

Let us consider a single-input-single-output LTI dynamical system $\hat{\mathbf{\Sigma}}$ in descriptor form

$$\hat{\mathbf{\Sigma}} : \begin{cases} \hat{\mathbf{E}}\,\dot{\boldsymbol{x}}(t) = \hat{\mathbf{A}}\,\boldsymbol{x}(t) + \hat{\mathbf{B}}\,u(t) \\ y(t) \quad\;\; = \hat{\mathbf{C}}\,\boldsymbol{x}(t) + \hat{d}u(t) \end{cases}, \tag{1}$$

with $\boldsymbol{x}(t) \in \mathbb{R}^N$ as the state variables, $u(t) \in \mathbb{R}$ as the control input, and $y(t) \in \mathbb{R}$ is the observed output, so that $\hat{\mathbf{A}}, \hat{\mathbf{E}} \in \mathbb{R}^{N\times N}$, $\hat{\mathbf{B}} \in \mathbb{R}^N$, $\hat{\mathbf{C}}^T \in \mathbb{R}^N$, $\hat{d} \in \mathbb{R}$. When the matrix $\hat{\mathbf{E}}$ is non-singular, system (1) is equivalent to a state-space realization for the system $\hat{\mathbf{\Sigma}}$. The transfer function associated with (1) is given by $\hat{H}(s) \in \mathbb{C}$

$$\hat{H}(s) = \hat{\mathbf{C}}(s\hat{\mathbf{E}} - \hat{\mathbf{A}})^{-1}\hat{\mathbf{B}} + \hat{d}. \tag{2}$$

Since the state-space realization (1) is defined upon real matrices, the system $\hat{\mathbf{\Sigma}}$ has a real impulse response and the transfer function (2) verifies the condition $\hat{H}^*(s) = \hat{H}(s^*)$.

In our setting, the system $\hat{\mathbf{\Sigma}}$ is known only in terms of evaluations of its transfer function retrieved at locations s_k, $k = 1, \ldots, K$ in the complex plane. We denote each of these measurements as $\hat{H}_k = \hat{H}(s_k)$. This article presents a solution to the following problem

Problem Statement 1 Given a full order model with transfer function $\hat{H}(s)$, a set of sampling points $\{s_k\}$ and corresponding sampled data $\{\hat{H}_k\}$, compute a LTI ROM with transfer function $H(s)$ and state-space realization $(\mathcal{A}, \mathcal{B}, \mathcal{C}, d)$ with $\mathcal{A} \in \mathbb{R}^{n\times n}$, $\mathcal{B}, \mathcal{C}^T \in \mathbb{R}^n$ and $d \in \mathbb{R}$ such that

***C*.1** $\mathcal{A} \in \mathbb{R}^{n\times n}$ with $n \ll N$,
***C*.2** $H(s) \approx \hat{H}(s)\ \forall s \in \{s_k\}$
***C*.3** $\text{eig}\{\mathcal{A}\} \subset \mathcal{D} = \{s \in \mathbb{C} : \mathbf{L} + s\mathbf{M} + s^*\mathbf{M}^T \prec 0\}$ where $\mathbf{L}$ and $\mathbf{M}$ are real matrices and $\mathbf{L}$ is symmetric.

The set $\mathcal{D}$ is called an LMI region of the complex plane [19, 20, 22] and is convex and symmetric with respect to the real axis. LMI regions include (but are not limited to) half-planes, horizontal strips, cones, disks, ellipses, and arbitrary intersections of these sets. Indeed, any convex region that is symmetric with respect to the real axis can be approximated with arbitrarily high accuracy by an LMI region [19, 20]. See [22, Sec. 3] for an explicit characterization of some relevant regions in terms of matrices $\mathbf{M}$, $\mathbf{L}$. When an LTI system satisfies condition **C.3**, it is called $\mathcal{D}$-stable.

Four examples of LMI regions are shown in Fig. 1 and include

1. the open left-half complex plane with $\Re\{s\} < -\alpha = -0.5$ defined by the LMI region with $\mathbf{L} = 2\alpha$ and $\mathbf{M} = 1$,
2. the interior of the unitary disk centered at the origin, with

$$\mathbf{L}_D = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}, \quad \mathbf{M}_D = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \tag{3}$$

3. the intersection between the above region and a cone centered at the origin and aperture $2 \times \theta = 2 \times \pi/3$, defined by $\mathbf{M} = \text{blkdiag}\{\mathbf{M}_\text{D}, \mathbf{M}_\text{cone}\}$, and $\mathbf{L} = \text{blkdiag}\{\mathbf{L}_\text{D}, \mathbf{L}_\text{cone}\}$, where

$$\mathbf{M}_\text{cone} = \begin{bmatrix} \sin\theta & \cos\theta \\ -\cos\theta & \sin\theta \end{bmatrix}, \quad \mathbf{L}_\text{cone} = \mathbf{0}, \tag{4}$$

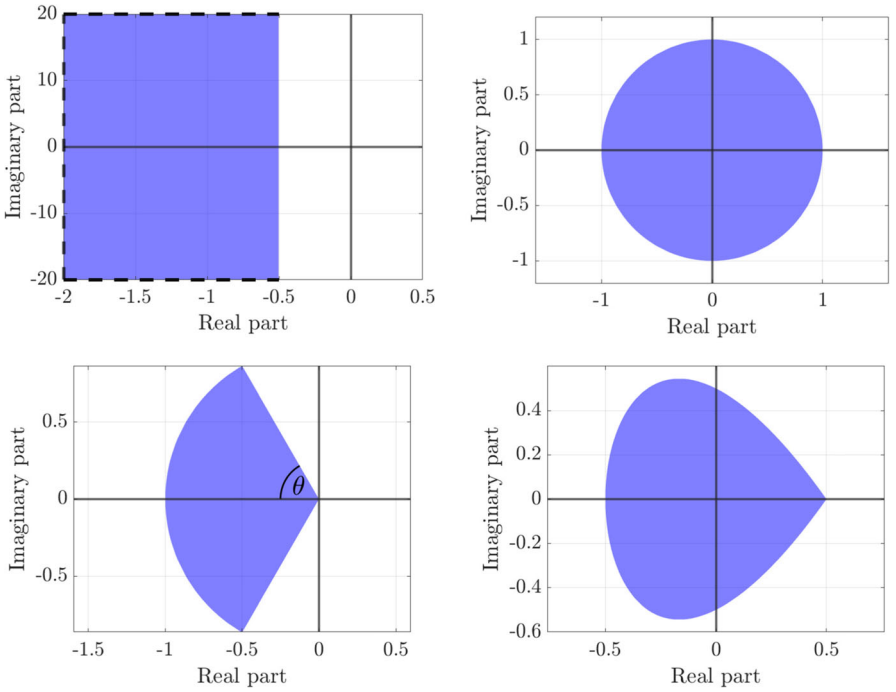


Fig. 1 Four examples of LMI regions

4. an unconventional LMI region taken from [39], with

$$\mathbf{L} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}, \quad \mathbf{M} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 1 \\ 1 & 1 & 0 \end{bmatrix}. \tag{5}$$

3 Background: vector fitting

Our approach for generating $\mathcal{D}$-stable ROMs builds on the well-known VF iteration. For completeness, we provide a brief overview of VF, since this rational approximation algorithm is well described in the literature. The interested reader is referred to [31, 34, 54] for an in-depth analysis. Here, we limit the exposition to the aspects that are strictly necessary to introduce the proposed innovation.

VF seeks for a rational function $H(s)$ in pole-residue form

$$H(s) = \sum_{i=1}^{n} \frac{r_i}{s - p_i} + r_\infty, \tag{6}$$

that meets the reduced order model structure **C. 1** by construction, following a standard state-space realization process. The approximation condition **C. 2** is achieved by optimizing the model unknowns $\{p_i\}, \{r_i\}, r_\infty$ to fit the available data samples in a

least-squares sense. To this aim, the pole-residue representation (6) is implicitly and iteratively rewritten in terms of the modified barycentric form employed in the VF iteration [34]

$$H^{(\mu)}(s) = \frac{N^{(\mu)}(s)}{D^{(\mu)}(s)}, \quad N^{(\mu)}(s) = \nu_0^{(\mu)} + \sum_{i=1}^{n} \frac{\nu_i^{(\mu)}}{s - q_i^{(\mu)}}, \quad D^{(\mu)}(s) = 1 + \sum_{i=1}^{n} \frac{\delta_i^{(\mu)}}{s - q_i^{(\mu)}} \tag{7}$$

where μ is an iteration index and $H^{(\mu)}(s)$ is the estimate of $H(s)$ at iteration μ, expressed via the instrumental functions $N^{(\mu)}(s)$ and $D^{(\mu)}(s)$. In (7), the poles $\{q_i^{(\mu)}\}$ of $N^{(\mu)}(s)$ and $D^{(\mu)}(s)$ are known complex scalars (set is closed under complex conjugation), while $\nu_0^{(\mu)}$, $\{\nu_i^{(\mu)}\}$, $\{\delta_i^{(\mu)}\}$ are the iteration-dependent complex model parameters. During each iteration, two main operations are performed, namely unknowns optimization and pole relocation. After the iteration stops, a final step is performed to return the required rational approximation in the pole-residue form (6).

Optimization Under model structure (7), the iteration-dependent unknowns are optimized by enforcing the modified approximation condition

$$N^{(\mu)}(s) \approx \hat{H}(s) D^{(\mu)}(s) \quad \forall s \in \{s_k\} \tag{8}$$

that is the linearized counterpart of **C.2**. Since (8) is linear in the model parameters, it can be easily enforced in a least-squares sense over the available data-points, as commonly done also in other state-of-the-art rational approximation methods [2, 44].

From the practical standpoint, setting up the subsequent pole relocation stage requires only the knowledge of the denominator unknowns $\{\delta_i^{(\mu)}\}$. Hence, during the optimization stage, the numerator residues $\nu_0^{(\mu)}$, $\{\nu_i^{(\mu)}\}$ are commonly eliminated applying the QR-based decoupling introduced in [24], that also greatly improves the efficiency of the algorithm in case of massively MIMO systems (not covered here). This approach returns the optimal value of the parameters $\{\delta_i^{(\mu)}\}$ as the solution of a least-squares problem of the form

$$\min_{\tilde{\mathbf{C}}^{(\mu)}} \left\| \mathbf{R} \tilde{\mathbf{C}}^{(\mu)T} - \tilde{\mathbf{b}} \right\|_2^2 \tag{9}$$

where $\mathbf{R}$, $\tilde{\mathbf{b}}$ are a data matrix and a data vector. See [24] for full details about their definitions. The vector

$$\tilde{\mathbf{C}}^{(\mu)} = [\delta_1^{(\mu)}, \delta_2^{(\mu)}, \ldots, \delta_n^{(\mu)}] \in \mathbb{R}^{1 \times n} \tag{10}$$

collects the optimal denominator parameters of the current iteration.

Pole relocation The objective of this step is to update the basis poles $\{q_i^{(\mu)}\}$ of $N^{(\mu)}(s)$ and $D^{(\mu)}(s)$ defining the model structure (7). The update is performed by noticing that in (7), the poles of $H^{(\mu)}(s)$ coincide with the zeros of the denominator function $D^{(\mu)}(s)$. The availability of the optimal $\tilde{\mathbf{C}}^{(\mu)}$ retrieved as the solution of (9) allows to

redefine these zeros explicitly. Defining the realization

$$D^{(\mu)}(s) = \mathbf{C}^{(\mu)}(s\mathbf{I} - \tilde{\mathbf{A}}^{(\mu)})^{-1}\tilde{\mathbf{B}}^{(\mu)} + 1, \tag{11}$$

where the constant matrices $\tilde{\mathbf{A}}^{(\mu)}$, $\tilde{\mathbf{B}}^{(\mu)}$ are

$$\tilde{\mathbf{A}}^{(\mu)} = \text{blkdiag}\{\tilde{\mathbf{A}}_i^{(\mu)}\} \in \mathbb{R}^{n\times n} \quad \tilde{\mathbf{B}}^{(\mu)} = [\tilde{\mathbf{B}}_1^{(\mu)}; \tilde{\mathbf{B}}_2^{(\mu)}; \ldots; \tilde{\mathbf{B}}_n^{(\mu)}] \in \mathbb{R}^n, \tag{12}$$

the semicolon denotes the column-stacking operation and

$$\tilde{\mathbf{A}}_i^{(\mu)} = \begin{cases} q_i^{(\mu)}, & q_i^{(\mu)} \in \mathbb{R} \\ \begin{bmatrix} \sigma_i^{(\mu)} & \omega_i^{(\mu)} \\ -\omega_i^{(\mu)} & \sigma_i^{(\mu)} \end{bmatrix}, & q_i^{(\mu)} = \sigma_i^{(\mu)} \pm j\omega_i^{(\mu)} \in \mathbb{C} \end{cases} \tag{13}$$

$$\tilde{\mathbf{B}}_i^{(\mu)} = \begin{cases} 1, & q_i^{(\mu)} \in \mathbb{R} \\ \begin{bmatrix} 2 ; 0 \end{bmatrix}, & q_i^{(\mu)} = \sigma_i^{(\mu)} \pm j\omega_i^{(\mu)} \in \mathbb{C} \end{cases} \tag{14}$$

we have that $(D^{(\mu)}(s))^{-1}$ can be realized as $(\tilde{\mathbf{A}}^{(\mu)} - \tilde{\mathbf{B}}^{(\mu)}\tilde{\mathbf{C}}^{(\mu)}, \tilde{\mathbf{B}}^{(\mu)}, \tilde{\mathbf{C}}^{(\mu)}, 1)$ so that the zeros of $D^{(\mu)}(s)$ are computed as $\text{eig}\{\tilde{\mathbf{A}}^{(\mu)} - \tilde{\mathbf{B}}^{(\mu)}\tilde{\mathbf{C}}^{(\mu)}\}$. Hence the pole relocation defines the quantities $\{q_i^{(\mu+1)}\}$ for the next iteration according to the rule

$$\{q_i^{(\mu+1)}\} \equiv \text{eig}\{\tilde{\mathbf{A}}^{(\mu)} - \tilde{\mathbf{B}}^{(\mu)}\tilde{\mathbf{C}}^{(\mu)}\}. \tag{15}$$

Stability of the MOR is achieved by *pole flipping*, i.e., by simply mirroring any element of $\{q_i^{(\mu+1)}\}$ having a positive real part into the left-half complex plane at each iteration.

Ideally, the convergence condition for the iteration is met when the estimate of $\{q^{(\mu)}\}$ stabilizes that is when

$$\{q_i^{(\mu+1)}\} = \{q_i^{(\mu)}\}, \tag{16}$$

which requires $\tilde{\mathbf{C}}^{(\mu)}$ to vanish in (15). Under this condition, it holds $D^{(\mu)}(s) = 1$ and the linearized approximation condition (8) becomes equivalent to the actually desired relation **C2**. In practice, as shown in [40], condition (16) may not be attained when $n \neq N$, as often happens in MOR scenarios. Hence, empirical stop conditions are enforced in practice to terminate the VF iteration (e.g., hitting of a maximum prescribed number of iterations, or requiring $\left\|\tilde{\mathbf{C}}^{(\mu)}\right\|_2$ below a given tolerance).

Residues optimization Assuming that some stopping condition is met after $\bar{\mu}$ iterations, the rational approximation in pole-residue form is finally obtained by setting $\{p_i\} \equiv \{q_i^{(\bar{\mu})}\}$ in (6) and performing a linear regression to estimate the residues $\{r_i\}$, r_∞. This final step completes the model generation. Once the ROM transfer function

$H(s)$ is available, it can be realized as a minimal state-space system $(\mathcal{A}, \mathcal{B}, \mathcal{C}, d)$ using standard approaches.

4 Enforcing $\mathcal{D}$-stability

The VF iteration outlined in Section 3 is among the methods of choice to build ROMs fulfilling requirements **C.1** and **C.2** defined in the problem statement. Our contribution here is to adapt the algorithm optimization step in order to guarantee the fulfillment of the additional requirement **C.3**. We make use of the following theorem, proved in [19]

Theorem 1 *Given an LMI region* $\mathcal{D} = \{s \in \mathbb{C} : \boldsymbol{L} + s\boldsymbol{M} + s^*\boldsymbol{M}^T \prec 0\}$ *(with* $\boldsymbol{L}$ *and* $\boldsymbol{M}$ *real matrices with* $\boldsymbol{L}$ *symmetric) and a matrix* $\boldsymbol{X}$*, it holds* $\text{eig}\{\boldsymbol{X}\} \subset \mathcal{D}$ *if and only if there exists a symmetric matrix* $\boldsymbol{Q} \succ 0$ *such that*

$$\boldsymbol{L} \otimes \boldsymbol{Q} + \boldsymbol{M} \otimes (\boldsymbol{X}\boldsymbol{Q}) + \boldsymbol{M}^T \otimes (\boldsymbol{X}\boldsymbol{Q})^T \prec 0. \tag{17}$$

The above theorem provides explicit algebraic conditions to check the $\mathcal{D}$-stability of a given matrix and hence of an arbitrary LTI system. The following descends naturally

Proposition 1 *Let* $\mathcal{D}$ *be an LMI region defined as in Theorem 1, and let* μ *be an arbitrary VF iteration. Under the barycentric model representation* (7) *and the state-space realization* (11)*, it holds*

$$\{q_i^{(\mu+1)}\} \subset \mathcal{D} \tag{18}$$

if and only if there exists $\boldsymbol{Q} \succ 0$ *such that*

$$\boldsymbol{L} \otimes \boldsymbol{Q} + \boldsymbol{M} \otimes (\tilde{\boldsymbol{A}}^{(\mu)}\boldsymbol{Q} - \tilde{\boldsymbol{B}}^{(\mu)}\tilde{\boldsymbol{C}}^{(\mu)}\boldsymbol{Q}) + \boldsymbol{M}^T \otimes (\tilde{\boldsymbol{A}}^{(\mu)}\boldsymbol{Q} - \tilde{\boldsymbol{B}}^{(\mu)}\tilde{\boldsymbol{C}}^{(\mu)}\boldsymbol{Q})^T \prec 0. \tag{19}$$

The model $H(s) \leftrightarrow (\mathcal{A}, \mathcal{B}, \mathcal{C}, d)$ *is* $\mathcal{D}$*-stable if* (19) *is verified at the last VF iteration* $\mu = \bar{\mu}$.

Proposition 1 provides the exact algebraic condition that must be verified to meet requirement **C.3** of our problem statement. Clearly, such condition is not necessarily met by the optimal solution retrieved by the unconstrained least-squares (9). Hence, a constrained optimization approach must be pursued. Such approach is presented next.

4.1 Constrained pole relocation

In the following derivations, we drop any iteration index μ, since irrelevant. Our numerical approach for enforcing $\mathcal{D}$-stability of $H(s)$ starts by constraining (9):

$$\min_{\tilde{\mathbf{C}}, \mathbf{Q}} \left\| \mathbf{R}\tilde{\mathbf{C}}^T - \tilde{\mathbf{b}} \right\|_2^2 \tag{20}$$

$$
\begin{aligned}
&\text{subject to :}\\
&\mathbf{Q} = \mathbf{Q}^T \succ 0\\
&\mathbf{L} \otimes \mathbf{Q} + \mathbf{M} \otimes (\tilde{\mathbf{A}}\mathbf{Q} - \tilde{\mathbf{B}}\tilde{\mathbf{C}}\mathbf{Q}) + \mathbf{M}^T \otimes (\tilde{\mathbf{A}}\mathbf{Q} - \tilde{\mathbf{B}}\tilde{\mathbf{C}}\mathbf{Q})^T \prec 0.
\end{aligned}
$$

Solving problem (20) allows to enforce exactly the linearized approximation condition (8) required by VF, while at the same time constraining the pole relocation stage to the LMI region defined by $\mathbf{M}$ and $\mathbf{L}$. From the computational standpoint, (20) is a semidefinite optimization problem with bilinear constraints arising from the products $\tilde{\mathbf{C}}\mathbf{Q}$. Although several methods based on convex relaxations have been proposed to address this type of problem (see, e.g., [57]), they do not guarantee recovery of the global solution. Conversely, directly solving the problem using global optimization techniques would only be practical when the number of optimization variables is very small. Hence, in this work, we propose to apply a relaxation to problem (20) in order to formulate the constrained optimization as a convex program. The approach we follow here is closely related to the one derived in [11] to enforce stability of ROMs generated via the AAA algorithm.

We start by simplifying (20), defining $\mathbf{R} = \mathbf{U}\Sigma\mathbf{V}^T$ the "thin" SVD of $\mathbf{R}$, and rewriting the problem equivalently as

$$
\begin{aligned}
&\min_{\tilde{\mathbf{C}},\mathbf{Q}} \left\| \Sigma\mathbf{V}^T\tilde{\mathbf{C}}^T - \mathbf{U}^T\tilde{\mathbf{b}} \right\|_2^2 \qquad (21)\\
&\text{subject to :}\\
&\mathbf{Q} = \mathbf{Q}^T \succ 0\\
&\mathbf{L} \otimes \mathbf{Q} + \mathbf{M} \otimes (\tilde{\mathbf{A}}\mathbf{Q} - \tilde{\mathbf{B}}\tilde{\mathbf{C}}\mathbf{Q}) + \mathbf{M}^T \otimes (\tilde{\mathbf{A}}\mathbf{Q} - \tilde{\mathbf{B}}\tilde{\mathbf{C}}\mathbf{Q})^T \prec 0,
\end{aligned}
$$

We then introduce the state-space transformation matrix $\mathbf{T} = \mathbf{V}\Sigma$, so that, based on (11) we obtain

$$
D(s) \leftrightarrow (\underbrace{\mathbf{T}^{-1}\tilde{\mathbf{A}}\mathbf{T}}_{\mathbf{A}}, \underbrace{\mathbf{T}^{-1}\tilde{\mathbf{B}}}_{\mathbf{B}}, \underbrace{\tilde{\mathbf{C}}\mathbf{T}}_{\mathbf{C}}, 1) \leftrightarrow (\mathbf{A}, \mathbf{B}, \mathbf{C}, 1), \qquad (22)
$$

and, with the definition $\mathbf{b} = \mathbf{U}^T\tilde{\mathbf{b}}$, we rewrite (21) as

$$
\begin{aligned}
&\min_{\mathbf{C},\mathbf{Q}} \left\| \mathbf{C}^T - \mathbf{b} \right\|_2^2 \qquad (23)\\
&\text{subject to :}\\
&\mathbf{Q} = \mathbf{Q}^T \succ 0\\
&\mathbf{L} \otimes \mathbf{Q} + \mathbf{M} \otimes (\mathbf{A}\mathbf{Q} - \mathbf{B}\mathbf{C}\mathbf{Q}) + \mathbf{M}^T \otimes (\mathbf{A}\mathbf{Q} - \mathbf{B}\mathbf{C}\mathbf{Q})^T \prec 0,
\end{aligned}
$$

which is equivalent to (20) since $\text{eig}\{\tilde{\mathbf{A}} - \tilde{\mathbf{B}}\tilde{\mathbf{C}}\} = \text{eig}\{\mathbf{A} - \mathbf{BC}\}$. With the variable transformation $\mathbf{CQ} = \boldsymbol{\delta}$ we obtain

$$\begin{aligned}
&\min_{\boldsymbol{\delta},\mathbf{Q}} \left\| \mathbf{Q}^{-1}\boldsymbol{\delta}^T - \mathbf{b} \right\|_2^2 = \min_{\boldsymbol{\delta},\mathbf{Q}} (\boldsymbol{\delta}^T - \mathbf{Qb})^T \mathbf{Q}^{-1}\mathbf{Q}^{-1}(\boldsymbol{\delta}^T - \mathbf{Qb}) \qquad (24)\\
&\text{subject to :}\\
&\mathbf{Q} = \mathbf{Q}^T \succ 0\\
&\mathbf{L} \otimes \mathbf{Q} + \mathbf{M} \otimes (\mathbf{AQ} - \mathbf{B}\boldsymbol{\delta}) + \mathbf{M}^T \otimes (\mathbf{AQ} - \mathbf{B}\boldsymbol{\delta})^T \prec 0.
\end{aligned}$$

Minimizing the (unconstrained) cost function in (24) is equivalent to minimizing the following epigraphic form based on the instrumental variable t

$$\begin{aligned}
&\min_{\boldsymbol{\delta},\mathbf{Q},t} t\\
&\text{subject to :}\\
&t \geq (\boldsymbol{\delta}^T - \mathbf{Qb})^T \mathbf{Q}^{-1}\mathbf{Q}^{-1}(\boldsymbol{\delta}^T - \mathbf{Qb}) \qquad (25)
\end{aligned}$$

Applying the (inverse) Schur complement to (25) and equipping the resulting optimization problem with the $\mathcal{D}$-stability constraints gives the following

$$\begin{aligned}
&\min_{t,\mathbf{Q},\boldsymbol{\delta}} t && (26a)\\
&\text{subject to :} && (26b)\\
&\mathbf{Q} = \mathbf{Q}^T \succ 0 && (26c)\\
&\mathbf{L} \otimes \mathbf{Q} + \mathbf{M} \otimes (\mathbf{AQ} - \mathbf{B}\boldsymbol{\delta}) + \mathbf{M}^T \otimes (\mathbf{AQ} - \mathbf{B}\boldsymbol{\delta})^T \prec 0. && (26d)\\
&\begin{bmatrix} t & (\boldsymbol{\delta}^T - \mathbf{Qb})^T \\ (\boldsymbol{\delta}^T - \mathbf{Qb}) & \mathbf{QQ} \end{bmatrix} \succeq 0 && (26e)
\end{aligned}$$

The above semidefinite problem is still non-convex due to the presence of the quadratic term $\mathbf{QQ}$ in the constraint (26e). However, a convexification is easily achieved through the relaxation strategy introduced in [11, Sec. 4.3], where it is shown that every feasible solution of the following problem

$$\begin{aligned}
&\min_{t,\mathbf{Q},\boldsymbol{\delta}} t && (27a)\\
&\text{subject to :} && (27b)\\
&\mathbf{Q} = \mathbf{Q}^T \succ 0 && (27c)\\
&\mathbf{L} \otimes \mathbf{Q} + \mathbf{M} \otimes (\mathbf{AQ} - \mathbf{B}\boldsymbol{\delta}) + \mathbf{M}^T \otimes (\mathbf{AQ} - \mathbf{B}\boldsymbol{\delta})^T \prec 0. && (27d)\\
&\begin{bmatrix} t & (\boldsymbol{\delta}^T - \mathbf{Qb})^T \\ (\boldsymbol{\delta}^T - \mathbf{Qb}) & \mathbf{Q} \end{bmatrix} \succeq 0, && (27e)
\end{aligned}$$

is also feasible for (26). Note that the constraint (26e) is replaced in (27) by the LMI (27e), that is linear in $\mathbf{Q}$. The convex problem (27) has a unique solution that can be computed efficiently.

The proposed relaxation can be shown to be equivalent to a modification of the original cost function that is minimized during the VF optimization stage. In fact, applying to (27) the same manipulations that were used to convert problem (24) into problem (26) in reverse order shows that the epigraphic form (27) is equivalent to the following minimization problem

$$\min_{\boldsymbol{\delta},\mathbf{Q}} \left\| \mathbf{Q}^{\frac{1}{2}} (\mathbf{Q}^{-1}\boldsymbol{\delta}^T - \mathbf{b}) \right\|_2^2 = \min_{\boldsymbol{\delta},\mathbf{Q}} \left\| \mathbf{Q}^{-\frac{1}{2}} (\boldsymbol{\delta}^T - \mathbf{Q}\mathbf{b}) \right\|_2^2 . \tag{28}$$

The proposed relaxation can therefore be interpreted as the introduction of an additional weight in the VF residual error, expressed by the (unknown) matrix $\mathbf{Q}^{\frac{1}{2}}$ that is itself a design variable. An apriori estimate of the impact of this weighting on the final model accuracy is difficult to obtain. However, the bias introduced by this weight can be compensated (e.g., using a further relaxation as in [52]) or overbounded as in [57]. These approaches are well documented and not further developed here. We remark that, since $\mathbf{Q}$ is symmetric and positive definite, the only difference between (28) and the original cost function (23) resides in a different scaling of the various components of the residual in a different orthogonal basis. The numerical experiments presented in Section 5 will confirm the observations already discussed in [11], namely that the influence of the bias term is practically negligible.

4.2 Practical implementations

Problem (27) is convex with respect to the decision variables and can be solved using standard interior-point methods, which can compute the optimal solution to arbitrary precision in polynomial time [45]. In practice, solving (27) requires enforcing strict positivity of the left-hand side matrices in constraints (27d) and (27c). This is done by adding a positivity margin to such strict inequalities. Hence, in practice, we solve the following problem

$$\min_{t,\mathbf{Q},\boldsymbol{\delta}} t \tag{29a}$$

$$\text{subject to :} \tag{29b}$$

$$\mathbf{Q} = \mathbf{Q}^T \succeq \sigma \mathbf{I} \tag{29c}$$

$$\mathbf{L} \otimes \mathbf{Q} + \mathbf{M} \otimes (\mathbf{A}\mathbf{Q} - \mathbf{B}\boldsymbol{\delta}) + \mathbf{M}^T \otimes (\mathbf{A}\mathbf{Q} - \mathbf{B}\boldsymbol{\delta})^T \preceq -\sigma \mathbf{I}, \tag{29d}$$

$$\begin{bmatrix} t & (\boldsymbol{\delta}^T - \mathbf{Q}\mathbf{b})^T \\ (\boldsymbol{\delta}^T - \mathbf{Q}\mathbf{b}) & \mathbf{Q} \end{bmatrix} \succeq 0. \tag{29e}$$

where σ is a positive constant. Theoretically, the proposed approach is unaffected by the particular choice of the parameter σ, since by choosing a different threshold $\gamma\sigma$ with $\gamma > 0$, one can define a rescaled set of variables $t_\gamma = \gamma t$, $\mathbf{Q}_\gamma = \gamma \mathbf{Q}$ and

$\boldsymbol{\delta}_\gamma = \gamma\boldsymbol{\delta}$, retrieve the optimal solution triple $\{t^*_\gamma, \mathbf{Q}^*_\gamma, \boldsymbol{\delta}^*_\gamma\}$, and recover the optimal values of the problem before the γ-scaling as

$$t^* = \frac{t^*_\gamma}{\gamma}, \quad \mathbf{Q}^* = \frac{\mathbf{Q}^*_\gamma}{\gamma} \quad \boldsymbol{\delta}^* = \frac{\boldsymbol{\delta}^*_\gamma}{\gamma}. \tag{30}$$

Using this solution, we can check that $\mathbf{C}^* = \mathbf{Q}^{*-1}\boldsymbol{\delta}^* = \mathbf{Q}^{*-1}_\gamma\boldsymbol{\delta}^*_\gamma$. Hence, the optimal solution vector remains unchanged after the variable rescaling. In practice, the numerical solution of the problem is slightly affected by the particular choice of σ. However, as will be shown experimentally in Section 5.3, the sensitivity of the final ROM accuracy on such a parameter is negligible in realistic MOR scenarios. The formal analysis of the asymptotic complexity of these iterative methods largely depends on the specific algorithm used, its implementation, and the underlying structure of the problem. These aspects are not discussed here. From the implementation standpoint, the constrained pole relocation can be performed at any arbitrary iteration. Different choices can be pursued, e.g., performing it throughout every iterations, after a fixed number of unconstrained iterations, or at the last iteration only. Algorithm 1 provides a general pseudo-code of the proposed approach. Besides standard VF input parameters (e.g., n_1, the maximum number of unconstrained iterations) the additional user-defined value n_2 represents the number of required constrained iterations. By properly choosing n_1 and n_2, the algorithm can either coincide with classical unconstrained VF, perform all constrained iterations, or perform a given number of constrained iterations after classical VF is applied.

We remark that, as for the classical unconstrained case, different criteria can be pursued to break the two iteration cycles before the iteration number hits n_1 or n_2. In general, such criteria are of the form

$$R^{(\mu)} \leq \epsilon \tag{31}$$

where ϵ is a user-defined positive convergence threshold while $R^{(\mu)}$ is a convergence index. Different choices have been proposed in the literature to define $R^{(\mu)}$, e.g., it can represent the norm of the denominator coefficients $\tilde{\mathbf{C}}^{(\mu)}$, or the improvement of the model against data error between two successive iterations. Further discussion about the most proper choice of the definition of $R^{(\mu)}$ is outside the scope of this work. For illustration purposes, here we will assume $R^{(\mu)} = \|\tilde{\mathbf{C}}^{(\mu)}\|_2$.

In this article, we consider two particular ways of choosing n_1 and n_2, namely

1. $n_1 = 0$ and $n_2 > 0$: in this case, all the VF iterations are constrained, and the ROM is guaranteed $\mathcal{D}$-stable at each iteration.
2. $n_1 > 0$ and $n_2 = 1$: in this case, only the last iteration is constrained.

Choice 1 is the most natural modification of VF when implementing the proposed stability constraints, but it requires the repeated application of semidefinite programming, which can be computationally intensive for large values of n and which, although numerically robust, is still less reliable than the simpler standard algebraic techniques

Algorithm 1 VF with $\mathcal{D}$-stability

Input: ROM order n. Maximum number of unconstrained iterations n_1. Maximum number of constrained iterations n_2. Matrices $\mathbf{L}$, $\mathbf{M}$. Data $\{s_k\}$, $\{\hat{H}_k\}$. Convergence threshold ϵ
Output: ROM $H(s)$
Initialize $\{q_i^0\}$
for $\mu = 1, 2, \ldots, n_1$ **do**
  Solve problem (9)
  Compute $\{q_i^{\mu+1}\}$ according to (15).
  if $R^{(\mu)} \leq \epsilon$ **then**
    break
  end if
end for
$\mu_1 \leftarrow \mu$
for $\mu = \mu_1 + 1, \mu_1 + 2, \ldots, \mu_1 + n_2$ **do**
  Solve problem (27)
  Compute $\{q_i^{\mu+1}\}$ according to (15).
  if $R^{(\mu)} \leq \epsilon$ **then**
    break
  end if
end for
$\bar{\mu} \leftarrow \mu$
set $p_i \leftarrow q_i^{\bar{\mu}}$
compute r_i and r_∞ of $H(s)$ in (6) via least-squares.
return: $H(s)$

required by standard VF iteration. We empirically observed that choice 2 is generally both effective and more efficient than choice 1 when only few elements of $\{q_i^{\mu_1}\}$ are outside the desired $\mathcal{D}$-region.

As a concluding remark, we highlight that, contrarily to other available strategies that perform model perturbation to place the ROM poles in specified regions of the complex plane (see, e.g., [26] for the case of the left-half plane), the proposed approach explicitly optimizes the model coefficients to minimize the residual error against the available data. To the best of the authors' knowledge, this feature, combined with the enforcement of $\mathcal{D}$-stability, is not granted by any available DDMOR technique. Also, since $\mathcal{D}$-regions are by definition described in terms of LMIs, any approach aimed at enforcing concurrently accuracy and $\mathcal{D}$-stability must inevitably rely on semidefinite programming.

5 Applications

In this section, we provide a collection of practical applications enabled by the proposed approach. All the experiments are performed in MATLAB using a workstation equipped with 32 GB of memory and a 3.3 GHz Intel i9-X7900 CPU. The semidefinite optimization is performed using YALMIP [41] in combination with the MOSEK solver [4]. In our experiments, we define the point-wise residual error $E(s_k)$ and the

root mean square (RMS) error between the model and the available data as

$$E(s_k) = \hat{H}_k - H(s_k), \quad E_{\text{RMS}} = \sqrt{\frac{1}{K}\sum_{k=1}^{K} |E(s_k)|^2}, \tag{32}$$

5.1 Poles within a polytopic region

We first consider an academic example, namely the Atmospheric Storm Tracker [46] first presented in [25]. The FOM for this example is provided in terms of a state-space of size $N = 598$. For the sake of demonstration, we assume that the knowledge of such state-space description is limited to the matrix $\hat{\mathbf{A}}$ and to $K = 1500$ samples of its transfer function retrieved over a logarithmic grid on the imaginary axis within the interval $[j2\pi f_{\min}, j2\pi f_{\max}]$, with $f_{\min} = 0.01$ Hz and $f_{\max} = 10$ Hz. We normalize the available sampling locations with respect to $f_{\max}$ so that the standardized sampling points are $\{s_k\} \subseteq [j2\pi \frac{f_{\min}}{f_{\max}}, j2\pi]$.

Also, the available samples are normalized to have a maximum magnitude equal to one.

We then compute the spectrum of $\hat{\mathbf{A}}$, and we embed it within the polytopic $\mathcal{D}$ region shown in Fig. 2, after normalizing it by $f_{\max}$. The region is the intersection of a horizontal strip requiring $|\Im\mathfrak{m}\{s\}| < 1.19$, a right-half plane such that $\Re\mathfrak{e}\{s\} > -0.3635$, and a cone with apex at the origin and inner angle 2θ with $\theta = 1.544$ rad. The three regions are characterized respectively by the matrices

$$\mathbf{M}_{\text{str}} = \begin{bmatrix} 0 & \frac{1}{2} \\ -\frac{1}{2} & 0 \end{bmatrix}, \quad \mathbf{L}_{\text{str}} = -1.19 \times \mathbf{I}_2, \quad \text{for the strip,} \tag{33}$$

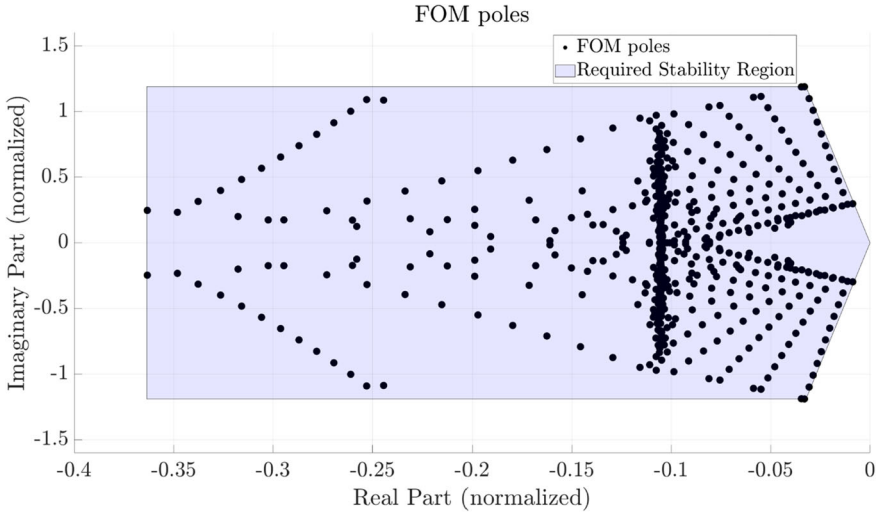


Fig. 2 Poles of the Atmospheric Storm Tracker FOM and the prescribed $\mathcal{D}$ region where we require the ROM poles to lie

$$\mathbf{M}_{\text{rhp}} = -1, \quad \mathbf{L}_{\text{rhp}} = -0.3635 \times 2, \quad \text{for the right-half plane,} \tag{34}$$

$$\mathbf{M}_{\text{cone}} = \begin{bmatrix} \sin\theta & \cos\theta \\ -\cos\theta & \sin\theta \end{bmatrix}, \quad \mathbf{L}_{\text{cone}} = \mathbf{0} \quad \text{for the cone.} \tag{35}$$

The overall LMI characterization for the whole $\mathcal{D}$ region is obtained by block-diagonalizing $\mathbf{M}_{\text{str}}$, $\mathbf{M}_{\text{rhp}}$, $\mathbf{M}_{\text{cone}}$ and $\mathbf{L}_{\text{str}}$, $\mathbf{L}_{\text{rhp}}$, $\mathbf{L}_{\text{cone}}$ to form $\mathbf{M}$ and $\mathbf{L}$ respectively:

$$\mathbf{L} = \text{blkdiag}\{\mathbf{L}_{\text{str}}, \mathbf{L}_{\text{rhp}}, \mathbf{L}_{\text{cone}}\} \qquad \mathbf{M} = \text{blkdiag}\{\mathbf{M}_{\text{str}}, \mathbf{M}_{\text{rhp}}, \mathbf{M}_{\text{cone}}\}. \tag{36}$$

We then build two different ROMs of order $n = 34$, testing the proposed approach with two different strategies. In the first case, we apply Algorithm 1 with $n_1 = 14$ and $n_2 = 1$, in such a way that only the last VF iteration is constrained to return a $\mathcal{D}$-stable model. In the second case, we require all constrained iterations by setting $n_1 = 0$ and $n_2 = 15$. On our hardware, the time requirements of the proposed approach are 3 s and 40 s, respectively.

The RMS error of the two ROMs throughout the iterations is shown in Fig. 3. The results show that an RMS error $E_{\text{con2}} = 2.63 \times 10^{-8}$ can be attained by applying the second approach and retrieving the model obtained at the eleventh iteration. On the other hand, the ROM obtained by constraining only the last iteration commits a slightly larger error, with $E_{\text{con1}} = 3.94 \times 10^{-8}$. Such a small error difference is, however, insignificant in any realistic application of the ROMs. Hence, for this test case, the two strategies are almost equivalent, with the difference that the first is more efficient from a computational standpoint, as it requires solving the semidefinite optimization only once. In general, when data are not affected by noise (see Section 5.2), we suggest to apply the second strategy only in case the accuracy granted by the first is unsatisfactory.

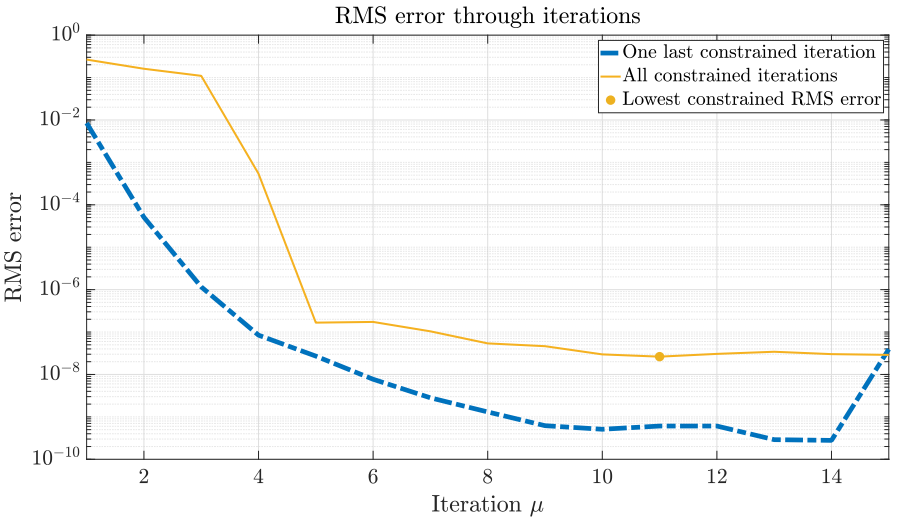


Fig. 3 The RMS error committed at VF iterations by two models estimated enforcing the required $\mathcal{D}$-stability constraints at each iteration (blue dashed line) and at the last iteration only (solid yellow line). The yellow dot represents the minimum error committed by a $\mathcal{D}$-stable model

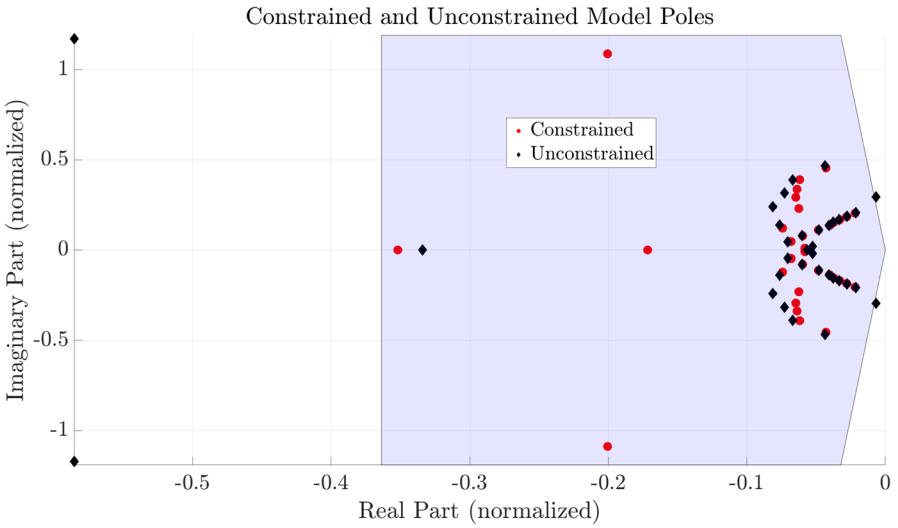


Fig. 4 Constrained and unconstrained poles of the two ROMs of the Atmospheric Storm Tracker. Application of the proposed approach guarantees the fulfillment of the required $\mathcal{D}$-stability requirement, removing four non-compliant poles identified by plain VF

The poles found after 14 unconstrained iterations and those found after 11 constrained iterations are shown in Fig. 4. Four non-compliant poles returned by the standard VF iteration are effectively relocated to the required $\mathcal{D}$-stability region by the proposed method.

Figure 5 shows that the relocation of such poles comes with no significant degradation in terms of model accuracy. Both the constrained and the unconstrained models are practically indistinguishable from the available data. Numerically, the unconstrained models commit a root mean square (RMS) error equal to $E_{\text{uncon}} = 2.8 \times 10^{-10}$, while the $\mathcal{D}$-stable model RMS error amounts to $E_{\text{con}} = 2.63 \times 10^{-8}$.

5.2 Fitting of noisy data

We consider the problem of fitting noisy samples of a Bounded Real (BR) transfer function. This is a common modeling problem in electrical and electronic engineering, as usually physically passive devices for high-frequency applications are characterized via measurements of their scattering parameters, that are known to be BR functions [1].

In this section, we model a transfer function expressed as the reflection coefficient of a transmission line network, shown in Fig. 6 (black solid line). The data were computed using an in-house Matlab script embedding both a quasi-static 2D field solver to determine the transmission line parameters, and the frequency-domain solution of the distributed network in the bandwidth [50] MHz. A number of $K = 996$ samples of the reflection coefficient, logarithmically distributed in the considered bandwidth, are used to drive the modeling process. The sampling locations are then normalized as in the previous section, considering $f_{\max} = 1000$ MHz. An additive noise $\nu_k =$

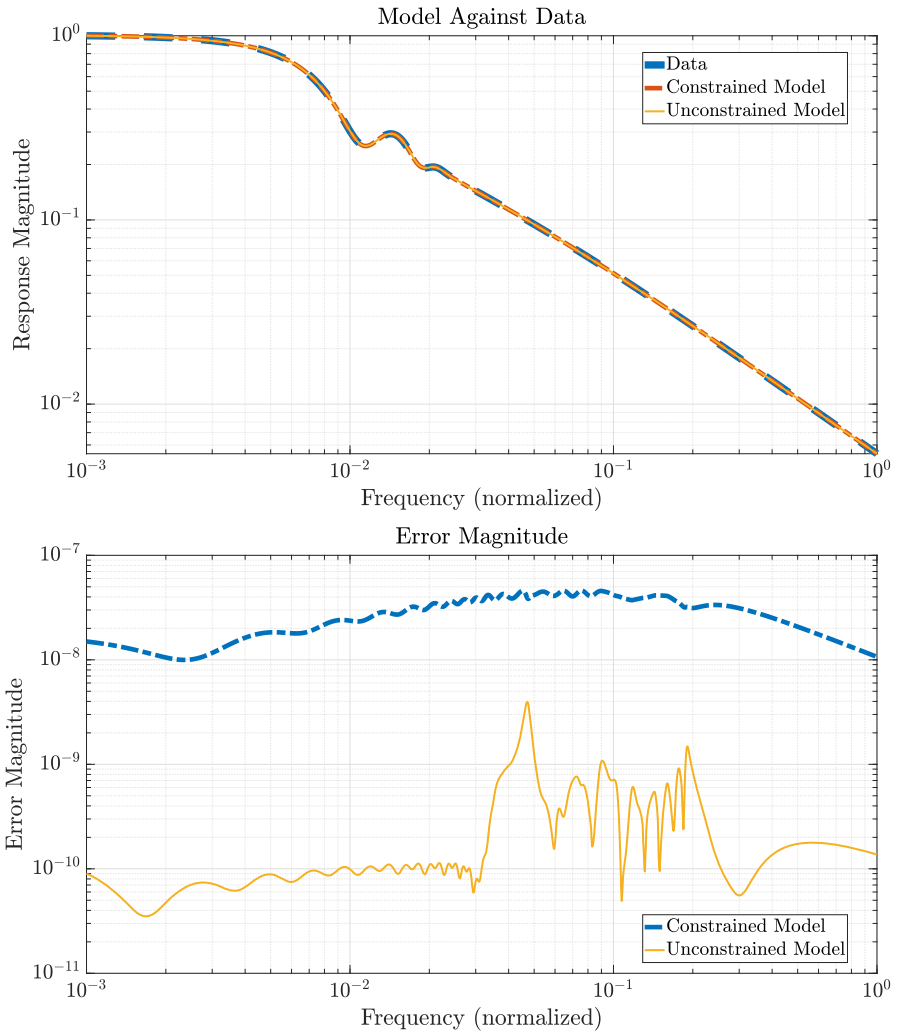


Fig. 5 Atmospheric Storm Tracker example. Top panel: fitting with and without the proposed $\mathcal{D}$-stability constraints. Bottom panel: error magnitude plot

$\nu_{\mathrm{Re},k} + j\,\nu_{\mathrm{Im},k}$ is added to each transfer function sample $\hat{H}_k$, where both $\nu_{\mathrm{Re},k}$ and $\nu_{\mathrm{Im},k}$ are random variables extracted from a zero mean gaussian distribution. The variance of the vector $\boldsymbol{\nu} = [\nu_1, \nu_2, \ldots \nu_{996}]$ is approximately 0.005.

The resulting noise-corrupted magnitude of the reflection coefficient is shown in Fig. 6 (blue line).

Applying the plain VF algorithm on the noisy dataset, setting the model order to $n = 22$, returns the model shown in Fig. 7 (yellow line) after 10 iterations. The quality of the approximation is severely impaired by the presence of noise, that leads to the identification of spurious undamped poles.

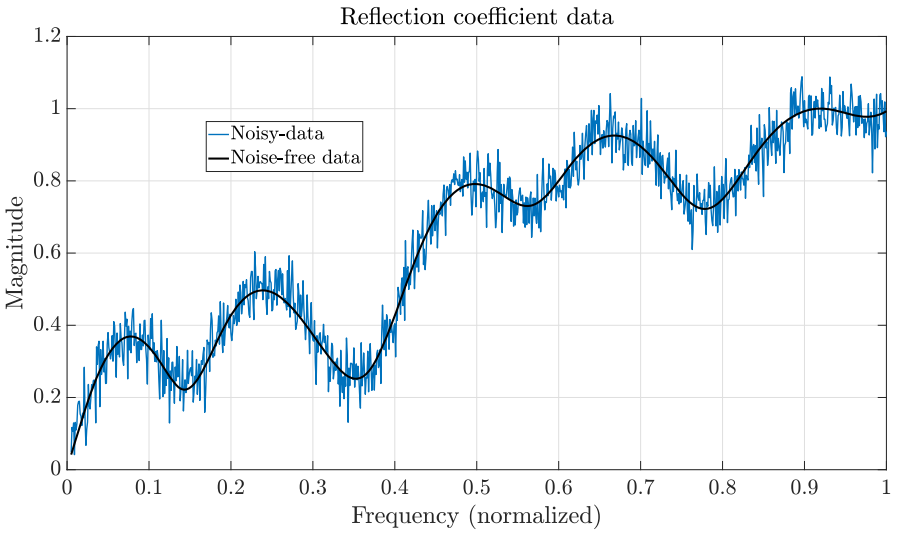


Fig. 6 Actual and noisy samples of a transfer function representing the reflection coefficient of a transmission line network

We apply the proposed approach to constrain the model poles in the left-half complex plane characterized by $\mathfrak{Re}\{s\} < -\alpha < 0$, with the aim of removing the fake model resonances. This is done by setting $\mathbf{L} = 2\alpha$ and $\mathbf{M} = 1$ in the constraints of problem (27), which defines the $\mathcal{D}$-region as

$$\mathcal{D} = \{s \in \mathbb{C} : 2\alpha + s + s^* = 2\sigma + 2\alpha < 0\}, \tag{37}$$

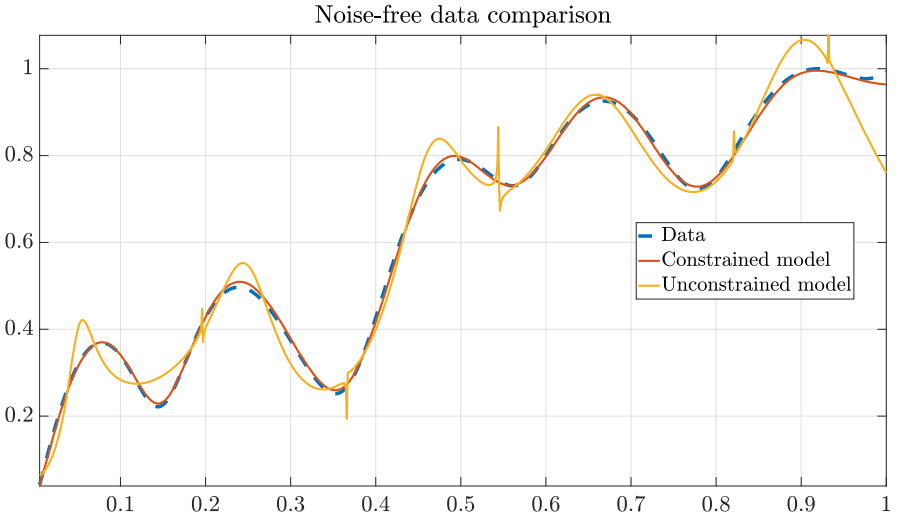


Fig. 7 Unconstrained (yellow) and constrained (red) VF models obtained by fitting noisy data, compared to the reference (noise-free) response (blue). The constraint on the real part of the model poles effectively removes the spurious resonances induced by the presence of noise

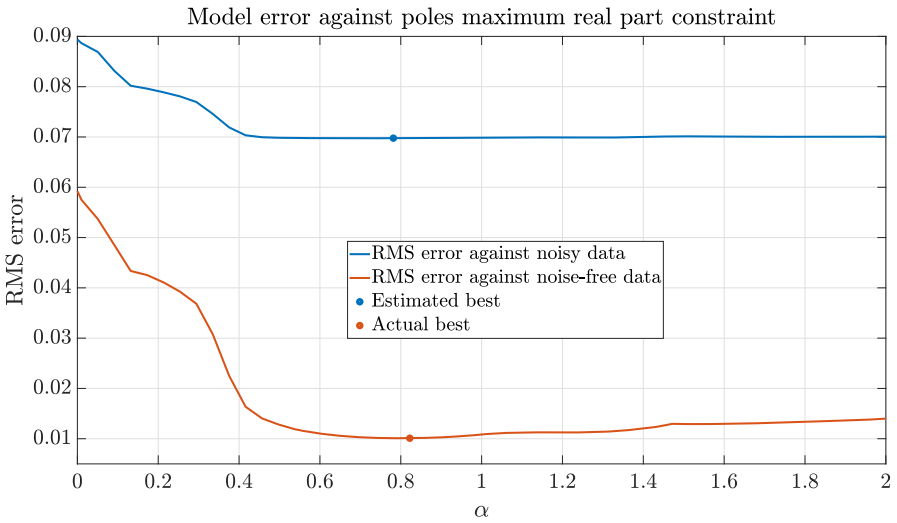


Fig. 8 RMS error committed by different VF models against the maximum value of the real part allowed during the pole relocation stage

and applying Algorithm 1 with $n_1 = 0$, $n_2 = 10$, and $\sigma = 10^{-6}$ for 50 different values of α logarithmically distributed in the interval [0, 1.2]. The most suitable value of α is found by computing the RMS error committed by the resulting models against the noisy data.

Figure 8 shows the results of these experiments. The best RMS error value is attained for $\alpha = 0.781$. Such value is in good agreement with the optimum, computed by computing again the same error metric using the noise-free data (that would not be available in a real measurement-driven scenario). The magnitude of the model obtained with $\alpha = 0.781$ is shown in Fig. 7 (dashed-blue line). Applying the proposed constrained fitting process performs significantly better than the unconstrained counterpart and successfully removes the ROM nonphysical resonances.

We also highlight that, although this result may not be general, applying the proposed approach also improves the convergence performance of the VF iteration in the presence of noise.

Indeed, recalling that one convergence criterion for VF is to have $\|\tilde{\mathbf{C}}^{(\mu)}\|_2$ or, equivalently, $D^{(\mu)}(s) = 1$, Fig. 9 shows that the norm of the denominator coefficients $\tilde{\mathbf{C}}^{(\mu)}$ monotonically decreases in the constrained case while it stagnates otherwise (50 iterations are considered here for illustration purposes only).

5.3 Out-of-band pole rejection

This last application concerns the generation of a stable ROM having poles inside the left-half closed disk of radius r centered at the origin. The system under modeling is a high-speed printed circuit board (PCB) interconnect introduced in [49]. Its transmission coefficient is sampled over $K = 499$ distinct locations logarithmically distributed

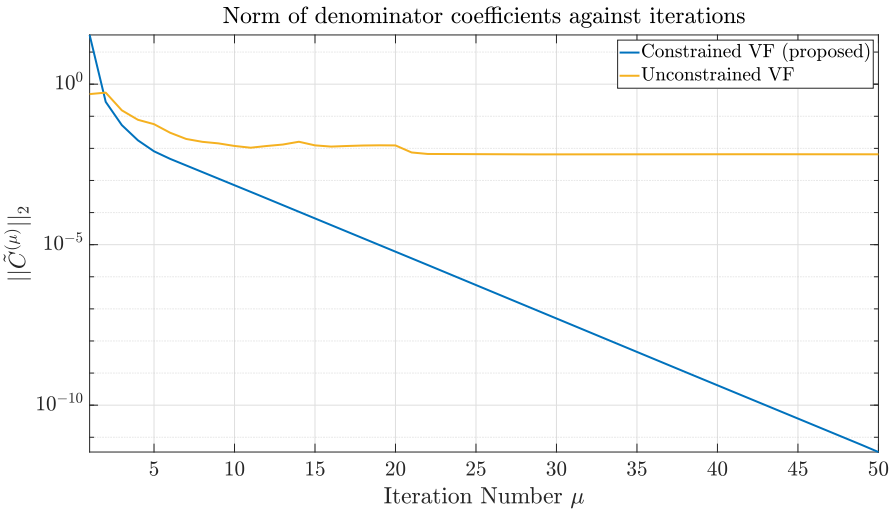


Fig. 9 Convergence of the VF iteration in terms of the norm of the denominator residues vector $\tilde{\mathbf{C}}^{(\mu)}$

over the imaginary axis, in the interval $[j2\pi f_{\min}, j2\pi f_{\max}]$, with $f_{\min} = 10$ MHz and $f_{\max} = 10$ GHz. The same normalization approach applied for Example 5.1 was also applied here, so that the standardized sampling points are $\{s_k\} \subseteq [j2\pi \frac{f_{\min}}{f_{\max}}, j2\pi]$. We set $r = 2\pi$ to place the ROMs poles over the frequency range where data are available, i.e., in such a way that there are no out-of-band poles. See [32] for practical implications of this ROM property.

The required $\mathcal{D}$ region is the intersection of two sets, the left-half complex plane for which

$$\mathbf{L}_{\text{LH}} = 0, \quad \mathbf{M}_{\text{LH}} = 1; \tag{38}$$

and the disk of radius 2π centered at the origin, defined by an LMI with [22]

$$\mathbf{L}_{\text{D}} = \begin{bmatrix} -2\pi & 0 \\ 0 & -2\pi \end{bmatrix} \quad \mathbf{M}_{\text{D}} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}. \tag{39}$$

Hence we have

$$\mathbf{L} = \text{blkdiag}\{\mathbf{L}_{\text{LH}}, \mathbf{L}_{\text{D}}\} \qquad \mathbf{M} = \text{blkdiag}\{\mathbf{M}_{\text{LH}}, \mathbf{M}_{\text{D}}\} \tag{40}$$

The matrices (40) are then used to define the constraints in (29). For this application, we apply 10 iterations of the standard unconstrained VF scheme with $n = 46$, and we add the required $\mathcal{D}$-stability constraint during an additional (last) iteration. In order to check the sensitivity of the ROM accuracy on the positivity threshold σ in (29), we built 200 different models using 200 logarithmically spaced values of σ in the interval $[10^{-10}, 10^{-6}]$.

Figure 10 shows the RMS error of these models against the employed σ. The average error computed over the experiments is 6.76×10^{-3} while its standard deviation is

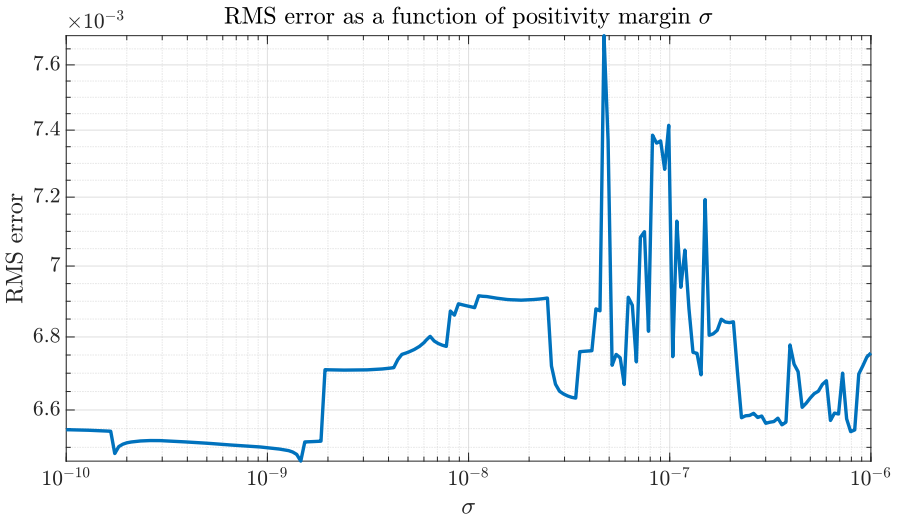


Fig. 10 RMS error of the ROM with only in-band poles against the positivity threshold σ used to solve (29)

2.07×10^{-4}, confirming that from the applications standpoint, the variability of the ROM accuracy with respect to σ is negligible.

Figure 11 shows the poles of the resulting constrained and unconstrained models. The procedure effectively removes the out-of-band poles and relocates them within the required half disk.

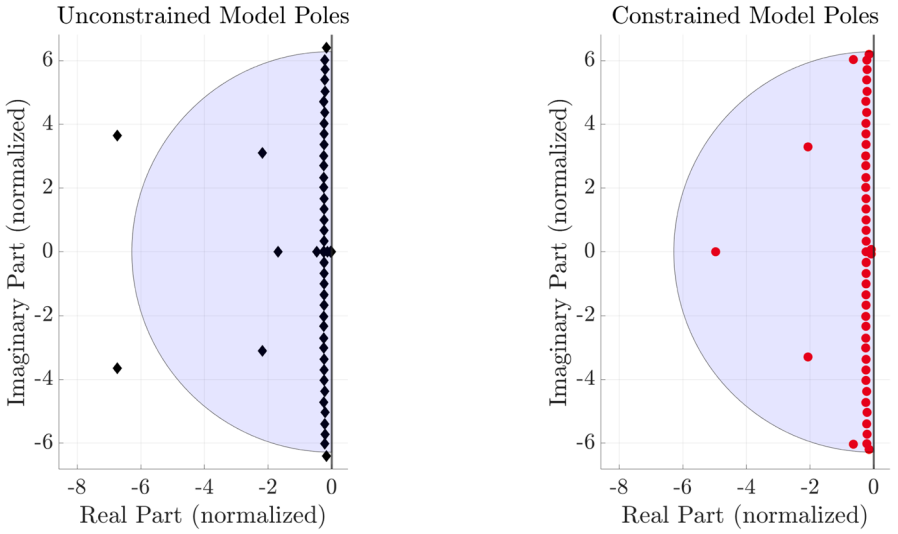


Fig. 11 Unconstrained poles for a ROM of a transmission line reflection coefficient (left) and the poles obtained by forcing them to lie over the intersection between the left-half complex plane and the disk of radius 2π centered at the origin (right)

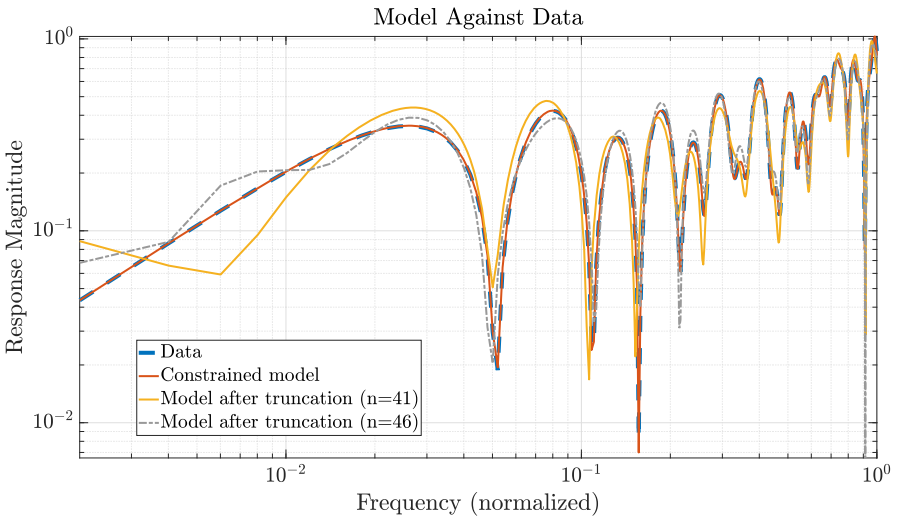


Fig. 12 The ROM obtained by forcing the poles to lie within the stable unit disk of radius 2π (red) against the available data (blue). In yellow and gray: two models obtained applying plain VF, discarding the poles falling outside the required $\mathcal{D}$-region and finally optimizing for the residues of the remaining poles (respectively of order 41 and 46 after truncation)

Figure 12 shows the agreement between the magnitude of the $\mathcal{D}$-stable model against the available data. In the same figure, we also report the magnitude of another ROM resulting from the truncation of the 5 unwanted poles, i.e., the poles outside the half-disk, and the subsequent least-squares optimization of the model residues. To

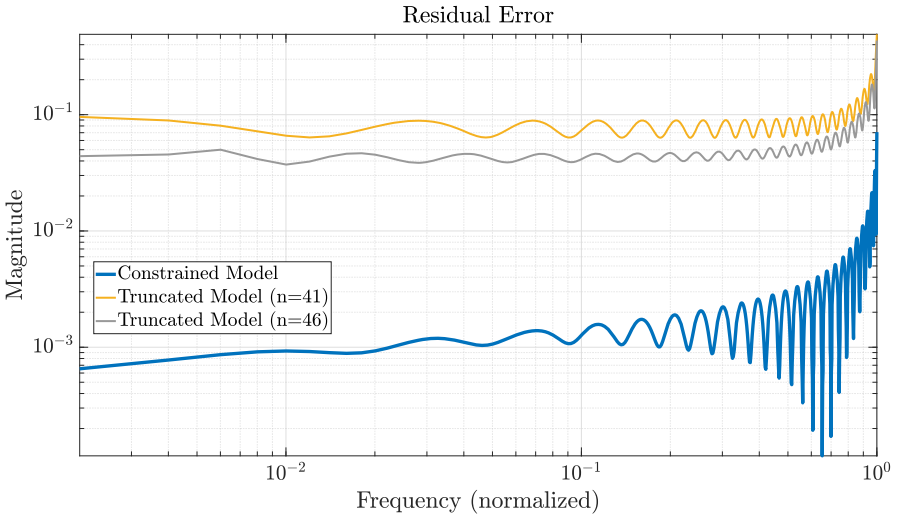


Fig. 13 Error magnitude of three models. Blue: the error of the proposed ROM with poles within the desired $\mathcal{D}$-region. Yellow and gray: two models obtained via standard VF after removing the poles outside the $\mathcal{D}$-region (of order $n = 41$ and $n = 46$ respectively)

perform a fair comparison, we also applied standard VF with an increasing number of poles, until, after truncation, the actual unconstrained model order is $n = 46$, the same as the one obtained by applying the proposed approach. Also, pursuing this strategy, the resulting model is less accurate than the constrained one. These results confirm that the proposed method provides superior accuracy compared to this naive approach.

The magnitude of the residual error defined as in (32) for the three models is reported in Fig. 13.

6 Conclusions

This paper presented a novel approach to perform rational fitting endowed with $\mathcal{D}$-stability constraints. The method allows placing the poles of the resulting approximant within arbitrary convex regions that are symmetric with respect to the real axis. The goal is achieved by solving a convex semidefinite program resulting from a relaxation of the exact $\mathcal{D}$-stability constraints expressed in terms of the rational barycentric model structure. Three testcases were used to demonstrate the effectiveness of the proposed constrained rational fitting algorithm, showing that $\mathcal{D}$-stability can be quite useful in practical engineering applications.

Acknowledgements The authors thank Dr. Ion Victor Gosea for the fruitful discussions about recent state-of-the-art advances.

Author contribution Tommaso Bradde: formulation, software implementation, numerical experiments, writing. Stefano Grivet-Talocia: numerical experiments, writing.

Funding Open access funding provided by Politecnico di Torino within the CRUI-CARE Agreement. No funding was received for conducting this study.

Code and data availability The authors declare that data and code will be made available upon reasonable request.

Declarations

Conflict of interest The authors declare no competing interests.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

1. Anderson, B., Vongpanitlerd, S.: Network Analysis and Synthesis: A Modern Systems Theory Approach, 2nd edn. Dover Publications (2006)

2. Antoulas, A.C., Anderson, B.D.O.: On the scalar rational interpolation problem. IMA J. Math. Control. Inf. **3**(2–3), 61–88 (1986). https://doi.org/10.1093/imamci/3.2-3.61
3. Antoulas, A.C., Beattie, C.A., Gugercin, S.: Interpolatory Methods for Model Reduction. Computational Science & Engineering. Society for Industrial and Applied Mathematics, Philadelphia, PA (2020). https://doi.org/10.1137/1.9781611976083
4. ApS, M.: The MOSEK optimization toolbox for MATLAB manual. Version 10.1.16. (2023). https://docs.mosek.com/latest/toolbox/index.html
5. Aumann, Q., Gosea, I.V.: Practical challenges in data-driven interpolation: dealing with noise, enforcing stability, and computing realizations. International Journal of Adaptive Control and Signal Processing (2023). https://onlinelibrary.wiley.com/doi/10.1002/acs.3691?af=R. Online Version of Record before inclusion in an issue
6. Benner, P., Grivet-Talocia, S., Quarteroni, A., Rozza, G., Schilders, W., Silveira, L.M. (eds.): Model Order Reduction, vol. 1: System- and Data-Driven Methods and Algorithms. De Gruyter, Berlin, Boston (2021). https://doi.org/10.1515/9783110498967
7. Berljafa, M., Güttel, S.: The rkfit algorithm for nonlinear rational approximation. SIAM J. Sci. Comput. **39**(5), A2049–A2071 (2017). https://doi.org/10.1137/15M1025426
8. Borghi, A., Breiten, T.: $\mathcal{H}_2$ optimal rational approximation on general domains. Adv. Comput. Math. **50**(3), 28 (2024)
9. Borghi, A., Breiten, T., S., G.: Balanced truncation with conformal maps. Systems & Control Letters 197, 106044 (2025). https://doi.org/10.1016/j.sysconle.2025.106044. https://www.sciencedirect.com/science/article/pii/S016769112500026X
10. Bradde, T., Chevalier, S., De Stefano, M., Grivet-Talocia, S., Daniel, L.: Handling initial conditions in vector fitting for real time modeling of power system dynamics. Energies **14**(9), (2021). https://doi.org/10.3390/en14092471
11. Bradde, T., Grivet-Talocia, S., Aumann, Q., Gosea, I.V.: A modified aaa algorithm for learning stable reduced-order models from data. J. Sci. Comput. **103**(1), 14 (2025). https://doi.org/10.1007/s10915-025-02825-0
12. Bradde, T., Grivet-Talocia, S., Zanco, A., Calafiore, G.C.: Data-driven extraction of uniformly stable and passive parameterized macromodels. IEEE Access **10**, 15786–15804 (2022). https://doi.org/10.1109/ACCESS.2022.3147034
13. Brunton, S.L., Kutz, J.N.: Data-driven methods for reduced-order modeling. In: P. Benner, S. Grivet-Talocia, A. Quarteroni, G. Rozza, W. Schilders, L.M. Silveira (eds.) Model Order Reduction - Volume 2 Snapshot-Based Methods and Algorithms, pp. 307–344. De Gruyter, Berlin, Boston (2021). https://doi.org/10.1515/9783110671490-007
14. Carlucci, A., Bradde, T., Grivet-Talocia, S.: Addressing load sensitivity of rational macromodels. IEEE Trans. Compon. Packag. Manuf. Technol. **13**(10), 1591–1602 (2023). https://doi.org/10.1109/TCPMT.2023.3284551
15. Carlucci, A., Gosea, I.V., Grivet-Talocia, S.: Approximation of generalized frequency response functions via vector fitting. In: Mathematical Optimization for Machine Learning: Proceedings of the MATH+ Thematic Einstein Semester 2023, pp. 169–180. De Gruyter (2025)
16. Carlucci, A., Gosea, I.V., Grivet-Talocia, S.: Data-driven modeling of weakly nonlinear circuits via generalized transfer function approximation. IEEE Access **13**, 2746–2762 (2025). https://doi.org/10.1109/ACCESS.2024.3520388
17. Carrera-Retana, L.E., Marin-Sanchez, M., Schuster, C., Rimolo-Donadio, R.: Improving accuracy after stability enforcement in the Loewner matrix framework. IEEE Trans. Microw. Theory Tech. **70**(2), 1037–1047 (2021)
18. Cherifi, K., Goyal, P., Benner, P.: A greedy data collection scheme for linear dynamical systems. Data-Centric Eng. **3**, (2022). https://doi.org/10.1017/dce.2022.16
19. Chilali, M., Gahinet, P.: $\mathcal{H}_\infty$ design with pole placement constraints: an LMI approach. IEEE Trans. Autom. Control **41**(3), 358–367 (1996). https://doi.org/10.1109/9.486637
20. Chilali, M., Gahinet, P., Apkarian, P.: Robust pole placement in LMI regions. IEEE Trans. Autom. Control **44**(12), 2257–2270 (1999). https://doi.org/10.1109/9.811208
21. Chinea, A., Grivet-Talocia, S.: On the parallelization of vector fitting algorithms. IEEE Trans. Compon. Packag. Manuf. Technol. **1**(11), 1761–1773 (2011). https://doi.org/10.1109/TCPMT.2011.2167973
22. Choudhary, N., Gillis, N., Sharma, P.: Characterizing matrices with eigenvalues in an LMI region: a dissipative-hamiltonian approach. Linear Multilin. Algebra **72**(17), 2984–2999 (2024). https://doi.org/10.1080/03081087.2024.2304144

23. Davis, L., Johns, W., Monzon, L., Reynolds, M.: Iterative stability enforcement in adaptive antoulas-anderson algorithms for H_2 model reduction. SIAM J. Scientif. Comput. **45**(4), (2023). https://doi.org/10.1137/21M1467043. https://www.osti.gov/biblio/2005602
24. Deschrijver, D., Mrozowski, M., Dhaene, T., De Zutter, D.: Macromodeling of multiport systems using a fast implementation of the vector fitting method. Microw. Wirel. Compon. Lett. IEEE **18**(6), 383–385 (2008). https://doi.org/10.1109/LMWC.2008.922585
25. Farrell, B., Ioannou, P.: Stochastic dynamics of the midlatitude atmospheric jet. J. Atmos. Sci. **52**(10), 1642–1656 (1995). https://doi.org/10.1175/1520-0469(1995)052<1642:SDOTMA>2.0.CO;2
26. Gosea, I.V., Antoulas, A.C.: Stability preserving post-processing methods applied in the Loewner framework. In: IEEE 20th Workshop on Signal and Power Integrity (SPI), Turin, Italy, May 8–11, pp. 1–4 (2016). https://doi.org/10.1109/SaPIW.2016.7496283
27. Gosea, I.V., Poussot-Vassal, C., Antoulas, A.C.: On enforcing stability for data-driven reduced-order models. In: 29th Mediterranean Conference on Control and Automation (MED), Virtual, pp. 487–493 (2021). https://doi.org/10.1109/MED51440.2021.9480216
28. Goyal, P., Peherstorfer, B., Benner, P.: Rank-minimizing and structured model inference. SIAM J. Sci. Comput. **46**(3), A1879–A1902 (2024). https://doi.org/10.1137/23M1554308
29. Goyal, P., Pontes Duff, I., Benner, P.: Guaranteed stable quadratic models and their applications in SINDy and operator inference. arXiv preprint (2023). https://doi.org/10.48550/arXiv.2308.13819
30. Goyal, P., Pontes Duff, I., Benner, P.: Inference of continuous linear systems from data with guaranteed stability. arXiv preprint (2023). https://doi.org/10.48550/arXiv.2301.10060
31. Grivet-Talocia, S., Gustavsen, B.: Passive Macromodeling: Theory and Applications. John Wiley & Sons (2015). ISBN: 978-1-118-09491-4
32. Grivet-Talocia, S., Ubolli, A.: On the generation of large passive macromodels for complex interconnect structures. IEEE Trans. Adv. Packag. **29**(1), 39–54 (2006). https://doi.org/10.1109/TADVP.2005.862659
33. Gugercin, S., Antoulas, A.C., Beattie, C.: $\mathcal{H}_2$ model reduction for large-scale linear dynamical systems. SIAM J. Matrix Anal. Appl. **30**(2), 609–638 (2008)
34. Gustavsen, B., Semlyen, A.: Rational approximation of frequency domain responses by vector fitting. IEEE Trans. Power Delivery **14**(3), 1052–1061 (1999). https://doi.org/10.1109/61.772353
35. Hund, M., Mitchell, T., Mlinarić, P., Saak, J.: Optimization-based parametric model order reduction via $\mathcal{H}_2 \otimes \mathcal{L}_2$ first-order necessary conditions. SIAM J. Sci. Comput. **44**(3), A1554–A1578 (2022)
36. Ionutiu, R., Rommes, J., Antoulas, A.C.: Passivity preserving model reduction using dominant spectral zero interpolation. IEEE Trans. Comput.-Aided Design Integr. Circuits Syst. **27**(12), 2250–2263 (2008). https://doi.org/10.1109/TCAD.2008.2006160
37. I.V. Gosea, C.P.V., Antoulas, A.: On loewner data-driven control for infinite-dimensional systems. In: 2021 European Control Conference (ECC), pp. 93–99 (2021). https://doi.org/10.23919/ECC54610.2021.9655097
38. Kergus, P.: Data-driven stability analysis and enforcement for Loewner data-driven control. In: Workshop Intersections between Learning, Control and Optimization (IPAM, UCLA) (2020). https://hal.science/hal-03095975/document
39. Kushel, O.Y.: Geometric properties of LMI regions (2019). arxiv:1910.10372
40. Lefteriu, S., Antoulas, A.C.: On the convergence of the vector-fitting algorithm. Microw. Theory Techniq. IEEE Trans. **61**(4), 1435–1443 (2013). https://doi.org/10.1109/TMTT.2013.2246526
41. Löfberg, J.: Yalmip : A toolbox for modeling and optimization in matlab. In: In Proceedings of the CACSD Conference. Taipei, Taiwan (2004). https://doi.org/10.1109/CACSD.2004.1393890
42. Mayo, A.J., Antoulas, A.C.: A framework for the solution of the generalized realization problem. Linear Algebra Appl. **425**(2–3), 634–662 (2007). https://doi.org/10.1016/j.laa.2007.03.008
43. Mlinarić, P., Gugercin, S.: $\mathcal{L}_2$-optimal reduced-order modeling using parameter-separable forms. SIAM J. Sci. Comput. **45**(2), A554–A578 (2023)
44. Nakatsukasa, Y., Sete, O., Trefethen, L.N.: The AAA algorithm for rational approximation. SIAM J. Sci. Comput. **40**(3), A1494–A1522 (2018). https://doi.org/10.1137/16M1106122
45. Nesterov, Y., Nemirovskii, A.: Interior-point polynomial algorithms in convex programming. Soc. Indust. Appl. Math. (1994). https://doi.org/10.1137/1.9781611970791. https://epubs.siam.org/doi/abs/10.1137/1.9781611970791
46. Niconet e.V.: http://www.slicot.org: SLICOT - subroutine library in systems and control theory.

47. Panzer, H., Jaensch, S., Wolf, T., Lohmann, B.: A greedy rational Krylov method for H_2-pseudooptimal model order reduction with preservation of stability. In: Proceedings of the American Control Conference, pp. 5512–5517 (2013). https://doi.org/10.1109/ACC.2013.6580700
48. Pradovera, D., Nobile, F.: A technique for non-intrusive greedy piecewise-rational model reduction of frequency response problems over wide frequency bands. J. Math. Ind. **12**(1), 1–12 (2022). https://doi.org/10.1186/s13362-021-00117-4
49. Preibisch, J.B., Jayaprakash, B., Reuschel, T., Scharff, K., Sen, B., Schuster, C.: Exploring efficient variability-aware analysis method for high-speed digital link design using pce. In: Proc. UBM DesignCon. Santa Clara, CA, USA (2017). https://www.signalintegrityjournal.com/articles/699-designcon-personal-reflections-on-the-past-twenty-years
50. Reis, T., Stykel, T.: Lyapunov balancing for passivity-preserving model reduction of RC circuits. SIAM J. Appl. Dyn. Syst. **10**(1), 1–34 (2011). https://doi.org/10.1137/090779802
51. Reis, T., Willems, J.C.: A balancing approach to the realization of systems with internal passivity and reciprocity. Syst. Control Lett. **60**(1), 69–74 (2011). https://doi.org/10.1016/j.sysconle.2010.10.009
52. Sanathanan, C., Koerner, J.: Transfer function synthesis as a ratio of two complex polynomials. IEEE Trans. Autom. Control **8**(1), 56–58 (1963). https://doi.org/10.1109/TAC.1963.1105517
53. Schwerdtner, P., Voigt, M.: SOBMOR: structured optimization-based model order reduction. SIAM J. Sci. Comput. **45**(2), A502–A529 (2023)
54. Triverio, P.: Vector fitting. In: P. Benner, S. Grivet-Talocia, A. Quarteroni, G. Rozza, W. Schilders, L.M. Silveira (eds.) Model Order Reduction - Volume 1 System- and Data-Driven Methods and Algorithms, pp. 275–310. De Gruyter, Berlin, Boston (2021). https://doi.org/10.1515/9783110498967-008
55. Triverio, P., Grivet-Talocia, S., Nakhla, M.S., Canavero, F.G., Achar, R.: Stability, causality, and passivity in electrical interconnect models. IEEE Trans. Adv. Packag. **30**(4), 795–808 (2007). https://doi.org/10.1109/TADVP.2007.901567
56. Valera-Rivera, A., Engin, A.E.: AAA algorithm for rational transfer function approximation with stable poles. IEEE Lett. Electromagn. Compatibil. Pract. Appl. **3**(3), 92–95 (2021). https://doi.org/10.1109/LEMCPA.2021.3104455
57. Warner, E.C., Scruggs, J.T.: Control of vibratory networks with passive and regenerative systems. In: 2015 American Control Conference (ACC), pp. 5502–5508 (2015). https://doi.org/10.1109/ACC.2015.7172200
58. Benner, P., Grivet-Talocia, S., Quarteroni, A., Rozza, G., Schilders, W., Silveira, L.M. (eds.): Model Order Reduction, vol. 2: Snapshot-Based Methods and Algorithms, De Gruyter, Berlin, Boston (2021). https://doi.org/10.1515/9783110671490
59. Benner, P., Grivet-Talocia, S., Quarteroni, A., Rozza, G., Schilders, W., Silveira, L.M. (eds.): Model Order Reduction, vol. 3: Applications, De Gruyter, Berlin, Boston (2021). https://doi.org/10.1515/9783110499001
60. Benner, P., Grivet-Talocia, S., Quarteroni, A., Rozza, G., Schilders, W., Silveira, L.M. (eds.): Model Order Reduction, vol. 3: Applications, De Gruyter, Berlin, Boston (2021). https://doi.org/10.1515/9783110499001
61. Beattie, C., Gugercin, S.: Realization-independent $\mathcal{H}_2$-approximation. In: 2012 IEEE 51st IEEE Conference on Decision and Control (CDC), Maui, HI, USA, 2012, pp. 4953–4958 (2012). https://doi.org/10.1109/CDC.2012.6426344
62. "IdEM. Electronic Device Characterization", 2026, available: https://www.3ds.com/products/simulia/idem, last access February 19th 2026
63. "Sigrity Broadband SPICE" (2026). available: https://www.cadence.com/en_US/home/tools/ic-package-design-and-analysis/si-pi-analysis-point-tools/sigrity-broadband-spice.html/1000. Access 19 Feb 2026
64. "Vector fitting software" (2026) available: https://www.sintef.no/en/software/vector-fitting/. Access 19 Feb 2026

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.