A New Error Estimator for Reduced-order Modeling of Linear Parametric Systems

Lihong Feng and Peter Benner

***Abstract*—Efficient error estimation is important for reliable reduced-order modeling with guaranteed accuracy. We propose an error estimator for reduced-order modeling of linear parametric dynamical systems. The error estimator estimates the error of the reduced transfer function in frequency domain and can be easily extended to output error estimation of reduced-order models for linear steady parametric systems. It is tight and cheap to compute. Using the error estimator, the reduced-order model can be adaptively obtained with high reliability. Numerical results show that the error estimator can accurately estimate the true error even for transfer functions with many resonances. Compared with an existing error bound, the proposed error estimator can be orders of magnitudes sharper and needs much less computational time.**

***Index Terms*—Adaptive algorithms, estimation error, large-scale systems, reduced-order systems.**

I. Introduction

MODEL ORDER reduction (MOR) has been recognized as an efficient tool for reducing the complexity and sizes of very large-scale complex models arising from many application areas including, but not limited to, circuit simulation, electro-thermal analysis, electromagnetic simulation, and Micro-Electro-Mechanical Systems (MEMS) design. Accurate simulation of such large-scale models is becoming important for reliable estimation of complex phenomena, such as, parameter uncertainty, signal delay etc. This poses high demand for computational resources. The aim of MOR is to enable real-time simulation even in a computational environment with limited resources.

Many MOR methods have been proposed during the last decades and have attracted attention from computational electromagnetics and microwave community [1], [2], [3], [4], [5], [6]. However, efficient and accurate error estimation of the reduced-order model (ROM) is often a critical issue. Although there exists an error bound for the balanced truncation method, this is not applicable to other MOR methods. Many error bounds for balancing related and moment-matching methods [7] are only applicable to non-parametric systems. Moreover, some of them face high computational complexity [8], while others are heuristic [9].

In recent years, numerous MOR methods for linear parametric systems have been developed and found wide applications in engineering, for example, the Krylov based (multi-moment matching) methods [5], [10], [11], [12], the reduced basis methods [6], [13], [2], [3], etc. Many error estimators developed for the reduced basis methods estimate the error for the state vector [14], [6], but are not directly applicable to the output error or transfer function error. Furthermore, those error estimators are developed based on the bilinear weak forms of the systems of Partial Differential Equations (PDEs) [13], [14], which limit their applicability mostly to only finite-element discretized problems, though extensions to finite volume method exist [15].

Recently, an a-posteriori error bound for the transfer function of the ROM has been proposed in [16], being applicable to ROMs constructed by different MOR methods. It is independent of the discretization method (finite difference, finite element, finite volume) applied to the original PDEs and can be directly used in the discretized vector space $\mathbb{C}^n$.

However, one disadvantage of the error bound in [16] is its need to compute the smallest singular value of a matrix with size equal to the degree of the original full-order model, at every parameter sample, leading to high computational cost. A second issue is that for many models from, e.g., circuit simulation, MEMS simulation, the smallest singular value can be zero at some samples of the parameter [3], making the error bound unavailable for those samples. The error bound often overestimates the true error for those systems with the smallest singular value being close to zero, as it appears in the denominator of the error bound.

In this work, we propose a new estimator of the transfer function error, as well as the output error. The proposed error estimator avoids computing the singular values of any matrix, and depends mainly on the ROM. It is applicable to any system whose ROMs are computed using a projection based MOR method. It is illustrated by the numerical results that the error estimator is usually much sharper than the error bound in [16]. Using the proposed error estimation, the adaptive greedy algorithm in [16] converges much faster than using the error bound. It is shown in [9] that for a MIMO system with high-frequency and many resonances in frequency domain, the heuristic adaptive technique proposed there cannot catch all the resonances. However, combing the adaptive greedy algorithm [16] with the proposed error estimator, all the resonances can be reproduced by the ROM.

Compared with our initial work in the conference

Manuscript received Month DD, YYYY; revised Month DD, YYYY; accepted Month DD, YYYY.

This paper is an expanded version from the IEEE MTT-S International Microwave Symposium (IMS 2019), Boston, MA, USA, June 2-7, 2019. (Corresponding author: Lihong Feng.)

Lihong Feng and Peter Benner are with the Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, Germany. E-mail:feng@mpi-magdeburg.mpg.de, benner@mpi-magdeburg.mpg.de.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier XXXXXXXXXXXXX

paper [17], new contributions in this work include: more theoretical analysis on the error estimator is done in Subsection II-A and Subsection II-D; A much simpler proof for the error bound in [16] is proposed in Subsection II-B; Detailed extension of the error estimator to output error estimation for linear steady parametric systems is presented in Subsection II-C; An adaptive algorithm for automatic MOR of parametric linear systems using the new error estimator is proposed in Section III; More simulation results on two new models are given in Section IV.

The paper is organized as follows. In Section II, we propose the error estimator and show how to efficiently compute it. Section III presents greedy algorithms for adaptively computing the ROM using the new error estimator. Numerical results for a CD player model, an interconnect model, a multi-input-multi-output (MIMO) system from circuit simulation and a parametric system from MEMS simulation are presented in Section IV. Conclusions are drawn in the end.

II. The Proposed Error Estimator

We use first-order linear systems in the following form to propose our error estimator, though the error estimator also applies to second-order systems (see the example in Section IV):

$$\begin{aligned} E(\mu)\tfrac{d}{dt}x(t,\mu) &= A(\mu)x(t,\mu)+B(\mu)u(t), \\ y(t,\mu) &= C(\mu)x(t,\mu). \end{aligned} \tag{1}$$

Here, $x(t,\mu)\in\mathbb{R}^n$ is the parameter-dependent state vector. n is often referred to as the *order* of the system. The vector $\mu := (\mu_1,\ldots,\mu_m)\in\mathbb{R}^{1\times m}$ includes all of the geometrical and physical parameters. The system matrices $E(\mu), A(\mu)\in\mathbb{R}^{n\times n}$, and $B(\mu)\in\mathbb{R}^{n\times n_I}$, $C(\mu)\in\mathbb{R}^{n_O\times n}$, $D(\mu)\in\mathbb{R}^{n_O\times n_I}$ may depend on the parameters.

MOR based on projection finds a matrix V whose columns constitute the basis of an approximate subspace of the state space, then replaces the state vector $x(t,\mu)$ by its approximation represented by the linear combination of the columns of V, i.e. $x(t,\mu)\approx Vz(t,\mu)$:

$$\begin{aligned} E(\mu)\tfrac{d}{dt}Vz(t,\mu) &\approx A(\mu)Vz(t,\mu)+B(\mu)u(t), \\ \hat{y}(t,\mu) &= C(\mu)Vz(t,\mu). \end{aligned} \tag{2}$$

Note that the first line in (2) is not an equation anymore, and it induces a residual $e = E(\mu)\frac{d}{dt}Vz(t,\mu) - A(\mu)Vz(t,\mu) + B(\mu)u(t)$ which is nonzero. If we enforce e to be zero in a test subspace spanned by the columns of another matrix W, this is equivalent to letting

$$W^T e = 0,$$

which leads to the first equation of the ROM in (3). This process of deriving the ROM is called MOR based on Petrov-Galerkin projection. Finally, the ROM of the original system can be written as

$$\begin{aligned} \hat{E}(\mu)\tfrac{d}{dt}z(t,\mu) &= \hat{A}(\mu)z(t,\mu)+\hat{B}(\mu)u(t), \\ \hat{y}(t,\mu) &= \hat{C}(\mu)z(t,\mu), \end{aligned} \tag{3}$$

where $\hat{E}(\mu)=W^TEV\in\mathbb{R}^{r\times r}$, $\hat{A}(\mu)=W^TAV\in\mathbb{R}^{r\times r}$, $\hat{B}(\mu)=W^TB\in\mathbb{R}^{r\times n_I}$, $\hat{C}(\mu)=C(\mu)V\in\mathbb{R}^{n_O\times r}$, and $z(t,\mu)\in\mathbb{R}^r$ with $r\ll n$. The state vector $x(t,\mu)$ can be recovered by $x(t,\mu)\approx Vz(t,\mu)$. Different ways of computing W,V result in different MOR methods. For simplicity, we assume Galerkin-projection, i.e. $W=V\in\mathbb{R}^{n\times r}$, though the error estimator immediately applies to ROMs obtained with Petrov-Galerkin projection as well, i.e. $W\neq V$.

The transfer function of the original system is defined as

$$H(\tilde{\mu}) = C(\mu)Q^{-1}(\tilde{\mu})B(\mu),$$

where $Q(\tilde{\mu}) = sE(\mu) - A(\mu)$. Here, s is the Laplace variable in frequency domain, and $\tilde{\mu} := (\mu, s)\in\mathbb{R}^{1\times(m+1)}$. Similarly, the transfer function of the ROM is $\hat{H}(\tilde{\mu}) = \hat{C}(\mu)\hat{Q}^{-1}(\tilde{\mu})\hat{B}(\mu)$, where $\hat{Q}(\tilde{\mu}) = s\hat{E}(\mu) - \hat{A}(\mu)$. Note that the non-singularity of $Q(\tilde{\mu})$ and $\hat{Q}(\tilde{\mu})$ guarantee the existence of $H(\tilde{\mu})$ and $\hat{H}(\tilde{\mu})$, respectively.

We focus on estimating the error of the reduced transfer function $\hat{H}(\tilde{\mu})$, i.e. $|H(\tilde{\mu}) - \hat{H}(\tilde{\mu})|$. As has been discussed in the Introduction, an existing error bound [16] for $|H(\tilde{\mu}) - \hat{H}(\tilde{\mu})|$ needs high computational cost for large-scale systems and is not tight for systems with small $\sigma_{\min}Q(\tilde{\mu})$ (where $\sigma_{\min}$ denotes the minimum singular value of a matrix) . In subsection II-A, we propose a new error estimator which avoids the above difficulties of the existing error bound. Furthermore, we extend the error estimator to output error estimation of steady systems in II-C and discuss its extension to MIMO systems in II-E. We also discuss computing the key components of the error estimator in II-D and fast computation of the error estimator in II-F.

A. The New Error Estimator

In the following, we propose an error estimator for the transfer function $\hat{H}(\tilde{\mu})$. We first consider single-input single-output (SISO) systems. Extension of the error estimator to MIMO systems is considered afterwards. Proofs for Theorems 1-3 are given as Appendices. Define a primal system in frequency domain as

$$Q(\tilde{\mu})x_{pr}(\tilde{\mu}) = B(\mu). \tag{4}$$

The reduced primal system is defined as

$$\hat{Q}(\tilde{\mu})z_{pr}(\tilde{\mu}) = \hat{B}(\mu), \tag{5}$$

so that $\hat{x}_{pr}(\tilde{\mu}) := Vz_{pr}(\tilde{\mu})$ well approximates the solution $x_{pr}(\tilde{\mu})$. The *primal residual* is defined as

$$r_{pr}(\tilde{\mu}) = B(\mu) - Q(\tilde{\mu})\hat{x}_{pr}(\tilde{\mu}).$$

$x_{du}(\tilde{\mu})$ solves the dual system

$$Q^T(\tilde{\mu})x_{du}(\tilde{\mu}) = C^T(\mu), \tag{6}$$

and $z_{du}(\tilde{\mu})$ is the solution to the reduced dual system:

$$\hat{Q}_{du}(\tilde{\mu})z_{du}(\tilde{\mu}) = \hat{C}_{du}(\mu), \tag{7}$$

where $\hat{Q}_{du} = V_{du}^TQ^T(\tilde{\mu})V_{du}$, $\hat{C}_{du} = V_{du}^TC^T(\mu)$, so that $\hat{x}_{du}(\tilde{\mu}) := V_{du}z_{du}(\tilde{\mu})$ well approximates $x_{du}(\tilde{\mu})$. The matrix V_{du} is used to obtain the reduced dual system. The *dual residual* is then defined as $r_{du}(\tilde{\mu}) = C^T(\mu) - Q^T(\tilde{\mu})\hat{x}_{du}(\tilde{\mu})$. Details on how to compute V_{du} are given in subsection II-D.

Theorem 1: The approximation error for the reduced transfer function $\hat{H}(\tilde{\mu})$ can be bounded as

$$|H(\tilde{\mu}) - \hat{H}(\tilde{\mu})| \leq |x_{r_{du}}^T(\tilde{\mu}) r_{pr}(\tilde{\mu})| + |\hat{x}_{du}^T(\tilde{\mu}) r_{pr}(\tilde{\mu})|,$$

where $|\cdot|$ denotes the absolute value of a scalar, $\hat{x}_{du}(\tilde{\mu})$ is defined by (7). $x_{r_{du}}(\tilde{\mu})$ is the solution to the dual-residual system defined as

$$Q^T(\tilde{\mu}) x_{r_{du}}(\tilde{\mu}) = r_{du}(\tilde{\mu}). \tag{8}$$

Note that computing $x_{r_{du}}(\tilde{\mu})$ in (8) needs solving a large system. Instead, we compute the ROM of (8)

$$\tilde{Q}(\tilde{\mu}) z_{r_{du}} = \tilde{r}_{du}(\tilde{\mu}), \tag{9}$$

where $\tilde{Q}(\tilde{\mu}) = V_{r_{du}}^T Q^T(\tilde{\mu}) V_{r_{du}}$, $\tilde{r}_{du}(\tilde{\mu}) = V_{r_{du}}^T r_{du}(\tilde{\mu})$. Then $x_{r_{du}}(\tilde{\mu}) \approx \hat{x}_{r_{du}}(\tilde{\mu}) := V_{r_{du}} z_{r_{du}}(\tilde{\mu})$. Replacing $x_{r_{du}}(\tilde{\mu})$ in the error bound with $\hat{x}_{r_{du}}(\tilde{\mu})$, we get the error estimator:

$$|H(\tilde{\mu}) - \hat{H}(\tilde{\mu})| \lesssim |\hat{x}_{r_{du}}^T(\tilde{\mu}) r_{pr}(\tilde{\mu})| + |\hat{x}_{du}^T(\tilde{\mu}) r_{pr}(\tilde{\mu})| =: \Delta(\tilde{\mu})$$

If $V_{r_{du}} = V_{du}$, then we can prove that the first part of $\Delta(\tilde{\mu})$, i.e. $|\hat{x}_{r_{du}}^T(\tilde{\mu}) r_{pr}(\tilde{\mu})| = 0$.

Theorem 2: If $V_{r_{du}} = V_{du}$, then $|\hat{x}_{r_{du}}^T(\tilde{\mu}) r_{pr}(\tilde{\mu})| = 0$.

Theorem 2 implies that if $V_{r_{du}} = V_{du}$, the fist part of $\Delta(\tilde{\mu})$ will not contribute to the error estimator, and may lead to an estimator underestimating the true error too much. This will be illustrated by the numerical examples in Section IV.

B. A More Concise Proof of the Error Bound in [16]

An error bound for $\hat{H}(\tilde{\mu})$ is proposed in [16], which we recall in the following theorem.

Theorem 3: [16] The approximation error for the reduced transfer function $\hat{H}(\tilde{\mu})$ is bounded by

$$|H(\tilde{\mu}) - \hat{H}(\tilde{\mu})| \leq \Delta_0(\tilde{\mu}) := \\ \|r_{du}(\tilde{\mu})\|_2 \|r_{pr}(\tilde{\mu})\|_2 / \sigma_{\min}(G(\tilde{\mu})) + \|\hat{x}_{du}^T(\tilde{\mu}) r_{pr}(\tilde{\mu})\|_2$$

It is worth pointing out that from (32), we can derive the above error bound in a much simpler way as shown Appendix C.

C. Output Error Estimator for Steady Parametric Systems

The above error estimator for the reduced transfer function $\hat{H}(\tilde{\mu})$ can be easily extended to estimating the output error of MOR for steady parametric systems

$$A(\mu)x(\mu) = B(\mu), \qquad y(\mu) = C(\mu)x(\mu), \tag{10}$$

just by noting that $y(\mu) = C(\mu)A^{-1}(\mu)B(\mu)$ is in the same form as $H(\tilde{\mu})$. In this case, the primal system is the original steady system (10). The reduced primal system can be similarly constructed as the one in (5), i.e.

$$\hat{A}(\mu)z(\mu) = \hat{B}(\mu), \qquad \hat{y}(\mu) = \hat{C}(\mu)z(\mu), \tag{11}$$

which is also the ROM of the original steady system. The dual system is

$$A^T(\mu)x_{du}(\mu) = C^T(\mu). \tag{12}$$

The reduced dual system of (12) can also be similarly constructed as

$$V_{du}^T A^T(\mu) V_{du} z_{du}(\mu) = V_{du}^T C^T(\mu). \tag{13}$$

Then the dual system solution can be approximated by the reduced dual system solution $x_{du}(\mu) \approx \hat{x}_{du} := V_{du} z_{du}(\mu)$. Following a similar argumentation as in the proof for Theorem 1, we can obtain an estimator for the output error $|y(\mu) - \hat{y}(\mu)|$ as below (assuming a SISO system)

$$|y(\mu) - \hat{y}(\mu)| \lesssim |\hat{x}_{r_{du}}^T(\mu) r_{pr}(\mu)| + |\hat{x}_{du}^T(\mu) r_{pr}(\mu)|, \tag{14}$$

where $\hat{x}_{r_{du}} = V_{r_{du}} z_{r_{du}}$ is the approximate solution to the dual-residual system

$$A^T(\mu)x_{du}(\mu) = r_{du}(\mu) \tag{15}$$

and $z_{r_{du}}$ solves the ROM of the dual-residual system (15) i.e.,

$$\tilde{A}(\mu) z_{r_{du}}(\mu) = \tilde{r}_{du}(\mu), \tag{16}$$

where $\tilde{A}(\mu) = V_{r_{du}}^T A(\mu)^T V_{r_{du}}$, $\tilde{r}_{du}(\mu) = V_{r_{du}}^T r_{du}(\mu)$. Note that system (16) is of similar form as the system in (9), just by replacing $\tilde{Q}(\tilde{\mu})$ with $\tilde{A}(\mu)$.

D. Subspace Spanned by $V, V_{du}, V_{r_{du}}$

The key components for computing $\Delta(\tilde{\mu})$ are the projection matrices V, V_{du} and $V_{r_{du}}$ which are used to construct the reduced systems in (5), (7), (9). By definition of (5), V is also the projection matrix for constructing the ROM of the original model. It depends on which method one use to obtain the ROM. In this work, we apply the multi-moment-matching method [12] to derive the ROM. Usually, V_{du} needs to be computed separately and can be computed using the multi-moment-matching method [12] or the reduced basis method. The subspace spanned by $V_{r_{du}}$ should be analyzed based on the state vector of the dual-residual system (8). In the following, we discuss how to construct V, V_{du} and $V_{r_{du}}$ separately.

1) Constructing V Using the Multi-moment-matching Method [12]: If using the multi-moment-matching method proposed in [12] to construct the ROM, then V can be computed as follows. We first consider the state vector $x(t, \mu)$ in frequency domain, i.e., the state vector $x(\tilde{\mu})$ of the primal system. Assume that $Q(\tilde{\mu})$ has the following affine decomposition as

$$Q(\tilde{\mu}) = Q_0 + h_1(\tilde{\mu})Q_1 + \ldots + h_p(\tilde{\mu})Q_p,$$

where $h_i(\tilde{\mu}) : \mathbb{C}^{m+1} \mapsto \mathbb{C}$ are scalar functions of $\tilde{\mu}$. The series expansion of $x(\tilde{\mu})$ as a function of the multiple variables $h_1(\tilde{\mu}), \ldots, h_p(\tilde{\mu})$, can be derived as below,

$$\begin{aligned} x(\tilde{\mu}) &= [Q(\tilde{\mu})]^{-1} B(\mu) \\ &= [Q_0 + Q_1 h_1(\tilde{\mu}) + \ldots + Q_p h_p(\tilde{\mu})]^{-1} B(\mu) \\ &= [Q(\tilde{\mu}^i) + \sigma_1 Q_1 + \ldots + \sigma_p Q_p]^{-1} B(\mu) \\ &= [I - (\sigma_1 M_1 + \ldots + \sigma_p M_p)]^{-1} B_M \\ &= \sum_{i=0}^{\infty} (\sigma_1 M_1 + \ldots + \sigma_p M_p)^i B_M, \end{aligned} \tag{17}$$

where $\sigma_j = h_j(\tilde{\mu}) - h_j(\tilde{\mu}^i)$, and $\tilde{\mu}^i := (\tilde{\mu}_1^i, \ldots, \tilde{\mu}_{m+1}^i)$ is a sample of $\tilde{\mu} = (\mu, s) := (\tilde{\mu}_1, \ldots, \tilde{\mu}_{m+1})$. $B_M = [Q(\tilde{\mu}^i)]^{-1} B(\mu)$, $M_j = -[Q(\tilde{\mu}^i)]^{-1} Q_i$, $j = 1, 2, \ldots, p$. The series expansion in the last equality is the multiplication of B_M with the MacLaurin series expansion of the multivariate

matrix function $[I - M(\sigma)]^{-1}$, where $M(\sigma) = \sigma_1 M_1 + \ldots + \sigma_p M_p$ with $\sigma = (\sigma_1, \ldots, \sigma_p)$. From the above relation between σ_j and $h_j(\tilde{\mu}^i)$, $h(\tilde{\mu}^i) := (h_1(\tilde{\mu}^i), \ldots, h_p(\tilde{\mu}^i))$ is the expansion point at which the series of $x(\tilde{\mu})$ is derived when $x(\tilde{\mu})$ is considered as a function of multiple variables: $h_1(\tilde{\mu}), \ldots, h_p(\tilde{\mu})$. For a given function $h(\cdot)$, $h(\tilde{\mu}^i)$ is uniquely determined by $\tilde{\mu}^i$. Therefore, we call $\tilde{\mu}^i$ the expansion point in the following text, for simplicity. There exist recursions between the coefficients of the series expansion as below,

$$\begin{aligned} R_0 &= \tilde{B}_M, \\ R_1 &= [M_1 R_0, \ldots, M_p R_0], \\ R_2 &= [M_1 R_1, \ldots, M_p R_1], \\ &\vdots \\ R_q &= [M_1 R_{q-1}, \ldots, M_p R_{q-1}], \\ &\vdots \end{aligned} \tag{18}$$

Here, $\tilde{B}_M = B_M$, if $B(\mu)$ does not depend on μ, i.e. $B(\mu) = B$. Otherwise, $\tilde{B}_M = [B_{M_1}, \ldots, B_{M_p}]$, $B_{M_j} = [Q(\tilde{\mu}^i)]^{-1} B_j$, $j = 1, \ldots, p$, if $B(\mu)$ can be approximately wirtten in affine form, e.g., $B(\mu) \approx B_1\beta_1(\mu) + \ldots + B_p\beta_p(\mu)$, $\beta_i(\mu) : \mathbb{C}^m \mapsto \mathbb{C}$ are functions of the parameter vector μ and are the coefficients of the parameter-independent terms $B_i, i = 1, \ldots, p$, in the affine approximation of $B(\mu)$, respectively. Then $V_{\tilde{\mu}^i}$ is computed as

$$\text{range}(V_{\tilde{\mu}^i}) = \text{span}\{R_0, R_1, \ldots, R_q\}_{\tilde{\mu}^i}, \tag{19}$$

Note that the number of columns in $R_j, j \geq 0$, grows exponentially with j. To avoid exponential increase of column dimension, we usually require $q \leq 1$ and use only the first two R_j's. The matrix $V_{\tilde{\mu}^i}$ depends on the expansion point $\tilde{\mu}^i$. Finally, V can be constructed as

$$V = \text{orth}\{V_{\tilde{\mu}^1}, \ldots, V_{\tilde{\mu}^l}\}. \tag{20}$$

Since the proposed error estimator does not depend on the MOR method, another possibility of constructing V is using time-domain MOR methods, such as the reduced basis method (RBM), or the method of proper orthogonal decomposition (POD). These methods use the snapshots in time domain (trajectories of the state vector x) to obtain V. As it is not the focus of this work, we refer to [18], [19] for detailed information on these two methods for parametric dynamical systems.

2) Constructing V_{du} Using Reduced Basis Method: Similarly, we could construct V_{du} through snapshot based methods, or the multi-moment-matching method. The dual system is a steady system, which does not involve the trajectory of the state vector in time domain. The snapshots are the state vector at different samples of the parameter μ. In this case the reduced basis method can be seen as a special case of the multi-moment-matching method. If using the reduced basis method, we have

$$\text{range}(V_{du}) = \text{span}\{Q^{-T}(\tilde{\mu}^1)C^T(\tilde{\mu}^1), \ldots, Q^{-T}(\tilde{\mu}^l)C^T(\tilde{\mu}^l)\}, \tag{21}$$

where $Q^{-T}(\tilde{\mu}^i)C^T(\tilde{\mu}^i), i = 1, \ldots, l$, are the snapshots selected through a greedy algorithm.

3) Constructing V_{du} Using Multi-moment-matching: If using the multi-moment-matching method, V_{du} can also be constructed similarly as V. For completeness, we shortly discuss how to compute V_{du} based on multi-moment-matching. Considering the dual system in (6), $x_{du}(\tilde{\mu})$ can be written as

$$\begin{aligned} x_{du}(\tilde{\mu}) &= [Q(\tilde{\mu})]^{-T} C^T(\mu) \\ &= [Q_0^T + Q_1^T h_1(\tilde{\mu}) + \ldots + Q_p^T h_p(\tilde{\mu})]^{-1} C^T(\mu) \\ &= [I - (\sigma_1 \tilde{M}_1 + \ldots + \sigma_p \tilde{M}_p)]^{-1} C_M \\ &= \sum_{i=0}^{\infty} (\sigma_1 \tilde{M}_1 + \ldots + \sigma_p \tilde{M}_p)^i C_M, \end{aligned} \tag{22}$$

where $C_M = [Q(\tilde{\mu}^i)]^{-T} C^T(\mu)$, $\tilde{M}_j = [Q(\tilde{\mu}^i)]^{-T} Q_j^T$, $j = 1, 2, \ldots, p$. The recursions between the coefficients of the series expansion in (22) are

$$\begin{aligned} \tilde{R}_0 &= \tilde{C}_M, \\ \tilde{R}_1 &= [\tilde{M}_1 \tilde{R}_0, \ldots, \tilde{M}_p \tilde{R}_0], \\ \tilde{R}_2 &= [\tilde{M}_1 \tilde{R}_1, \ldots, \tilde{M}_p \tilde{R}_1], \\ &\vdots \\ \tilde{R}_q &= [\tilde{M}_1 \tilde{R}_{q-1}, \ldots, \tilde{M}_p \tilde{R}_{q-1}], \\ &\vdots \end{aligned} \tag{23}$$

Here, $\tilde{C}_M = C_M$, if $C(\mu)$ does not depend on μ, i.e. $C(\mu) = C$. Otherwise, $\tilde{C}_M = [C_{M_1}, \ldots, C_{M_p}]$, $C_{M_i} = [Q(\tilde{\mu}^i)]^{-1} C_j$, $j = 1, \ldots, p$, if $C(\mu)$ can be approximated by an affine form, e.g., $C(\mu) \approx C_1\gamma_1(\mu) + \ldots + C_p\gamma_p(\mu)$, where $\gamma_i(\mu) : \mathbb{C}^m \mapsto \mathbb{C}$ are functions of the parameter vector μ and are the coefficients of the parameter-independent terms $C_i, i = 1, \ldots, p$, in the affine approximation of $C(\mu)$, respectively. Then $V_{du}^{\tilde{\mu}^i}$ is computed as

$$\text{range}(V_{du}^{\tilde{\mu}^i}) = \text{span}\{\tilde{R}_0, \tilde{R}_1, \ldots, \tilde{R}_q\}_{\tilde{\mu}^i}. \tag{24}$$

Finally, V_{du} can be constructed as

$$V_{du} = \text{orth}\{V_{du}^{\tilde{\mu}^1}, \ldots, V_{du}^{\tilde{\mu}^l}\}. \tag{25}$$

It is easy to see that if $C(\mu)$ does not depend on μ, i.e. $C(\mu) = C$, then $R_0 = C_M = [Q(\tilde{\mu}^i)]^{-T} C^T$, which is the snapshot of $x_{du}(\tilde{\mu})$ at $\tilde{\mu}^i$. In this case, if only R_0 is included in $V_{du}^{\tilde{\mu}^i}$ (24) for each expansion point $\tilde{\mu}^i$, then the multi-moment-matching method for the steady dual system (6) is the reduced basis method in (21).

4) Constructing $V_{r_{du}}$: From the state vector of the dual-residual system (8), we see that

$$\begin{aligned} x_{r_{du}}(\tilde{\mu}) &= Q^{-T}(\tilde{\mu}) r_{du}(\tilde{\mu}) \\ &= Q^{-T}(\tilde{\mu})[C^T(\mu) - Q^T(\tilde{\mu})\hat{x}_{du}(\tilde{\mu})] \\ &= Q^{-T}(\tilde{\mu}) C^T(\mu) - V_{du} z_{du}(\tilde{\mu}), \end{aligned} \tag{26}$$

where $Q^{-T}(\tilde{\mu})C^T(\mu)$ is nothing but the state vector $x_{du}(\tilde{\mu})$ of the dual system. Considering the series expansion of $x_{du}(\tilde{\mu})$ in (22), we see that taking the same expansion point as in (22), the series expansion leads to the subspace range(V_{du}). Finally, $Q^{-T}(\tilde{\mu})C^T(\mu)$ in the last equality of (26) provides no new information than V_{du}, so that we can use range(V_{du}) as the subspace for approximating the trajectory space of $x_{r_{du}}(\tilde{\mu})$, i.e. $V_{r_{du}} = V_{du}$. However, from Theorem 2, we know that $V_{r_{du}}$

should be different from V_{du}. Therefore, if we use expansion points different from those used for V_{du} to obtain a second projection matrix $V^1_{r_{du}}$ which is different from V_{du}, then the projection matrix $V_{r_{du}} := \text{orth}\{V^1_{r_{du}}, V_{du}\}$ should represent the trajectory of $x_{r_{du}}(\tilde{\mu})$ well.

There are two choices for computing $V^1_{r_{du}}$. The first choice is using the reduced basis method, i.e.

$$\text{range}(V^1_{r_{du}}) = \text{span}\{Q^{-T}(\tilde{\mu}^1)C^T(\mu^1), \ldots, Q^{-T}(\tilde{\mu}^l)C^T(\mu^l)\}, \tag{27}$$

where $\tilde{\mu}^j, j = 1, \ldots, l$, should be different from $\tilde{\mu}^i$ used for V_{du} in either (21) or (25). The second choice is using the multi-moment-matching method as in (24) and (25), by choosing expansion points which are different from those used there.

$$\text{range}(V^{\tilde{\mu}^j}_{r_{du}}) = \text{span}\{\tilde{R}_0, \tilde{R}_1, \ldots, \tilde{R}_q\}_{\tilde{\mu}^j}, j = 1, \ldots, l. \tag{28}$$

Finally,

$$\text{range}(V_{r_{du}}) = \text{orth}\{V^{\tilde{\mu}^1}_{r_{du}}, \ldots, V^{\tilde{\mu}^l}_{r_{du}}, V_{du}\}. \tag{29}$$

The $\tilde{\mu}^j$ in (27) or (28) can be selected through a greedy algorithm by searching the maximum of $|\hat{x}^T_{r_{du}}(\tilde{\mu})r_{pr}(\tilde{\mu})|$, the first part of $\Delta(\tilde{\mu})$ associated with $\hat{x}_{r_{du}}$, and are usually different from the $\tilde{\mu}^i$ used for computing V_{du}. For example, $\tilde{\mu}^j$ in (28) is computed as $\tilde{\mu}^i_\alpha$ in Algorithm 2 and is computed in Step 11, whereas $\tilde{\mu}^i$ used for computing V_{du} is computed in Step 10. Analogously, for non-parametric systems, s^α_i for $V_{r_{du}}$ and s_i for V_{du}, are computed in Step 10 and Step 9 in Algorithm 1, respectively.

E. Error Estimator for MIMO Systems

For MIMO systems, the transfer function is a matrix, the error of each entry in $\hat{H}(\tilde{\mu})$ can be estimated by

$$\begin{aligned} &|H_{ij}(\tilde{\mu}) - \hat{H}_{ij}(\tilde{\mu})| \\ &\lesssim |\hat{x}^T_{r_{du}}(\tilde{\mu})r_{pr}(\tilde{\mu})| + |\hat{x}^T_{du}(\tilde{\mu})r_{pr}(\tilde{\mu})| =: \Delta_{ij}(\tilde{\mu}), \end{aligned}$$

where $\hat{x}_{du}(\tilde{\mu})$ is the approximate solution to the dual system by replacing C^T in (6) with $C^T(:, i)$, the ith row of C. Similarly, $r_{pr}(\tilde{\mu})$ is the residual of $\hat{x}_{pr}(\mu)$, the approximate solution to the primal system by replcing B in (4) with $B(:, j)$, the jth column of B. Finally, $\Delta(\tilde{\mu}) = \max_{ij} \Delta_{ij}(\tilde{\mu})$.

F. Efficiently Computing the Error Estimator

As analyzed in [16], if the system matrices are of affine forms, then the residuals involved in the error estimator can be even more efficiently computed. For example, if $Q(\tilde{\mu})$ is affine with respect to the parameters, i.e.

$$Q(\tilde{\mu}) = Q_0 + Q_1 h_1(\tilde{\mu}) + \ldots + Q_p h_p(\tilde{\mu}),$$

then

$$\begin{aligned} r_{pr}(\tilde{\mu}) &= B(\tilde{\mu}) - V^T Q(\tilde{\mu}) V z_{pr}(\tilde{\mu}) \\ &= B(\tilde{\mu}) - [V^T Q_0 V + \ldots + V^T Q_p V h_p(\tilde{\mu})] z_{pr}(\tilde{\mu}). \end{aligned} \tag{30}$$

The parameter independent terms $V^T Q_0 V$,...,$V^T Q_p V$ can be precomputed and repeatedly used for estimating the ROM error at different values of the parameters.

In the next section, we show how to adaptively compute $V, V_{du}, V_{r_{du}}$, so that the reduced systems for error estimation as well as the ROM of the original system are adaptively constructed.

III. Adaptive MOR

The aim of an efficient error estimator is to construct a ROM of the original system with satisfying accuracy and high reliability. In the following, we show algorithms for constructing the ROM of the original system, where the error estimator acts as a guidance for greedy construction of the projection matrix V for the ROM. To compute the error estimator, the projection matrices V_{du}, $V_{r_{du}}$ need to be constructed simultaneously with V.

Greedy algorithms using the error bound $\Delta_0(\tilde{\mu})$ have been proposed in [16] for adaptive MOR of linear systems with and without parameters. In this work, we extend those algorithms and adapt them using the proposed error estimator. The resulting algorithms usually take much fewer iterative steps and converges much faster than those in [16].

Algorithm 1 presents the greedy algorithm for non-parametric linear systems. For non-parametric systems, $\tilde{\mu} = s$ in $\Delta(\tilde{\mu})$ and $\Delta_0(\tilde{\mu})$. In this case, the ROM of the system can be constructed using moment-matching [7]. However, for high-frequency problems, expanding the transfer function only at zero to construct the projection matrix V usually results in large errors of the reduced transfer function at high frequencies. The following algorithm provides a way of enriching V by adaptively selecting the expansion points. In Algorithm 1, $\tilde{C}(s_i) = (s_i E - A)^{-T} C^T$, $\tilde{B}(s_i) = (s_i E - A)^{-1} B$ and $q \ll n$. ε_{tol} is the tolerance for the error of the reduced transfer function. Once the maximal error estimator over the whole sample set Ξ is below ε_{tol}, the algorithm stops. At every iteration, the s sample corresponding to the maximal error estimator is chosen as the next expansion point. An additional expansion point is selected in Step 10 for the construction of $V_{r_{du}}$. Steps 7-8 orthogonalize the vectors in V_{s_i}, $V^{s_i}_{du}$ and $V^{s_i}_{r_{du}}$ against the existing vectors, respectively. Algorithm 2 shows the adaptive scheme for linear parametric systems. The expansion points $\tilde{\mu}^i$ are completely determined by the corresponding original parameter sample μ^i. Actually, by computing the expansion points $\tilde{\mu}^i$ from the error estimator, we immediately know the corresponding μ^i which generates $\tilde{\mu}^i$. This is useful when the input matrix $B(\mu)$ or the output matrix $C(\mu)$ is parametric, and when the reduced basis method is used to compute, e.g., $V_{r_{du}}$ from (27). As in Algorithm 1, orth in Steps 7-9 orthogonalizes $V_{\tilde{\mu}^i}$, $V^{\tilde{\mu}^i}_{du}$, and $V^{\tilde{\mu}^i_\alpha}_{r_{du}}$ w.r.t. the already existing basis vectors. For more clarity, we present two flowcharts in Figures 1-2 to describe the processes of the greedy ROM construction for non-parametric and parametric systems, respectively.

IV. Simulation Results

We use four models to show the robustness of the error estimator. The first two are non-parametric SISO systems. One is a CD player model, the other is a model of an RLC tree

Algorithm 1 Greedy ROM construction for non-parametric systems.

Input: System matrices $E, A, B, C, \varepsilon_{tol}$, Ξ: a set of samples of s covering the interesting frequency range.

Output: The projection matrix V and the ROM in (3).

1: $V = []$, $V_{du} = []$,$V_{r_{du}} = []$, set $\epsilon = \varepsilon_{tol} + 1, q > 1$.
2: Initial expansion point: s_1: the first sample in Ξ, s_1^α: the last sample in Ξ, $i = 1$.
3: **while** $\epsilon > \varepsilon_{tol}$ **do**
4: $\text{range}(V_{s_i}) = \text{span}\{\tilde{B}(s_i), \ldots, (\tilde{A}(s_i))^{q-1}\tilde{B}(s_i)\}$, $\tilde{A}(s) = (sE - A)^{-1}E$.
5: $\text{range}(V_{du}^{s_i}) = \text{span}\{\tilde{C}(s_i), \ldots, (\tilde{A}_c(s_i))^{q-1}\tilde{C}(s_i)\}$, $\tilde{A}_c(s) = (sE - A)^{-T}E^T$.
6: $\text{range}(V_{r_{du}}^{s_i^\alpha}) = \text{span}\{\tilde{C}(s_i^\alpha), \ldots, (\tilde{A}_c(s_i^\alpha)^{q-1}\tilde{C}(s_i^\alpha)\}$.
7: $V = \text{orth}\{V, V_{s_i}\}$, $V_{du} = \text{orth}\{V_{du}, V_{du}^{s_i}\}$.
8: $V_{r_{du}} = \text{orth}\{V_{du}, V_{r_{du}}, V_{r_{du}}^{s_i^\alpha}\}$.
9: $i = i + 1$, $s_i = \arg\max_{s\in\Xi} \Delta(s)$.
10: $s_i^\alpha = \arg\max_{s\in\Xi} |\hat{x}_{r_{du}}^T(s) r_{pr}(s)|$. (first part of $\Delta(s)$)
11: $\epsilon = \Delta(s_i)$.
12: **end while**
13: Compute the ROM in (3) using V and $W = V$.

Algorithm 2 Greedy ROM construction for parametric systems.

Input: System matrices $E(\mu), A(\mu), B(\mu), C(\mu), \varepsilon_{tol}$, Ξ: a set of samples of $\tilde{\mu}$ covering the interesting frequency range.

Output: The projection matrix V and the ROM in (3).

1: $V = []$, $V_{du} = []$,$V_{r_{du}} = []$, set $\epsilon = \varepsilon_{tol} + 1$.
2: Initial expansion point: $\tilde{\mu}^1$: the first sample in Ξ, $\tilde{\mu}_\alpha^1$: the last sample in Ξ, $i = 1$.
3: **while** $\epsilon > \varepsilon_{tol}$ **do**
4: Compute $V_{\tilde{\mu}^i}$ following (19).
5: Compute $V_{du}^{\tilde{\mu}^i}$ following (24).
6: Compute $V_{r_{du}}^{\tilde{\mu}_\alpha^i}$ following (28).
7: $V = \text{orth}\{V, V_{\tilde{\mu}^i}\}$.
8: $V_{du} = \text{orth}\{V_{du}, V_{du}^{\tilde{\mu}^i}\}$.
9: $V_{r_{du}} = \text{orth}\{V_{du}, V_{r_{du}}, V_{r_{du}}^{\tilde{\mu}_\alpha^i}\}$.
10: $i = i + 1$, $\tilde{\mu}^i = \arg\max_{\tilde{\mu}\in\Xi} \Delta(\tilde{\mu})$.
11: $\tilde{\mu}_\alpha^i = \arg\max_{\tilde{\mu}\in\Xi} |\hat{x}_{r_{du}}^T(\tilde{\mu}) r_{pr}(\tilde{\mu})|$. (first part of $\Delta(\tilde{\mu})$)
12: $\epsilon = \Delta(\tilde{\mu}^i)$.
13: **end while**
14: Compute the ROM in (3) using V and $W = V$.

circuit, which can be instantiated for any level l. Between two consecutive levels, the circuit branch segments of the lower level each split into two children, yielding $\sum_{i=0}^{l-1} 2^i$ in the circuit of level l. Each segment is made up of four RL pairs in series, representing the wiring on a chip, with four capacitors to ground, representing the wire-substrate interaction, see Fig. 3. The dimension of this model is $n = 6134$. The third example is derived from an electrical circuit of a CMOS-inverter driven

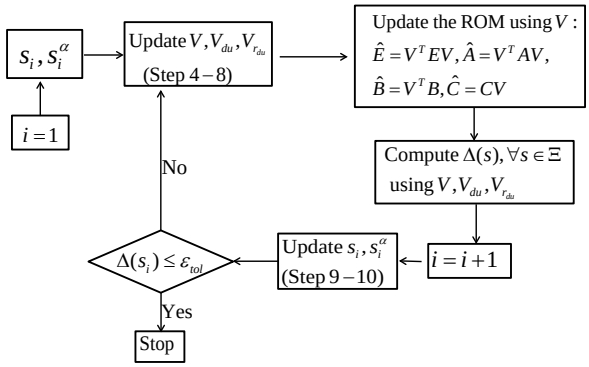

Fig. 1. The greedy process of ROM construction by Algorithm 1 for non-parametric systems.

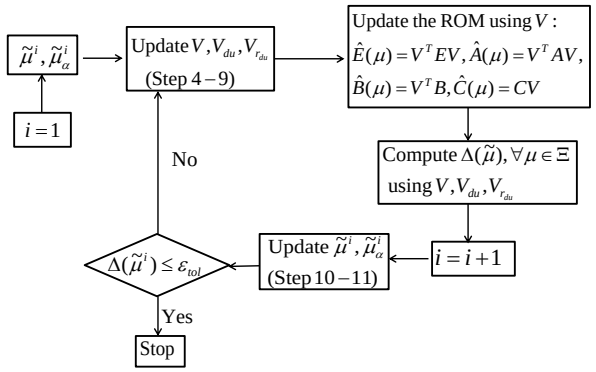

Fig. 2. The greedy process of ROM construction by Algorithm 2 for parametric systems.

two-bit bus modelled by 40 RLC sections. The discretized system was obtained using modified nodal analysis available in SPICE, with $n = 980$. It has 4 inputs and 4 outputs, and no parameters. Both the CD player model and the MIMO model are from the SLICOT benchmark collection[1]. The last one is the model of a butterfly-shaped micro-gyroscope. It is a parametric system with $n = 17,931$ and is available from the MOR benchmark collection[2].

The interesting frequency of the CD player model is $f \in [0, 1\text{ MHz}]$. The interesting frequency of the second and the third models is $f \in [0, 3\text{ GHz}]$. The Gyroscope model is a low frequency problem with $f \in [0.025, 0.25]$MHz. The samples for the Laplace variable s are taken along the imaginary axis, i.e. $s = \jmath\omega$, and $\omega = 2\pi f$. Here and below, $\jmath$ is the imaginary unit.

The error tolerance ε_{tol} used in the greedy algorithms, i.e. the error tolerance for the error of the ROMs, is set as 1×10^{-3} for the first three examples, while for the last example, we set $\varepsilon_{tol} = 1 \times 10^{-7}$, since the transfer function $H(\mu)$ is of small magnitude: $O(10^{-7})$ at some samples of the parameter.

[1]URL: http://slicot.org/20-site/126-benchmark-examples-for-model-reduction
[2]URL: https://morwiki.mpi-magdeburg.mpg.de/morwiki

The error bound $\Delta_0(\tilde{\mu})$ was tested using the first two examples in [16], the second and third example are also used in [9] to demonstrate the heuristic adaptive expansion point selection technique. However, as is discussed in the introduction, the technique proposed in [9] is not robust for the MIMO example.

For all the non-parametric examples, we use $q = 3$ for Steps 4-6 of Algorithm 1. For the parametric model, we use $(R_0, R_1)/(\tilde{R}_0, \tilde{R}_1)$ to generate the matrices $V_{\tilde{\mu}^i}$, $V_{du}^{\tilde{\mu}^i}$ and $V_{r_{du}}^{\tilde{\mu}_\alpha^i}$ in Steps 4-6 of Algorithm 2. At each iteration step of Algorithm 1 and Algorithm 2, the maximal error estimator/bound in Ξ, is computed, and is used as the error control for the ROM. Therefore, the maximal true error $\varepsilon_{\max} = \max_{\tilde{\mu}^i \in \Xi} \varepsilon(\tilde{\mu}^i)$ is used for a comparison, where $\varepsilon(\tilde{\mu}^i)$ is the true error of the ROM evaluated at $\tilde{\mu}^i$, at the current iteration of the algorithm.

For both algorithms, the initial expansion point for computing V, V_{du} is taken as the first sample in Ξ, and the initial expansion points for computing $V_{r_{du}}$ is taken as the last sample in Ξ to make the two groups of expansion points as different as possible. The sampling rule used for generating Ξ for each example is detailed in the following subsections.

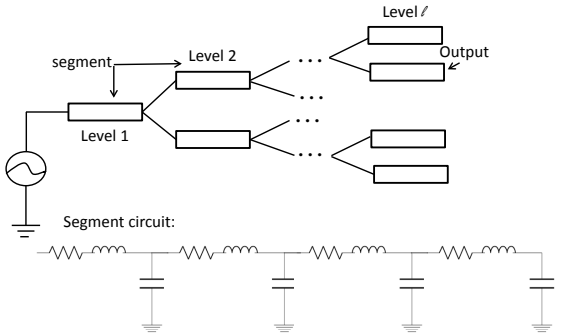

Fig. 3. An RLC tree example.

A. The CD Player Model

The CD player model describes a compact disc player which is an optical decoding device that reproduces high-quality audio from a digitally coded signal recorded as a spiral track on a reflective disc [20]. More details about the model and the device can be found in [20], where a schematic view of a compact disc mechanism is presented. The CD player model was collected as a benchmark example for MOR in [21] and is widely used for testing MOR methods. The training set Ξ for this model contains 60 samples of s, and then the finally obtained ROM is validated at 600 samples of s covering the whole interesting frequency range. The samples are taken from the interval $[0, 1\text{MHz}]$ using the MATLAB function "logspace". Algorithm 1 takes 8 iterations to converge when using $\Delta_0(\tilde{\mu})$, while it takes 7 iterations when $\Delta(\tilde{\mu})$ is used. The details for $\Delta_0(\tilde{\mu})$ are shown in Table II, and those for $\Delta(\tilde{\mu})$ are shown in Table I. Interesting to see is that $\Delta_0(\tilde{\mu})$

TABLE I
CD PLAYER, $\varepsilon_{tol} = 10^{-3}$, $q = 3$, $r = 52$.

iteration	$f := s_i/(2\pi\jmath)$ (Hz)	$\varepsilon_{\max}$	$\Delta(s_i)$
1	1	40.75	51
2	11.7	30.16	35.75
3	47.6	0.75	5.41
4	11.8	0.32	0.4
5	96.1	0.03	0.02
6	626	0.002	0.002
7	1000	8.28×10^{-4}	8.38×10^{-4}

TABLE II
CD PLAYER, $\varepsilon_{tol} = 10^{-3}$, $q = 3$, $r = 60$.

iteration	$f := s_i/(2\pi\jmath)$(Hz)	$\varepsilon_{\max}$	$\Delta_0(s_i)$
1	1	40.75	3.87×10^3
2	11.7	19.34	2.8×10^3
3	47.6	0.59	618
4	11.8	0.31	164
5	96.1	0.05	1.59
6	626	0.002	0.26
7	7.3	5.9×10^{-4}	0.08
8	1000	5.2×10^{-6}	1.43×10^{-4}

and $\Delta(\tilde{\mu})$ provide almost the same expansion points, except at the final iteration step. However, due to overestimation of the true error, Δ_0 takes one more iteration step, and produces a ROM with true error being too below the tolerance.

We further validate the ROM obtained by both $\Delta_0(s)$ and $\Delta(s)$ at 600 samples, the results are plotted in Figure 4-5. It is clear from the figure that there are relatively big gaps between $\Delta_0(s)$ and the true error at certain points, whereas $\Delta(s)$ tightly estimates the true error. The effectivity of the proposed error estimator $\texttt{eff}(s) := \Delta(s)/\epsilon_{true}(s)$, evaluated at 600 frequency samples, satisfies $0.12 < \texttt{eff}(s) < 17$. The effectivity of the error bound satisfies $0.11 < \texttt{eff}(s) < 91$. We observe that underestimation of both Δ and Δ_0 happens only at samples with very small true errors being smaller than 10^{-12} which may be caused by rounding errors. The

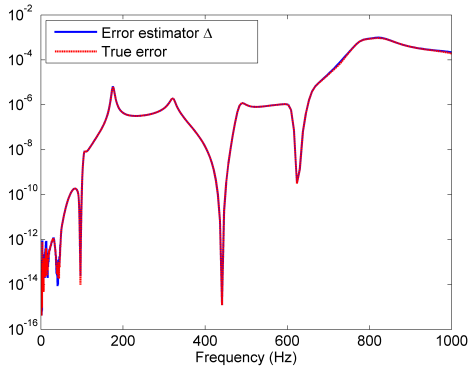

Fig. 4. CD player: $\Delta(s)$ vs true error at 600 frequency samples in [0, 1 MHz].

magnitude of the transfer function of the original model and the that of the ROM computed by Algorithm 1 using the proposed error estimator, are plotted in Figure 6. The reduced transfer function approximates the original transfer function well.

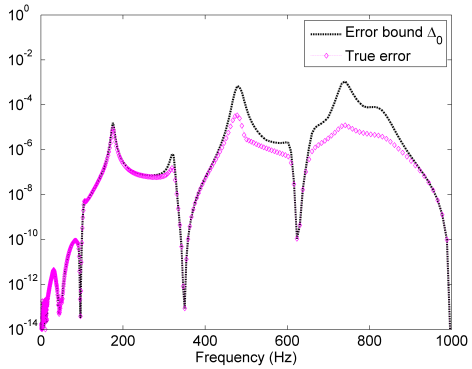

Fig. 5. CD player: $\Delta_0(s)$ vs true error at 600 frequency samples in [0, 1 MHz].

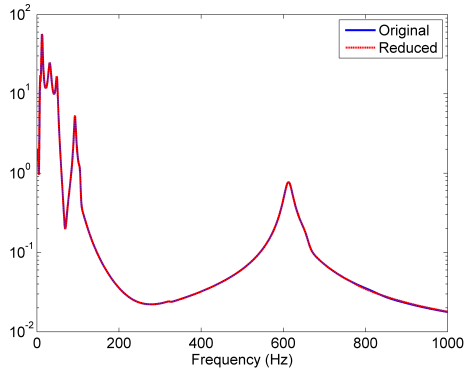

Fig. 6. CD player: original $|H(\jmath\omega)|$ vs reduced $|\hat{H}(\jmath\omega)|$.

B. The RLC Tree Model

We use a training set Ξ with 90 frequency samples covering the whole frequency range $[0, 3\text{ GHz}]$. The samples s_i are taken using the function $f_i = 3 \times 10^{i/10}, s_i = 2\pi\jmath, i = 1, \ldots, 90$. Here $\jmath$ is the imaginary unit. The greedy algorithm takes 3 iterations to converge with the error estimator, but uses 8 iterations to converge with the error bound, see Table VI-IV. This is mainly due to low tightness of the error bound leading to stagnation of the true error after the 4th iteration, and the unnecessary iterations after the 3rd iteration.

The derived ROMs are validated on 900 samples covering the interesting frequency range. The performance of $\Delta(\tilde{\mu})$ and $\Delta_0(\tilde{\mu})$ is plotted in Figure 7 and Figure 8, respectively. It is clear that the error bound is too rough at many samples, this is mainly due to the small magnitude of $\sigma_{\min}(Q(s))$ in (34) for this example. The effectivity of the proposed error estimator $\texttt{eff}(s) := \Delta(s)/\epsilon_{true}(s)$, evaluated at 900 frequency samples, satisfies $0.003 < \texttt{eff}(s) < 244$, which looks not sharp. However, it is noticed that the true error is close to zero at many sample points. If we only consider the true error larger than 1×10^{-11}, then the effectivity of the error estimator falls into the interval $[0.37, 51]$, i.e. $0.37 < \texttt{eff}(s) < 51$. The effectivity of the error bound satisfies $0.45 < \texttt{eff}(s) < 2 \times 10^8$, and is not always a bound. Similar as before, the underestimation only happens at samples with true errors smaller than 10^{-12}. This again

TABLE III
RLC TREE, $\varepsilon_{tol} = 10^{-3}$, $q = 3$, $r = 20$.

iteration	$f := s_i/(2\pi\jmath)$ (Hz)	$\varepsilon_{\max}$	$\Delta(s_i)$
1	3.77	0.18	0.6
2	2.38×10^9	0.016	0.06
3	1.19×10^9	6.12×10^{-6}	6.45×10^{-6}

TABLE IV
RLC TREE, $\varepsilon_{tol} = 10^{-3}$, $q = 3$, $r = 25$.

iteration	$f := s_i/(2\pi\jmath)$(Hz)	$\varepsilon_{\max}$	$\Delta_0(s_i)$
1	3.77	0.19	1×10^6
2	1.5×10^9	0.06	9.13×10^4
3	2.4×10^9	4.11×10^{-6}	1.14×10^3
4	3×10^9	5.75×10^{-10}	0.009
5	1.9×10^9	5.68×10^{-10}	0.004
6	9.5×10^8	5.76×10^{-10}	0.002
7	3.7×10^8	5.79×10^{-10}	0.01
8	1.5×10^8	5.70×10^{-10}	1.8×10^{-7}

be caused by rounding errors when evaluating true errors with very small values. Magnitudes of the original and the

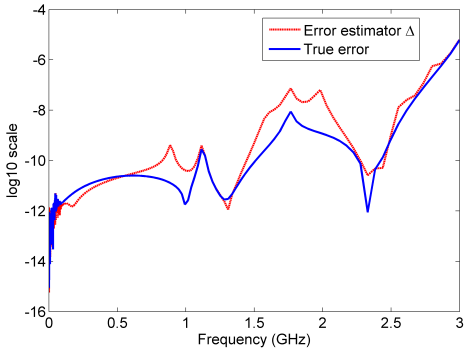

Fig. 7. RLC tree: $\Delta(s)$ vs true error at 900 frequency samples in [0, 3 GHz].

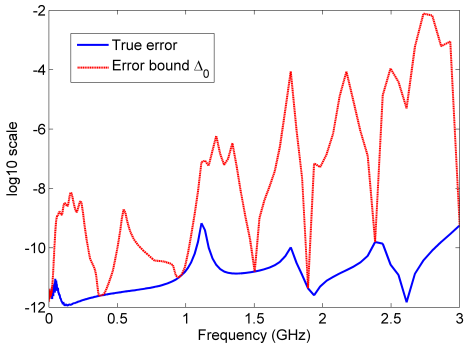

Fig. 8. RLC tree: $\Delta_0(s)$ vs true error at 900 frequency samples in [0, 3 GHz].

reduced transfer functions are plotted in Figure 9, respectively. The reduced transfer function is computed by Algorithm 1 using the proposed error estimator and reproduces the original transfer function.

C. The MIMO Example

This example has the same frequency range as the second example, therefore we use the same samples as for the RLC

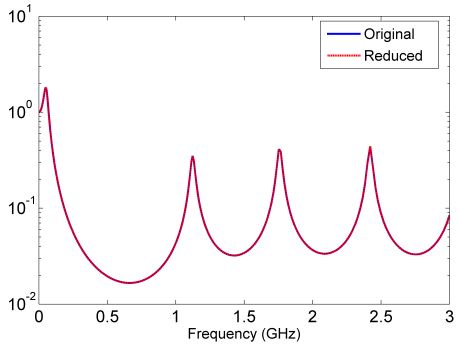

Fig. 9. RLCtree: original $|H(\jmath\omega)|$ vs reduced $|\hat{H}(\jmath\omega)|$.

tree model to construct Ξ. The error estimator is the maximal error estimator defined in subsection II-E. The true error is the maximal true error $\epsilon_{true}(s) = \max_{ij} |H_{ij}(s) - \hat{H}_{ij}(s)|$. Algorithm 1 converges in 3 iterations with $\Delta(s)$, but in 4 iterations with $\Delta_0(s)$.

The MIMO ROM is of order $r = 73$ using $\Delta(s)$. Fig. 10 and Fig. 11 plot $\Delta(s)$, $\Delta_0(s)$ and their corresponding $\epsilon_{true}(s)$ at 900 frequency samples in $[0, 3$ GHz$]$, respectively. It shows that $\Delta(s)$ catches the true error very well, and is much sharper than $\Delta_0(s)$. The effectivity $\texttt{eff}(s) := \Delta(s)/\epsilon_{true}(s)$, evaluated at 900 frequency samples, satisfies $0.11 < \texttt{eff}(s) < 5$. The effectivity of the error bound satisfies $0.46 < \texttt{eff}(s) < 6.5 \times 10^6$. The underestimation of both $\Delta(s)$ and $\Delta_0(s)$ happens at samples with true errors smaller than 5×10^{-9}. Fig. 12 plots the magnitudes of the functions $H(s)$ and $\hat{H}(s)$ at input port 1 and output port 4, though $\hat{H}(s)$ reproduces $H(s)$ at all other input-output ports, too. In order to show more resonances, we extend the frequency range to 5GHz. Algorithm 1 based on frequency samples taken from the extended frequency interval $[0, 5$ GHz$]$, can still produce a ROM meeting the required accuracy.

Remark 1: In the proceedings paper [17], we use a training set Ξ with 18 samples and $q = 5$ for moment-matching in Algorithm 1, therefore, we obtain different results. The error estimator works equally well. Due to space limitation, these details were not presented there. Techniques of adaptive training sampling [22], [23] could be applied to obtain an optimal training set so that the ROM is robust for as many other samples outside of the training set as possible. However, it is beyond the scope of this paper.

TABLE V
MIMO EXAMPLE, $\varepsilon_{tol} = 10^{-3}$, $q = 3$, $r = 73$.

iteration	$f := s_i/(2\pi\jmath)$ (Hz)	$\varepsilon_{\max}$	$\Delta(s_i)$
1	3.77	0.28	0.28
2	1.5×10^9	5.9×10^{-5}	0.0023
3	3.7×10^7	4.72×10^{-8}	1.43×10^{-7}

TABLE VI
MIMO EXAMPLE, $\varepsilon_{tol} = 10^{-3}$, $q = 3$, $r = 100$.

iteration	$f := s_i/(2\pi\jmath)$ (Hz)	$\varepsilon_{\max}$	$\Delta_0(s_i)$
1	3.77	0.28	2.47×10^7
2	1.89×10^9	1.87×10^{-5}	542
3	3×10^9	5.49×10^{-8}	61.89
4	1.19×10^6	9.83×10^{-10}	5.54×10^{-5}

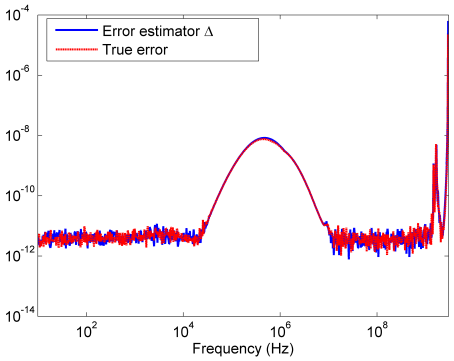

Fig. 10. MIMO example: $\Delta(s)$ vs true error at 900 frequency samples in [0, 3 GHz].

D. The Parametric Example

The micro-gyroscope model is a second-order parametric system with four parameters,

$$\begin{aligned} M(d)\ddot{x} + D(\theta,\alpha,\beta,d)\dot{x} + T(d)x &= Bu(t), \\ y &= Cx. \end{aligned}$$

Here, $M(d) = (M_1 + dM_2)$, $T(d) = (T_1 + \frac{1}{d}T_2 + dT_3)$, $D(\theta,\alpha,\beta,d) = \theta(D_1 + dD_2) + \alpha M(d) + \beta T(d) \in R^{n\times n}$, $n = 17,913$. The parameters are d, θ, α, β. $d \in [100\%, 200\%]$, the width of the bearing, taken as the percentage of the base value, and $\theta \in [10^{-7}, 10^{-5}]$MHz, the rotation velocity along the x-axis. α, β contribute to the proportional damping [24]. The numerical values of d, θ and f for simulation are taken from the intervals [1,2], $[10^{-7}, 10^{-5}]$ and [0.025, 0.25], respectively.

After Laplace transform, the system in frequency domain is

$$\begin{aligned} s^2M(d)x + sD(\theta,\alpha,\beta,d)x + T(d)x &= Bu_{\mathcal{L}}(s), \\ y &= Cx. \end{aligned}$$

The above system can be rewritten into the affine form,

$$\begin{aligned} Q(\tilde{\mu})x &= Bu_{\mathcal{L}}(\tilde{\mu}), \\ y &= Cx, \end{aligned}$$

where $Q(\tilde{\mu}) = T_1 + \tilde{\mu}_1 M_1 + \tilde{\mu}_2 M_2 + \tilde{\mu}_3 D_1 + \tilde{\mu}_4 D_2 + \tilde{\mu}_5 M_1 + \tilde{\mu}_6 M_2 + \tilde{\mu}_7 T_1 + \tilde{\mu}_8 T_2 + \tilde{\mu}_9 T_3 + \tilde{\mu}_{10} T_2 + \tilde{\mu}_{11} T_3$. Here $\tilde{\mu} = (\tilde{\mu}_1, \ldots, \tilde{\mu}_{11})^T$ includes the newly generated parameters, $\tilde{\mu}_1 = s^2$, $\tilde{\mu}_2 = s^2 d$, $\tilde{\mu}_3 = s\theta$, $\tilde{\mu}_4 = s\theta d$, $\tilde{\mu}_5 = s\alpha$, $\tilde{\mu}_6 = s\alpha d$, $\tilde{\mu}_7 = s\beta$, $\tilde{\mu}_8 = s\beta/d$, $\tilde{\mu}_9 = s\beta d$, $\tilde{\mu}_{10} = 1/d$, $\tilde{\mu}_{11} = d$.

For this example, we use 75 random training samples (3 for θ, 5 for s, 5 for d, $\alpha = \beta = 0$) for the error estimator $\Delta(\tilde{\mu})$, but use 150 random samples (3 for θ, 10 for s, 5 for d, $\alpha = \beta = 0$) for the error bound $\Delta_0(\tilde{\mu})$, because the error bound stagnates at around 1e-7 when using 75 samples. Therefore, the runtime of the greedy algorithm using the error bound is counted for the 150 training samples. Afterwards, both are validated at 2500 samples, with $\beta = 10^{-9}$ and $\alpha = 0.1$ being nonzero.

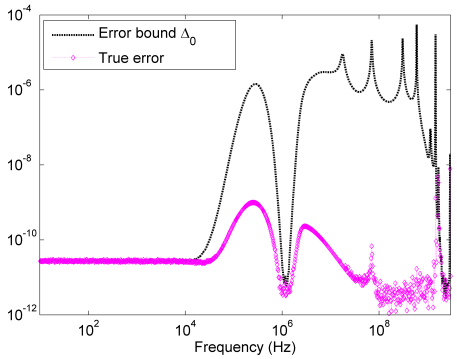

Fig. 11. MIMO example: $\Delta_0(s)$ vs true errors at 900 frequency samples in [0, 3 GHz].

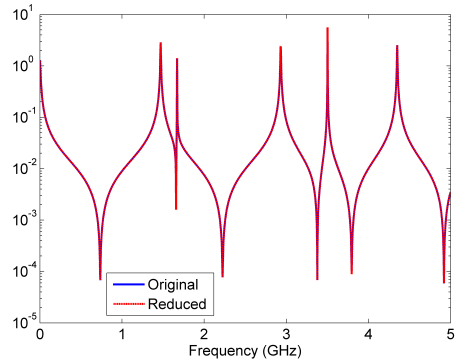

Fig. 12. MIMO example: input port 1, output port 4, original $|H_{41}(\jmath\omega)|$ vs reduced $|\hat{H}_{41}(\jmath\omega)|$.

The algorithm then converges in 6 iterations using $\Delta(\tilde{\mu})$, whereas it uses 36 iterations with the error bound $\Delta_0(\tilde{\mu})$. This is mainly because $\Delta_0(\tilde{\mu})$ is not sharp in the beginning of the greedy algorithm. The decays of both $\Delta(\tilde{\mu})$ and $\Delta_0(\tilde{\mu})$ with the iterations of the greedy algorithm are shown in Fig. 13. The resulting ROM derived using $\Delta(\tilde{\mu})$ is of order $r = 72$. In Fig. 14, we compare $\Delta(\tilde{\mu})$, $\Delta_0(\tilde{\mu})$ with the corresponding true errors of $\hat{H}(\tilde{\mu})$ at 2500 samples of $\tilde{\mu}$, including the samples of s. It shows $\Delta(\tilde{\mu})$ is orders of magnitudes sharper than $\Delta_0(\tilde{\mu})$. $\texttt{eff}(\tilde{\mu}) := \Delta(\tilde{\mu})/\epsilon_{true}(\tilde{\mu})$ at all samples satisfy $0.38 < \texttt{eff} < 15$. We show the runtime of the greedy algorithms using the proposed error estimator and the error bound in [16], respectively, in Table VII. The error estimator obviously saves much computational time, especially when the original system becomes large. We present the magnitude of the transfer function of the Gyroscope model computed by the ROM using the proposed error estimator in Figure 15. The plot shows the transfer function changing with both frequency and the parameter d, when the other parameter θ is fixed as $\theta = 10^{-6}$. Since the original transfer function shows no difference from the reduced one, we only present the reduced transfer function for clarity of the plot.

E. Performance of $\Delta(\tilde{\mu})$ when $V_{r_{du}} = V_{du}$

In the end, we use the MIMO example and the parametric example to show the performance of $\Delta(\tilde{\mu})$ when $V_{r_{du}} = V_{du}$.

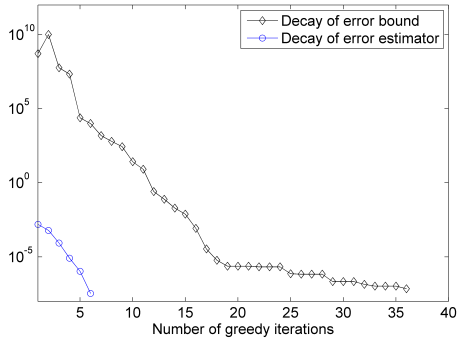

Fig. 13. Parametric example: decays of $\Delta(\tilde{\mu})$ and $\Delta_0(\tilde{\mu})$.

TABLE VII
RUNTIME (SECONDS) OF THE GREEDY ALGORITHMS

Models	Runtime-Δ (s)	Runtime-Δ_0 (s)	Speedup
CD player n =120	0.81	0.94	1.2
RLC tree $n = 6,134$	2.13	11.58	5.4
MIMO n =980	42	89	2.1
Parametric $n = 17,931$	326	3006	9.22

From Theorem 2, we know that if $V_{r_{du}} = V_{du}$, $\Delta(\tilde{\mu})$ reduces to $\Delta_1(\tilde{\mu}) = |\hat{x}_{du}^T(\tilde{\mu})r_{pr}(\tilde{\mu})|$. We plot $\Delta_1(\tilde{\mu})$ and the corresponding true error in Figures 16-17. It is clear that $\Delta_1(\tilde{\mu})$ cannot catch the true error well and is not a reliable error estimator. Therefore, the second part of $\Delta(\tilde{\mu})$, i.e. $|\hat{x}_{r_{du}}^T(\tilde{\mu})r_{pr}(\tilde{\mu})|$ does contribute to the rigorousness of $\Delta(\tilde{\mu})$.

V. CONCLUSIONS

Error estimation is still a critical issue in model order reduction. In this work, an efficient error estimator for reduced-order modeling of linear non-parametric and parametric systems is proposed. The error estimator measures the transfer function error or the output error. It is cheap to compute and outperforms other existing error bounds/estimators. The robustness of the error estimator is well demonstrated by four real-world models.

APPENDIX A
PROOF OF THEOREM 1

Proof From the definition of the transfer function, we have

$$\begin{aligned}
&|H(\tilde{\mu}) - \hat{H}(\tilde{\mu})| \\
&= |C(\mu)[Q^{-1}(\tilde{\mu})B(\mu) - V\hat{Q}^{-1}(\tilde{\mu})\hat{B}(\mu)]| \\
&= |C(\mu)Q^{-1}(\tilde{\mu})[B(\mu) - Q(\tilde{\mu})V \underbrace{\hat{Q}^{-1}(\tilde{\mu})\hat{B}(\mu)}_{z_{pr}(\tilde{\mu})}]| \\
&= |C(\mu)Q^{-1}(\tilde{\mu})r_{pr}(\tilde{\mu})|
\end{aligned} \tag{31}$$

Then,

$$\begin{aligned}
&|H(\tilde{\mu}) - \hat{H}(\tilde{\mu})| - |\hat{x}_{du}^T(\tilde{\mu})r_{pr}(\tilde{\mu})| \\
&= |C(\mu)Q^{-1}(\tilde{\mu})r_{pr}(\tilde{\mu})| - |\hat{x}_{du}^T(\tilde{\mu})r_{pr}(\tilde{\mu})| \\
&\leq |[Q^{-T}(\tilde{\mu})C^T(\mu) - \hat{x}_{du}(\tilde{\mu})]^T r_{pr}(\tilde{\mu})| \\
&= |[Q^{-T}(\tilde{\mu})(C^T(\mu) - Q^T(\tilde{\mu})\hat{x}_{du}(\tilde{\mu}))]^T r_{pr}(\tilde{\mu})| \\
&= |[\underbrace{Q^{-T}(\tilde{\mu})r_{du}(\tilde{\mu})}_{x_{r_{du}(\tilde{\mu}) \text{ in } (8)}}]^T r_{pr}(\tilde{\mu})|.
\end{aligned} \tag{32}$$

It follows

$$|H(\tilde{\mu}) - \hat{H}(\tilde{\mu})| \leq |x_{r_{du}}^T(\tilde{\mu})r_{pr}(\tilde{\mu})| + |\hat{x}_{du}^T(\tilde{\mu})r_{pr}(\tilde{\mu})|. \quad \blacksquare$$

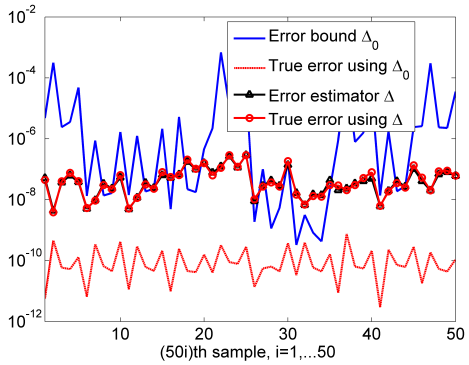

Fig. 14. Parametric example: $\Delta(\tilde{\mu}), \Delta_0(\tilde{\mu})$ vs true errors at 2500 samples of $\tilde{\mu}$.

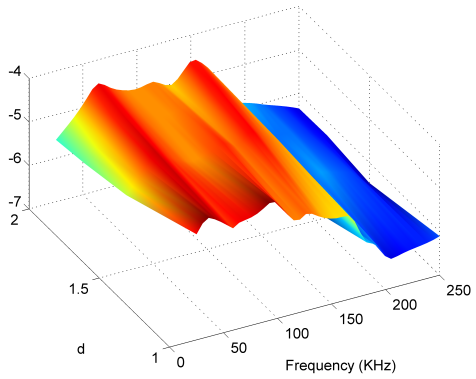

Fig. 15. Parametric model: $|\hat{H}(\mu, \jmath\omega)|$ of the ROM.

APPENDIX B
PROOF OF THEOREM 2

Proof

$$\begin{aligned} &|\hat{x}_{r_{du}}^T(\tilde{\mu}) r_{pr}(\tilde{\mu})| \\ &= |r_{pr}^T(\tilde{\mu}) \hat{x}_{r_{du}}(\tilde{\mu})| \\ &= |r_{pr}^T(\tilde{\mu}) V_{r_{du}} z_{r_{du}}(\tilde{\mu})| \\ &= |r_{pr}^T(\tilde{\mu}) V_{r_{du}} \tilde{Q}^{-1}(\tilde{\mu}) \tilde{r}_{du}(\tilde{\mu})| \\ &= |r_{pr}^T(\tilde{\mu}) V_{r_{du}} \tilde{Q}^{-1}(\tilde{\mu}) V_{r_{du}}^T r_{du}(\tilde{\mu})|. \end{aligned} \tag{33}$$

If $V_{r_{du}} = V_{du}$, then

$$\begin{aligned} V_{r_{du}}^T r_{du} = \; & V_{du}^T r_{du} \\ = \; & V_{du}^T (C^T(\mu) - Q^T(\tilde{\mu}) V_{du} z_{du}(\tilde{\mu})) \\ = \; & V_{du}^T C^T(\mu) - V_{du}^T Q^T(\tilde{\mu}) V_{du} z_{du}(\tilde{\mu})) \\ = \; & \hat{C}_{du}(\mu) - \hat{Q}_{du}(\tilde{\mu}) z_{du}(\tilde{\mu}) \\ = \; & 0. \end{aligned}$$

Substituting $V_{r_{du}}^T r_{du} = 0$ into the last equality of (33) results in $|\hat{x}_{r_{du}}^T(\tilde{\mu}) r_{pr}(\tilde{\mu})| = 0$. ∎

APPENDIX C
PROOF OF THEOREM 3

Proof From the first and last equality in (32), we obtain

$$\begin{aligned} &|H(\tilde{\mu}) - \hat{H}(\tilde{\mu})| \\ &\leq |Q^{-T}(\tilde{\mu}) r_{du}(\tilde{\mu}) r_{pr}(\tilde{\mu})| + |\hat{x}_{du}^T(\tilde{\mu}) r_{pr}(\tilde{\mu})| \\ &\leq \|Q^{-T}(\tilde{\mu})\|_2 \|r_{du}(\tilde{\mu})\|_2 \|r_{pr}(\tilde{\mu})\|_2 + \|\hat{x}_{du}^T(\tilde{\mu}) r_{pr}(\tilde{\mu})\|_2 \\ &= \|r_{du}(\tilde{\mu})\|_2 \|r_{pr}(\tilde{\mu})\|_2 / \sigma_{min}(Q(\tilde{\mu})) + \|\hat{x}_{du}^T(\tilde{\mu}) r_{pr}(\tilde{\mu})\|_2 \end{aligned} \tag{34}$$

∎

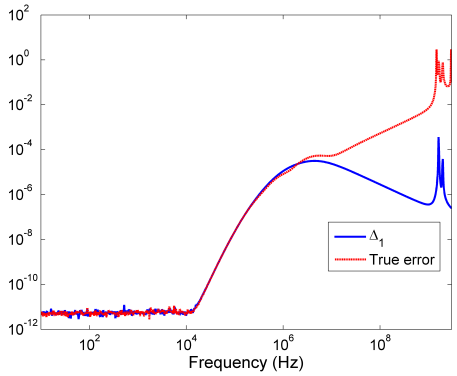

Fig. 16. MIMO example: $\Delta_1(\tilde{\mu})$ vs true errors at 900 samples of $\tilde{\mu}$.

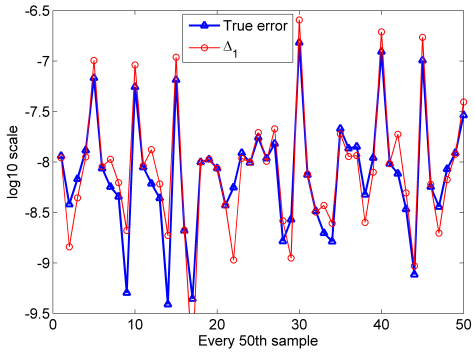

Fig. 17. Parametric example: $\Delta_1(\tilde{\mu})$ vs true errors at 2500 samples of $\tilde{\mu}$.

REFERENCES

[1] A. Hochman, J. F. Villena, A. G. Polimeridis, L. M. Silveira, J. K. White, and L. Daniel, "Reduced-order models for electromagnetic scattering problems problems," *IEEE Trans. Antennas Propag.*, vol. 62, no. 6, pp. 3150–3162, 2011.

[2] M. W. Hess and P. Benner, "Fast evaluation of time-harmonic Maxwell's equations using the reduced basis method," *IEEE Trans. Microw. Theory Techn.*, vol. 61, no. 6, pp. 2265–2274, 2013.

[3] M. W. Hess, S. Grundel, and P. Benner, "Estimating the inf-sup constant in reduced basis methods for time-harmonic Maxwell's equations," *IEEE Trans. Microw. Theory Techn.*, vol. 63, no. 11, pp. 3549–3557, 2015.

[4] M. Rewienski, A. Lamecki, and M. Mrozowski, "Greedy multipoint model-order reduction technique for fast computation of scattering parameters of electromagnetic systems," *IEEE Trans. Microw. Theory Techn.*, vol. 64, no. 6, pp. 1681–1693, 2016.

[5] M. Czarniewska, G. Fotyga, A. Lamecki, and M. Mrozowski, "Parametrized local reduced-order models with compressed projection basis for fast parameter-dependent finite-element analysis," *IEEE Trans. Microw. Theory Techn.*, vol. 66, no. 8, pp. 3656–3667, 2018.

[6] V. de la Rubia and M. Mrozowski, "A compact basis for reliable fast frequency sweep via the reduced-basis method," *IEEE Trans. Microw. Theory Techn.*, vol. 66, no. 10, pp. 4367–4382, 2018.

[7] A. Odabasioglu, M. Celik, and L. T. Pileggi, "PRIMA: passive reduced-order interconnect macromodeling algorithm," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 17, no. 8, pp. 645–654, 1998.

[8] T. Wolf, H. Panzer, and B. Lohmann, "Gramian-based error bound in model reduction by Krylov subspace methods," in *Proc. 18th IFAC World Cong.*, vol. 44, no. 1, 2011, pp. 3587–3592.

[9] L. Feng, J. G. Korvink, and P. Benner, "A fully adaptive scheme for model order reduction based on moment-matching," *IEEE Trans. Compon. Packag. Manuf. Technol.*, vol. 5, no. 12, pp. 1872–1884, 2015.

[10] L. Daniel, O. C. Siong, L. S. Chay, K. H. Lee, and J. White, "A multiparameter moment-matching model-reduction approach for generating geometrically parameterized interconnect performance models," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 23, no. 5, pp. 678–693, 2004.

[11] L. Feng, E. B. Rudnyi, and J. G. Korvink, "Preserving the film coefficient as a parameter in the compact thermal model for fast electro-thermal simulation," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 24, no. 12, pp. 1838–1847, 2005.
[12] L. Feng and P. Benner, "A robust algorithm for parametric model order reduction based on implicit moment matching," in *Reduced Order Methods for modeling and computational reduction, MS&A Series*, A. Quarteroni and G. Rozza, Eds. Berlin, Heidelberg, New York: Springer-Verlag, 2014, vol. 9, ch. 6, pp. 159–186.
[13] J. S. Hesthaven, G. Rozza, and B. Stamm, *Certified Reduced Basis Methods for Parametrized Partial Differential Equations*, ser. SpringerBriefs in Mathematics. Cham: Springer, 2016.
[14] G. Rozza, D. B. P. Huynh, and A. T. Patera, "Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations," *Arch. Comput. Methods Eng.*, vol. 15, no. 3, pp. 229–275, 2008.
[15] B. Haasdonk and M. Ohlberger, "Reduced basis method for finite volume approximations of parametrized linear evolution equations," *ESAIM: Math. Model. Numer. Anal.*, vol. 42, no. 2, pp. 277–302, 2008.
[16] L. Feng, A. C. Antoulas, and P. Benner, "Some a posteriori error bounds for reduced order modelling of (non-)parametrized linear systems," *ESAIM: M2AN*, vol. 51, pp. 2127–2158, 2017.
[17] L. Feng and P. Benner, "Efficient error estimator for model order reduction of linear parametric systems," in *Proc. IEEE MTT-S Int. Microw. Symp. (IMS 2019), Boston, MA, USA, June 2-7, 2019*, pp. 346–349.
[18] P. Benner, S. Gugercin, and K. Willcox, "A survey of model reduction methods for parametric systems," *SIAM Review*, vol. 57, no. 4, pp. 483–531, 2015.
[19] U. Baur, P. Benner, B. Haasdonk, C. Himpe, I. Martini, and M. Ohlberger, "Comparison of methods for parametric model order reduction of time-dependent problems," in *Model Reduction and Approximation: Theory and Algorithms*, P. Benner, A. Cohen, M. Ohlberger, and K. Willcox, Eds. SIAM, 2017, pp. 377–407.
[20] W. Draijer, M. Steinbuch, and O. Bosgra, "Adaptive control of the radial servo system of a compact disc player," *Automatica*, vol. 28, no. 3, pp. 455–462, 1992.
[21] Y. Chahlaoui and P. V. Dooren, "Benchmark examples for model reduction of linear time-invariant dynamical systems," in *Dimension Reduction of Large-Scale Systems*, ser. Lecture Notes in Computational Science and Engineering, P. Benner, D. C. Sorensen, and V. Mehrmann, Eds. Springer Berlin Heidelberg, 2005, vol. 45, pp. 379–392.
[22] A. Paul-Dubois-Taine and D. Amsallem, "An adaptive and efficient greedy procedure for the optimal training of parametric reduced-order models," *Internat. J. Numer. Methods Eng.*, vol. 102, pp. 1262–1292, 2015.
[23] J. S. Hesthaven, B. Stamm, and S. Zhang, "Efficient greedy algorithms for high-dimensional parameter spaces with applications to empirical interpolation and reduced basis methods," *ESAIM: Math. Model. Numer. Anal.*
[24] B. Salimbahrami, R. Eid, and B. Lohmann, "Model reduction by second order Krylov subspaces: extensions, stability and proportional damping," in *Proc. IEEE Conf. Comput. Aided Control Syst. Design*. Springer International Publishing, 2006, pp. 2997–3002.

Lihong Feng Lihong Feng received a Ph.D. in Computational Mathematics from Fudan University, Shanghai, in 2002. She held PostDoc positions at State Key Lab of ASIC & System, Fudan University, University of Freiburg from 2003-2004. She was a lecturer at the research school of Microelectonics, Fudan University from 2005-2006 and received an Alexander-von-Humboldt Fellowship for the period 2007–2009 at the host institute TU Chemnitz. Since 2010, she is a senior scientist at the Max Planck Institute for Dynamics of Complex Technical Systems in Magdeburg, and a team leader in the CSC group headed by Peter Benner. Her research interests include model order reduction, fast simulation of complex differential equations arising from engineering applications such as fluid dynamics, chemical engineering, MEMS simulation and integrated circuit simulation; numerical analysis; and scientific computing.

Peter Benner Peter Benner received the "Diplom" in Mathematics from the RWTH Aachen, Germany, in 1993. From 1993 to 1997, he worked on his Ph.D. at the University of Kansas, Lawrence, USA, and the TU Chemnitz–Zwickau, Germany, where he received his Ph.D. in February 1997. In 2001, he finished his Habilitation in Mathematics at the University of Bremen, where he held an assistant professor position from 1997 to 2001. After spending a term as a visiting associate professor at TU Hamburg–Harburg, Germany, he was a lecturer in Mathematics at TU Berlin from 2001–03. Since 2003, he is professor for Mathematics in Industry and Technology at Chemnitz University of Technology. In 2010, he was appointed as one of the four directors of the Max Planck Institute for Dynamics of Complex Technical Systems in Magdeburg. Since 2011, he is also honorary professor at the Otto-von-Guericke University of Magdeburg. He is Distinguished Professor at Shanghai University in 2015.

His research interests are in the areas of scientific computing, numerical mathematics, systems theory, optimal control. A particular emphasis has been on applying methods from numerical linear algebra and matrix theory in systems and control theory. Recent research focuses on numerical methods for optimal control of systems modeled by evolution equations (PDEs, DAEs, SPDEs), model order reduction, preconditioning in optimal control and UQ problems, and Krylov subspace methods for structured or quadratic eigenproblems. Research in all these areas is accompanied by the development of algorithms and mathematical software suitable for modern and high-performance computer architectures.