SIAM J. Sci. Comput.
Vol. 37, No. 6, pp. B910–B936

AN EFFICIENT OUTPUT ERROR ESTIMATION FOR MODEL ORDER REDUCTION OF PARAMETRIZED EVOLUTION EQUATIONS*

YONGJIN ZHANG†, LIHONG FENG‡, SUZHOU LI†, AND PETER BENNER†

Abstract. In this work we present an a posteriori output error bound for model order reduction of parametrized evolution equations. With the help of the dual system and a simple representation of the relationship between the field variable error and the residual of the primal system, a sharp output error bound is derived. Such an error bound successfully avoids the accumulation of the residual over time, which is a common drawback in the existing error estimations for time-stepping schemes. An estimation needs to be performed for practical computation of the error bound, and as a result, the output error bound reduces to an output error estimation. The proposed error estimation is applied to four kinds of problems. The first one is a linear convection-diffusion equation, which is used to compare the performance of the new error estimation and an existing primal-dual error bound. The second one is the unsteady viscous Burgers' equation, an academic benchmark of nonlinear evolution equations in fluid dynamics often used as a first test case to validate nonlinear model order reduction methods. The other two problems arise from chromatographic separation processes. Numerical experiments demonstrate the performance and efficiency of the proposed error estimation. Furthermore, optimization based on the resulting reduced-order models is successful in terms of accuracy and runtime for obtaining the optimal solution.

Key words. error estimation, model order reduction, output error bound, empirical interpolation, dual system, evolution equations, optimization, chromatography

AMS subject classifications. 65M15, 65D15, 35L90, 35K90

DOI. 10.1137/140998603

1. Introduction.

Numerical simulation of large-scale systems is challenging, especially when this task needs to be repeated many times under parameter variations, e.g., in the context of optimization, control, and parameter estimation. Model order reduction (MOR) is a useful technique for constructing a low-cost, simulation efficient surrogate reduced-order model (ROM), which can reproduce the dominant dynamics or the input-output response of the original large-scale system, at a compromise with accuracy to an acceptable extent. To generate a ROM, an efficient a posteriori error estimation is crucial because it enables the generation to be reliable and automatic. Rigorous, sharp, and cheaply computable are the desired properties of an efficient error estimation.

In the past years, many efforts have been devoted to the study of a posteriori error estimation for either the field variable (the solution to the underlying system) or the output of interest, which is usually expressed as a functional of the field variable. For example, research on an a posteriori error estimation for the reduced basis (RB) method started with [26, 28] and has been followed by many others [13, 14, 15, 16, 17, 18, 25, 27, 29, 32]. More recently, the space-time RB method was introduced

*Submitted to the journal's Computational Methods in Science and Engineering section December 3, 2014; accepted for publication (in revised form) September 22, 2015; published electronically November 24, 2015.
http://www.siam.org/journals/sisc/37-6/99860.html

†Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstrasse 1, D-39106 Magdeburg, Germany (zhangy@mpi-magdeburg.mpg.de, suzhou@mpi-magdeburg.mpg.de, benner@mpi-magdeburg.mpg.de).

‡Corresponding author. Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstrasse 1, D-39106 Magdeburg, Germany (feng@mpi-magdeburg.mpg.de).

for linear or quadratically nonlinear parabolic problems [34, 37, 38, 39]. Notably, these error estimations are all derived in the functional space in the framework of the finite element (FE) discretization except for [17]. In the FE discretization framework, the weak form of the partial differential equation (PDE) is used to derive the error bound, while the error bound in [17] is derived in the framework of the finite volume discretization for error estimation of the field variables.

In this paper, we propose an efficient output error estimation for projection-based MOR methods applied to parametrized nonlinear evolution problems. For (nonlinear) evolution problems, time-stepping schemes are often used to solve them [24], and error estimations for projection-based MOR methods have been studied in recent years; see, e.g., [13, 17, 40]. The error estimator, however, may lose sharpness when a large number of time steps are needed, because the error estimator is actually a summation of the error over the previous time steps. To circumvent this problem, we introduce a suitable dual system at each time instance in the evolution process associated with the primal system, i.e., the original system. The output error for the primal system can thus be estimated sharply and efficiently with the help of the dual system and a simple representation of the relationship between the residual and the error of the field variable.

Actually, an a posteriori output error bound for the RB method using the primal-dual approach can be found in [16]. However, the derived output error bound is suitable only for linear evolution equations. From the numerical comparison in section 6, the proposed error estimation outperforms the error bound in [16] for a linear evolution system.

The aforementioned error bounds introduced in [34, 37, 38, 39] are based on space-time variational formulation. Notably, the space-time error bounds are derived for the space-time model rather than the standard model addressed in many other MOR papers. The space-time model is obtained from the parabolic equations by first discretizing in space and then integrating in time. As a result, the state vector (unknown vector) of the space-time model is different from the state vector for the standard model derived by only discretization in space, and time-stepping in time. Roughly speaking, the solution vector of the space-time model can be considered as a long vector $u_{\mathrm{st}} \in \mathbb{R}^{\mathcal{N}\cdot K}$ including the spatial discretized vector at all the time steps, where $\mathcal{N}$ is the number of spatial grids and K is the number of time steps. The corresponding error bound measures the error of this long vector computed by the corresponding ROM. Our error bound is defined for the solution vector $u \in \mathbb{R}^{\mathcal{N}}$ of the spatially discretized model at each time step t^k, $k = 1, \dots, K$. The errors measured by the error bounds are different. Finally, the error bounds in those papers are valid only for linear and at most quadratically nonlinear systems. The error bounds are limited to Petrov–Galerkin discretization (in space), and an inf-sup constant (or its lower bound) for the corresponding variational (weak) form must be available. Our proposed error bound is valid for general linear and nonlinear systems (given Lipschitz continuity of the nonlinear term) and is applicable to any discretization approach.

The idea for the proposed error bound originates from the recent study in [10, 11], where some error bounds are derived for linear time-invariant systems. The main difference of the proposed error estimation from that in [10, 11] is that the new error estimation is derived directly in the time domain and is exactly designed for the output in the time domain. It is particularly useful for snapshot-based MOR methods, e.g., the RB method [17, 27, 40] and the proper orthogonal decomposition (POD) method [6, 8, 35, 36]. It is valid for nonlinear parametric systems, whereas the error bound in [10, 11] is an error estimation for the transfer function of the ROM, so that it is used for linear parametric systems. In other words, it is an error estimation for the output in the frequency domain, which is well suited for the frequency domain MOR methods, e.g., the Krylov subspace method [2, 4, 12].

The proposed error estimation is applied to four different evolution problems. The first one is a linear convection-diffusion equation from [13], describing the movement of fluids and other transport phenomena. The second one is the unsteady viscous Burgers' equation, which is used as the first test case to illustrate the applicability of the proposed error estimation to nonlinear evolution problems. The last two real-life models are described by parametrized coupled systems of convection-diffusion equations. They arise from chromatographic separation processes in chemical engineering. One is batch chromatography with bi-Langmuir isotherm equations, which is a nonlinear parametric evolution problem. The other is continuous simulated moving bed (SMB) chromatography with linear isotherm equations. It is a linear parametrized periodic switching system. Moreover, these systems require a long-time integration process. The resulting ROMs are further employed to solve the underlying optimization problems.

The paper is organized as follows. Section 2 reviews projection-based MOR for parametrized nonlinear evolution problems. Existing related error estimations are reviewed in section 3, and an output error estimation is derived in section 4. Section 5 discusses the POD-Greedy algorithm [17] used in RB methods, which is employed for constructing the ROMs in section 6. Numerical examples are given in section 6. Section 7 concludes the paper.

2. MOR of parametrized evolution equations.

In this paper, we consider a parametrized evolution problem defined over the spatial domain $\Omega \subset \mathbb{R}^d$ $(d = 1, 2, 3)$ and the parameter domain $\mathcal{P} \subset \mathbb{R}^p$,

$$\partial_t u(t, x; \mu) + \mathcal{L}(\mu)[u(t, x; \mu)] = 0, \quad t \in (0, T], \quad x \in \Omega, \quad \mu \in \mathcal{P}, \tag{2.1}$$

where $\mathcal{L}(\mu)[\cdot]$ is a spatial differential operator. For discretization, let $0 = t^0 < t^1 < \cdots < t^K = T$ be $K+1$ time instants in the time interval $[0, T]$, and $\mathcal{W}^{\mathcal{N}} \subset L^2(\Omega)$ be an $\mathcal{N}$-dimensional discrete space in which an approximate numerical solution to (2.1) is sought. Given $\mu \in \mathcal{P}$ with suitable initial and boundary conditions, the numerical solution $u^n(\mu)$ at time $t = t^n$ can be obtained by using suitable numerical methods, e.g., the finite volume method. Assume that $u^n(\mu) \in \mathcal{W}^{\mathcal{N}}$ satisfies the form

$$A_\mu^{(n)} u^{n+1}(\mu) = B_\mu^{(n)} u^n(\mu) + g(u^n(\mu); \mu), \tag{2.2}$$

where $A_\mu^{(n)}$, $B_\mu^{(n)} \in \mathbb{R}^{\mathcal{N} \times \mathcal{N}}$ are the coefficient matrices at the time instance t^n, and $g(\cdot; \mu)$ is a nonlinear operator w.r.t. $u^n(\mu)$ and/or nonaffine w.r.t. the parameter μ. The superscript (n) and the subscript μ in $A_\mu^{(n)}$ and $B_\mu^{(n)}$ indicate the dependency on time and the parameter, respectively. For MOR, the dimension $\mathcal{N}$ is usually large, which implies that the numerical solution $u^n(\mu)$ is a faithful approximation and is often called the "true" solution. The resulting large-scale system in (2.2) is called the full order model (FOM).

Solving such a FOM repeatedly under parameter variations is time-consuming or even prohibitive in a multiquery context, e.g., optimization or real-time control. Besides the available computing resources, MOR has been developed as a useful tool to handle this kind of problem and plays an important role in an efficient solution process for parametric systems. In the following subsections, we address the idea of MOR based on projection and the issue of simulating the ROM.

2.1. Projection-based MOR. In this paper, we focus on projection-based MOR methods. The fundamental assumption is that the solution to the parametrized systems, $u(\mu)$, resides in a lower dimensional subspace $\mathcal{V}^N \subset \mathcal{W}^{\mathcal{N}}$, i.e., $u(\mu)$ can be well approximated by a properly chosen basis of the subspace. For all the projection-based MOR methods, a right projection matrix $V \in \mathbb{R}^{\mathcal{N}\times N}$, whose columns span a basis of the subspace where $u(\mu)$ can be well represented, is computed. A left projection matrix $W \in \mathbb{R}^{\mathcal{N}\times N}$ is constructed based on proper approximation principles. The ROM is obtained by using the approximation $u^n(\mu) \approx \hat{u}^n(\mu) := Va^n(\mu)$ and employing Petrov–Galerkin projection with W,

$$\hat{A}_\mu^{(n)} a^{n+1}(\mu) = \hat{B}_\mu^{(n)} a^n(\mu) + W^T g(Va^n(\mu); \mu), \tag{2.3}$$

where $\hat{A}_\mu^{(n)} = W^T A_\mu^{(n)} V$, $\hat{B}_\mu^{(n)} = W^T B_\mu^{(n)} V$, and $a^n(\mu) \in \mathbb{R}^N$ is the vector of unknowns of the ROM.

Notably, the number of degrees of freedom of the ROM in (2.3) is usually much less than that of the FOM in (2.2), i.e., $N \ll \mathcal{N}$. The goal of MOR is that the ROM is much cheaper to solve compared to the FOM. This is not necessarily achieved by (2.3); it is required that the evaluation of $\hat{A}_\mu^{(n)}, \hat{B}_\mu^{(n)}$, and $W^T g(Va^n(\mu); \mu)$ is done without resorting to the full order dimension $\mathcal{N}$. For this, additional techniques may be necessary, as described in the following.

2.2. Simulation of the ROM. The goal of MOR is to provide a fast simulation stage, where for any given parameter μ the output response can be obtained rapidly based on the ROM. Particularly, in the RB method, an offline-online decomposition strategy is often employed to attain this goal. Similar to related studies, assume that the matrices $A_\mu^{(n)}$ and $B_\mu^{(n)}$ in (2.2) can be written in a separable way, the so-called affine form, i.e.,

$$A_\mu^{(n)} = \sum_{j=1}^{n_a} \xi_{\mu,j}^n A_j, \qquad B_\mu^{(n)} = \sum_{k=1}^{n_b} \zeta_{\mu,k}^n B_k,$$

where A_j, B_k are constant matrices, and $\xi_{\mu,j}^n$, $\zeta_{\mu,k}^n$ are the corresponding time and parameter dependent scalar coefficients. Note that the numbers n_a and n_b are desired to be small. Then

$$\hat{A}_\mu^{(n)} = W^T A_\mu^{(n)} V = \sum_{j=1}^{n_a} \xi_{\mu,j}^n \hat{A}_j, \qquad \hat{B}_\mu^{(n)} = W^T B_\mu^{(n)} V = \sum_{k=1}^{n_b} \zeta_{\mu,k}^n \hat{B}_k,$$

where $\hat{A}_j = W^T A_j V$ and $\hat{B}_k = W^T B_k V$, $j = 1, \dots, n_a$, $k = 1, \dots, n_b$. Notice that once the projection matrices V and W are obtained, $\hat{A}_j$ and $\hat{B}_k$ can be precomputed, and in turn the evaluation of $\hat{A}_\mu^{(n)}$ and $\hat{B}_\mu^{(n)}$ at μ is independent of the full dimension $\mathcal{N}$. However, the computation of the last term of (2.3), $W^T g(Va^n(\mu); \mu)$, cannot be done analogously because of the nonlinearity or nonaffinity of g. To achieve an efficient offline-online computation, empirical (operator) interpolation [3, 9] or the discrete empirical interpolation (EI) method [8] can be employed. Similar treatments can be found in [14, 15]. More precisely, a parameter-independent basis $S \in \mathbb{R}^{\mathcal{N}\times M}$ $(M \ll \mathcal{N})$ is precomputed based on snapshots of the nonlinear function evaluations at a set of properly selected parameter samples. Then an affine approximation is defined by an interpolation operator $\mathcal{I}_M : \mathbb{R}^{\mathcal{N}} \to \mathbb{R}^{\mathcal{N}}$, i.e., $g(\hat{u}^n(\mu); \mu) \approx \hat{g}^n(\mu) := \mathcal{I}_M[g(\hat{u}^n(\mu); \mu)] = S\beta^n(\mu)$, where $\beta^n(\mu) := \beta(a^n(\mu); \mu) \in \mathbb{R}^M$ is the corresponding vector of coefficients. The coefficient vector $\beta^n(\mu)$ can be determined by an enforced interpolation condition that $\hat{g}^n(\mu)$ interpolates $g(\hat{u}^n(\mu);\mu)$ at a set of properly selected components. For more details, please refer to, e.g., [40]. As a result, a low dimensional ROM is obtained as below,

$$\hat{A}_\mu^{(n)} a^{n+1}(\mu) = \hat{B}_\mu^{(n)} a^n(\mu) + \hat{G}\beta^n(\mu), \tag{2.4}$$

where $\hat{G} = W^T S$ is precomputed. With this ROM (2.4), the approximation of the field variable and/or the output can be obtained rapidly.

In what follows, the norm $\|\cdot\| : \mathbb{R}^{\mathcal{N}} \to \mathbb{R}$ for a vector v is defined as

$$\|v\| := \sqrt{v^T H v},$$

where H is a properly chosen symmetric positive definite matrix. When H is the identity matrix, it is the standard 2-norm. The matrix norm is defined as the corresponding induced norm. In the next section, we review an a posteriori output error bound and point out its limitations.

3. An output error bound based on the residual.

A common technique to derive an error estimation for the projection-based MOR method is based on the residual [9, 13, 17, 27, 40]. Motivated by the error estimation for the field variable in [9], an error estimation for the field variable in the vector space and a corresponding output error bound were proposed in [40], where the nonlinear term $g(\cdot;\cdot)$ was tackled by using EI [3]. By defining the residual

$$r^{n+1}(\mu) := B_\mu^{(n)} \hat{u}^n(\mu) + \mathcal{I}_M[g(\hat{u}^n(\mu);\mu)] - A_\mu^{(n)} \hat{u}^{n+1}(\mu) \tag{3.1}$$

for the ROM in (2.4), the error estimations in [40] are summarized as below.

PROPOSITION 3.1. *Assume that for all $\mu \in \mathcal{P}$ the operator $g(\cdot;\mu)\colon \mathcal{W}^{\mathcal{N}} \to \mathbb{R}^{\mathcal{N}}$ is Lipschitz continuous w.r.t. the first argument, i.e., there exists a positive constant L_g, such that*

$$\|g(u_1;\mu) - g(u_2;\mu)\| \le L_g \|u_1 - u_2\|, \quad u_1, u_2 \in \mathcal{W}^{\mathcal{N}},$$

and that the interpolation of g is "exact" with a certain dimension of $S = [s_1, \dots, s_{M+M'}]$, i.e.,

$$\mathcal{I}_{M+M'}[g(\hat{u}^n(\mu);\mu)] = \sum_{m=1}^{M+M'} s_m \beta_m^n(\mu) = g(\hat{u}^n(\mu);\mu).$$

Assume further that for all $\mu \in \mathcal{P}$, the initial projection error is vanishing: $e^0(\mu) = 0$, and the output of interest $y(u^n(\mu))$ is given as

$$y(u^n(\mu)) = P u^n(\mu), \tag{3.2}$$

where $P \in \mathbb{R}^{N_\text{o} \times \mathcal{N}}$ is a constant matrix. Then the errors for the field variable $e^n(\mu) := u^n(\mu) - \hat{u}^n(\mu)$ and the output error $e_\text{O}^n(\mu) := y(u^n(\mu)) - y(\hat{u}^n(\mu))$ respectively satisfy

$$\|e^n(\mu)\| \le \eta_{N,M}^n(\mu) := R_\mu^{n-1} + \sum_{k=0}^{n-2} \left(\prod_{j=k+1}^{n-1} \mathtt{G}_{\text{F},\mu}^{(j)} \right) R_\mu^k, \; n = 1, \dots, K, \tag{3.3}$$

$$\begin{aligned} \|e_\text{O}^{n+1}(\mu)\| &\le \tilde{\eta}_{N,M}^{n+1}(\mu) \\ &:= \mathtt{G}_{\text{O},\mu}^{(n)} \eta_{N,M}^n(\mu) + \|P(A_\mu^{(n)})^{-1}\| \epsilon_\text{EI}^n(\mu) + \|P\| \|(A_\mu^{(n)})^{-1} r^{n+1}(\mu)\|, \end{aligned} \tag{3.4}$$

where

$$R_\mu^k = \|(A_\mu^{(k)})^{-1}\|\epsilon_{\mathrm{EI}}^k(\mu) + \|(A_\mu^{(k)})^{-1}r^{k+1}(\mu)\|, \quad k = 0,\dots,n-1,$$
$$\mathtt{G}_{\mathrm{F},\mu}^{(j)} = \|(A_\mu^{(j)})^{-1}B_\mu^{(j)}\| + L_g\|(A_\mu^{(j)})^{-1}\|, \quad j = k+1,\dots,n-1,$$
$$\mathtt{G}_{\mathrm{O},\mu}^{(n)} = \|P(A_\mu^{(n)})^{-1}B_\mu^{(n)}\| + L_g\|P(A_\mu^{(n)})^{-1}\|,$$

and

$$\epsilon_{\mathrm{EI}}^n(\mu) = \sum_{m=M+1}^{M+M'} \|s_m\beta_m^n(\mu)\| \tag{3.5}$$

is the error due to the EI, $n = 0,\dots,K-1$.

Notice that the error bound for the field variable $\eta_{N,M}^n(\mu)$ is involved in the output error bound $\tilde{\eta}_{N,M}^{n+1}(\mu)$. Moreover, the former is a summation of the residual and the error $\epsilon_{\mathrm{EI}}^n(\mu)$ over all the previous time steps. This implies that both error bounds are accumulated over time. As a result, they may lose sharpness when a large number of time steps are needed, e.g., in the simulation of batch chromatography [40]. The same phenomenon also exists in the error estimation in [9]. Similar observations are reported in [25]. To circumvent the problem, we propose a new output error bound for the ROM in the next section.

4. An a posteriori output error bound using dual systems.

In this section, we derive a new bound for the output error in the time domain by defining and using the dual systems. Assume that the FOM from the spatial and temporal discretization of the PDEs can be written as

$$A_\mu^{(n)}u^{n+1}(\mu) = b(u^n(\mu);\mu), \tag{4.1}$$

where $A_\mu^{(n)}$ is assumed to be nonsingular for all $\mu \in \mathcal{P}$, $u^n(\mu) \in \mathcal{W}^{\mathcal{N}}$ is the numerical solution at time $t = t^n$, and $b(\cdot;\mu) : \mathcal{W}^{\mathcal{N}} \to \mathbb{R}^{\mathcal{N}}$ can be nonlinear (or linear) w.r.t. the first argument and/or nonaffine w.r.t. the parameter $\mu \in \mathcal{P}$, e.g., the right-hand side of the equation in (2.2). The output of interest is expressed as in (3.2). Here, we temporally assume $N_{\mathrm{o}} = 1$ for simplicity. The extension to the multiple output case is possible; see Remark 4.8.

To derive an efficient output error estimation, at each time step, we denote the original system as the primal system

$$\begin{cases} A_\mu^{(n)}u^{n+1}(\mu) = b(u^n(\mu);\mu), \\ \quad\quad y^{n+1}(\mu) = Pu^{n+1}(\mu) \end{cases} \tag{4.2}$$

and introduce a corresponding dual system as follows:

$$(A_\mu^{(n)})^T u_{\mathrm{du}}^{n+1}(\mu) = -P^T. \tag{4.3}$$

Assume that $(V_{\mathrm{pr}}, W_{\mathrm{pr}})$ and $(V_{\mathrm{du}}, W_{\mathrm{du}})$ are the projection matrix pairs for MOR of the primal and dual systems, respectively. Using Petrov–Galerkin projection, we have the ROMs for the primal and the dual systems, respectively,

$$\begin{cases} W_{\mathrm{pr}}^T A_\mu^{(n)}\hat{u}^{n+1}(\mu) = W_{\mathrm{pr}}^T b(\hat{u}^n(\mu);\mu), \\ \quad\quad \hat{y}^{n+1}(\mu) = P\hat{u}^{n+1}(\mu), \end{cases} \tag{4.4}$$

$$W_{\mathrm{du}}^T (A_\mu^{(n)})^T \hat{u}_{\mathrm{du}}^{n+1}(\mu) = -W_{\mathrm{du}}^T P^T, \tag{4.5}$$

where $\hat{u}^n(\mu) = V_{\text{pr}} a_{\text{pr}}^n(\mu)$, $\hat{u}_{\text{du}}^n(\mu) = V_{\text{du}} a_{\text{du}}^n(\mu)$ are the approximations to $u^n(\mu)$ and $u_{\text{du}}^n(\mu)$, respectively. The vectors $a_{\text{pr}}^n(\mu)$ and $a_{\text{du}}^n(\mu)$ are the unknowns of the reduced primal and the reduced dual systems in (4.4) and (4.5), respectively. The residuals for both systems read

$$r_{\text{pr}}^{n+1} := r_{\text{pr}}^{n+1}(\mu) = b(\hat{u}^n(\mu);\mu) - A_\mu^{(n)} \hat{u}^{n+1}(\mu), \tag{4.6}$$

$$r_{\text{du}}^{n+1} := r_{\text{du}}^{n+1}(\mu) = -P^T - (A_\mu^{(n)})^T \hat{u}_{\text{du}}^{n+1}(\mu), \tag{4.7}$$

respectively. Define an auxiliary vector

$$\tilde{r}_{\text{pr}}^{n+1} := b(u^n(\mu);\mu) - A_\mu^{(n)} \hat{u}^{n+1}(\mu) = A_\mu^{(n)} u^{n+1}(\mu) - A_\mu^{(n)} \hat{u}^{n+1}(\mu). \tag{4.8}$$

Notice that the only difference of $\tilde{r}_{\text{pr}}^{n+1}$ from r_{pr}^{n+1} is that $b(\hat{u}^n(\mu);\mu)$ in (4.6) is replaced by $b(u^n(\mu);\mu)$ in (4.8), so that we have a direct relation between $\tilde{r}_{\text{pr}}^{n+1}$ and $u^{n+1}(\mu) - \hat{u}^{n+1}(\mu)$, the error of the approximate solution. This relation will aid the derivation of the error bound in Theorem 4.1. For simplicity, we define

$$\Phi_\mu^{n+1} := \|(A_\mu^{(n)})^{-T}\| \|r_{\text{du}}^{n+1}\| + \|\hat{u}_{\text{du}}^{n+1}(\mu)\|. \tag{4.9}$$

THEOREM 4.1. *For the systems* (4.2) *and* (4.4), *assume that* $A_\mu^{(n)}$ *is invertible for any* $\mu \in \mathcal{P}$. *Then the output error* $e_{\text{O}}^{n+1}(\mu) = y^{n+1}(\mu) - \hat{y}^{n+1}(\mu)$ *at the time instance* t^{n+1} *satisfies*

$$\|e_{\text{O}}^{n+1}(\mu)\| \le \Phi_\mu^{n+1} \|\tilde{r}_{\text{pr}}^{n+1}\|, \quad n = 0, \dots, K-1. \tag{4.10}$$

Proof. Left-multiplying both sides of (4.3) by $\left(u^{n+1}(\mu) - \hat{u}^{n+1}(\mu)\right)^T$, we have

$$\left(u^{n+1}(\mu) - \hat{u}^{n+1}(\mu)\right)^T (A_\mu^{(n)})^T u_{\text{du}}^{n+1}(\mu) = -\left(u^{n+1}(\mu) - \hat{u}^{n+1}(\mu)\right)^T P^T.$$

Transposing this equation, we obtain

$$\left(u_{\text{du}}^{n+1}(\mu)\right)^T A_\mu^{(n)} \left(u^{n+1}(\mu) - \hat{u}^{n+1}(\mu)\right) = -P\left(u^{n+1}(\mu) - \hat{u}^{n+1}(\mu)\right). \tag{4.11}$$

By the definition of $\tilde{r}_{\text{pr}}^{n+1}$, we have

$$\tilde{r}_{\text{pr}}^{n+1} = A_\mu^{(n)} \left(u^{n+1}(\mu) - \hat{u}^{n+1}(\mu)\right). \tag{4.12}$$

Left-multiplying both sides of (4.12) by $\left(u_{\text{du}}^{n+1}(\mu)\right)^T$ yields

$$\left(u_{\text{du}}^{n+1}(\mu)\right)^T \tilde{r}_{\text{pr}}^{n+1} = \left(u_{\text{du}}^{n+1}(\mu)\right)^T A_\mu^{(n)} \left(u^{n+1}(\mu) - \hat{u}^{n+1}(\mu)\right). \tag{4.13}$$

Combining (4.11) and (4.13), we obtain

$$-P\left(u^{n+1}(\mu) - \hat{u}^{n+1}(\mu)\right) = \left(u_{\text{du}}^{n+1}(\mu)\right)^T \tilde{r}_{\text{pr}}^{n+1}.$$

Introducing a vector $\tilde{y}^{n+1}(\mu) = P\hat{u}^{n+1}(\mu) - \left(\hat{u}_{\text{du}}^{n+1}(\mu)\right)^T \tilde{r}_{\text{pr}}^{n+1}$, we have

$$\begin{aligned} |y^{n+1}(\mu) - \tilde{y}^{n+1}(\mu)| &= |Pu^{n+1}(\mu) - P\hat{u}^{n+1}(\mu) + \left(\hat{u}_{\text{du}}^{n+1}(\mu)\right)^T \tilde{r}_{\text{pr}}^{n+1}| \\ &= |-\left(u_{\text{du}}^{n+1}(\mu)\right)^T \tilde{r}_{\text{pr}}^{n+1} + \left(\hat{u}_{\text{du}}^{n+1}(\mu)\right)^T \tilde{r}_{\text{pr}}^{n+1}| \\ &= |-\left(u_{\text{du}}^{n+1}(\mu) - \hat{u}_{\text{du}}^{n+1}(\mu)\right)^T \tilde{r}_{\text{pr}}^{n+1}| \\ &\le \|u_{\text{du}}^{n+1}(\mu) - \hat{u}_{\text{du}}^{n+1}(\mu)\| \|\tilde{r}_{\text{pr}}^{n+1}\|. \end{aligned} \tag{4.14}$$

By the definition of the residual in (4.7) and the dual system in (4.3), we have

$$\begin{aligned} r_{\mathrm{du}}^{n+1} &= -P^T - (A_\mu^{(n)})^T \hat{u}_{\mathrm{du}}^{n+1}(\mu) \\ &= (A_\mu^{(n)})^T u_{\mathrm{du}}^{n+1}(\mu) - (A_\mu^{(n)})^T \hat{u}_{\mathrm{du}}^{n+1}(\mu) \\ &= (A_\mu^{(n)})^T \left(u_{\mathrm{du}}^{n+1}(\mu) - \hat{u}_{\mathrm{du}}^{n+1}(\mu) \right). \end{aligned}$$

Since $A_\mu^{(n)}$ is invertible, we have

$$u_{\mathrm{du}}^{n+1}(\mu) - \hat{u}_{\mathrm{du}}^{n+1}(\mu) = (A_\mu^{(n)})^{-T} r_{\mathrm{du}}^{n+1}. \tag{4.15}$$

Combining (4.14) and (4.15), we obtain

$$|y^{n+1}(\mu) - \tilde{y}^{n+1}(\mu)| \le \|(A_\mu^{(n)})^{-T} r_{\mathrm{du}}^{n+1}\| \|\tilde{r}_{\mathrm{pr}}^{n+1}\| \le \|(A_\mu^{(n)})^{-T}\| \|r_{\mathrm{du}}^{n+1}\| \|\tilde{r}_{\mathrm{pr}}^{n+1}\|.$$

Thus

$$\begin{aligned} |y^{n+1}(\mu) - \hat{y}^{n+1}(\mu)| &= |y^{n+1}(\mu) - \tilde{y}^{n+1}(\mu) - \left(\hat{u}_{\mathrm{du}}^{n+1}(\mu)\right)^T \tilde{r}_{\mathrm{pr}}^{n+1}| \\ &\le |y^{n+1}(\mu) - \tilde{y}^{n+1}(\mu)| + | \left(\hat{u}_{\mathrm{du}}^{n+1}(\mu)\right)^T \tilde{r}_{\mathrm{pr}}^{n+1}| \\ &\le \|(A_\mu^{(n)})^{-T}\| \|r_{\mathrm{du}}^{n+1}\| \|\tilde{r}_{\mathrm{pr}}^{n+1}\| + \| \left(\hat{u}_{\mathrm{du}}^{n+1}(\mu)\right)^T \| \|\tilde{r}_{\mathrm{pr}}^{n+1}\| \\ &= \Phi_\mu^{n+1} \|\tilde{r}_{\mathrm{pr}}^{n+1}\|. \qquad \square \end{aligned} \tag{4.16}$$

Notice that the error bound in (4.10) is not feasible to compute in practice, because the detailed solution $u^{n+1}(\mu)$ is involved in the evaluation of $\|\tilde{r}_{\mathrm{pr}}^{n+1}\|$. For this, defining

$$\rho^{n+1}(\mu) := \frac{\|\tilde{r}_{\mathrm{pr}}^{n+1}\|}{\|r_{\mathrm{pr}}^{n+1}\|}, \tag{4.17}$$

we have the following two corollaries, showing the existence of $\rho^{n+1}(\mu)$ by an upper bound and a lower bound, under certain assumptions. As a result, the output error bound in Theorem 4.1 becomes

$$\|e_{\mathrm{O}}^{n+1}(\mu)\| \le \Delta_N^{n+1}(\mu) := \Phi_\mu^{n+1} \rho^{n+1}(\mu) \|r_{\mathrm{pr}}^{n+1}\|. \tag{4.18}$$

COROLLARY 4.2. *Under the assumptions in Theorem* 4.1, *for the vectors* $\{\tilde{r}_{\mathrm{pr}}^n\}_{n=1}^K$, *assume that there exists a positive constant* α *such that*

$$\alpha \le \frac{\|\tilde{r}_{\mathrm{pr}}^{n+1}\|}{\|\tilde{r}_{\mathrm{pr}}^{n}\|}, \quad n = 1, \dots, K-1, \quad \mu \in \mathcal{P}. \tag{4.19}$$

Assume that for all $\mu \in \mathcal{P}$ *the operator* $b(\cdot;\mu)$ *in* (4.2) *is Lipschitz continuous w.r.t. the first argument, i.e., there exists a positive constant* L_b *such that*

$$\|b(u_1;\mu) - b(u_2;\mu)\| \le L_b \|u_1 - u_2\|, \quad u_1, u_2 \in \mathcal{W}^{\mathcal{N}}, \quad \mu \in \mathcal{P}. \tag{4.20}$$

Assume further

$$L_b < \alpha / \|(A_\mu^{(n)})^{-1}\|, \quad n = 0, \dots, K-1, \quad \mu \in \mathcal{P}. \tag{4.21}$$

Then

$$\underline{\rho}^{n+1}(\mu) \le \rho^{n+1}(\mu) \le \bar{\rho}^{n+1}(\mu), \tag{4.22}$$

where $\underline{\rho}^{n+1}(\mu) = \frac{\alpha}{\alpha + L_b\|(A_\mu^{(n-1)})^{-1}\|}$, $\bar{\rho}^{n+1}(\mu) = \frac{\alpha}{\alpha - L_b\|(A_\mu^{(n-1)})^{-1}\|}$, $n = 1, \ldots, K-1$, *and* $\mu \in \mathcal{P}$.

Proof. By the definition of the vectors $r_{\rm pr}^{n+1}$ and $\tilde{r}_{\rm pr}^{n+1}$ (in (4.6) and (4.8), respectively), and using the Lipschitz condition in (4.20), we have

$$\begin{aligned}\|r_{\rm pr}^{n+1} - \tilde{r}_{\rm pr}^{n+1}\| &= \|b(\hat{u}^n(\mu);\mu) - b(u^n(\mu);\mu)\| \\ &\le L_b\|\hat{u}^n(\mu) - u^n(\mu)\| = L_b\|(A_\mu^{(n-1)})^{-1}\tilde{r}_{\rm pr}^n\| \\ &\le L_b\|(A_\mu^{(n-1)})^{-1}\|\|\tilde{r}_{\rm pr}^n\|.\end{aligned} \tag{4.23}$$

By the inequality in (4.19), we have

$$\|\tilde{r}_{\rm pr}^n\| \le \|\tilde{r}_{\rm pr}^{n+1}\|/\alpha. \tag{4.24}$$

Substituting (4.24) into (4.23), and using the triangle inequality, we have

$$\|\tilde{r}_{\rm pr}^{n+1}\| - \|r_{\rm pr}^{n+1}\| \le \|r_{\rm pr}^{n+1} - \tilde{r}_{\rm pr}^{n+1}\| \le L_b\|(A_\mu^{(n-1)})^{-1}\|\|\tilde{r}_{\rm pr}^{n+1}\|/\alpha. \tag{4.25}$$

With simple calculations, we have

$$\frac{\|\tilde{r}_{\rm pr}^{n+1}\|}{\|r_{\rm pr}^{n+1}\|} \le \frac{\alpha}{\alpha - L_b\|(A_\mu^{(n-1)})^{-1}\|}, \tag{4.26}$$

i.e., the second inequality in (4.22) is thus proven. Analogously, replacing the left-hand side in (4.25) with $\|r_{\rm pr}^{n+1}\| - \|\tilde{r}_{\rm pr}^{n+1}\|$ yields the first inequality in (4.22). ☐

The assumption for L_b in (4.21) in Corollary 4.2 is reasonable only when $\|(A_\mu^{(n)})^{-1}\|$ is relatively small or moderate at most. When $\|(A_\mu^{(n)})^{-1}\|$ is large, we have the following corollary, where other upper and lower bounds for $\rho^n(\mu)$ are provided.

COROLLARY 4.3. *Under the assumptions in Theorem* 4.1, *for the vectors* $\{\tilde{r}_{\rm pr}^n\}_{n=1}^K$, *assume that there exist two positive constants* $\underline{\alpha}$, $\bar{\alpha}$ *such that*

$$\underline{\alpha} \le \frac{\|\tilde{r}_{\rm pr}^n\|}{\|\tilde{r}_{\rm pr}^{n+1}\|} \le \bar{\alpha}, \quad n = 1, \ldots, K-1, \quad \mu \in \mathcal{P}. \tag{4.27}$$

Assume that for all $\mu \in \mathcal{P}$ *the operator* $b(\cdot;\mu)$ *in* (4.2) *is bi-Lipschitz continuous w.r.t. the first argument, i.e., there exist two positive constants* $\underline{L}_b$, $\bar{L}_b$ *such that*

$$\underline{L}_b\|u_1 - u_2\| \le \|b(u_1;\mu) - b(u_2;\mu)\| \le \bar{L}_b\|u_1 - u_2\|, \quad u_1, u_2 \in \mathcal{W}^{\mathcal{N}}, \quad \mu \in \mathcal{P}. \tag{4.28}$$

Assume further

$$\underline{L}_b > \underline{\alpha}^{-1}/\|(A_\mu^{(n)})^{-1}\|, \quad n = 0, \ldots, K-1, \quad \mu \in \mathcal{P}. \tag{4.29}$$

Then

$$\underline{\rho}^{n+1}(\mu) \le \rho^{n+1}(\mu) \le \bar{\rho}^{n+1}(\mu), \tag{4.30}$$

where $\underline{\rho}^{n+1}(\mu) = \frac{1}{\bar{\alpha}\bar{L}_b\|(A_\mu^{(n-1)})^{-1}\|+1}$, $\bar{\rho}^{n+1}(\mu) = \frac{1}{\underline{\alpha}\,\underline{L}_b\|(A_\mu^{(n-1)})^{-1}\|-1}$, $n = 1, \ldots, K-1$, $\mu \in \mathcal{P}$.

Proof. By the definition of the vectors $r_{\rm pr}^{n+1}$ and $\tilde{r}_{\rm pr}^{n+1}$ (in (4.6) and (4.8), respectively), and using the Lipschitz condition in (4.28), we have

$$\begin{aligned}\|r_{\rm pr}^{n+1} - \tilde{r}_{\rm pr}^{n+1}\| &= \|b(\hat{u}^n(\mu);\mu) - b(u^n(\mu);\mu)\| \\ &\ge \underline{L}_b\|\hat{u}^n(\mu) - u^n(\mu)\| = \underline{L}_b\|(A_\mu^{(n-1)})^{-1}\tilde{r}_{\rm pr}^n\| \\ &\ge \underline{L}_b\|(A_\mu^{(n-1)})^{-1}\|\|\tilde{r}_{\rm pr}^n\|.\end{aligned} \tag{4.31}$$

By the first inequality in (4.27), we have

$$\|\tilde{r}^{n}_{\mathrm{pr}}\| \geq \underline{\alpha}\|\tilde{r}^{n+1}_{\mathrm{pr}}\|. \tag{4.32}$$

Substituting (4.32) into (4.31), and using the triangle inequality, we have

$$\|\tilde{r}^{n+1}_{\mathrm{pr}}\| + \|r^{n+1}_{\mathrm{pr}}\| \geq \|r^{n+1}_{\mathrm{pr}} - \tilde{r}^{n+1}_{\mathrm{pr}}\| \geq \underline{\alpha}\,\underline{L}_b\|(A^{(n-1)}_{\mu})^{-1}\|\|\tilde{r}^{n+1}_{\mathrm{pr}}\|,$$

so that

$$(\underline{\alpha}\,\underline{L}_b\|(A^{(n-1)}_{\mu})^{-1}\| - 1)\|\tilde{r}^{n+1}_{\mathrm{pr}}\| \leq \|r^{n+1}_{\mathrm{pr}}\|,$$

which implies that the second inequality in (4.30) holds. For the first inequality in (4.30), analogous to (4.25), we have

$$\|r^{n+1}_{\mathrm{pr}}\| - \|\tilde{r}^{n+1}_{\mathrm{pr}}\| \leq \|r^{n+1}_{\mathrm{pr}} - \tilde{r}^{n+1}_{\mathrm{pr}}\| \leq \bar{\alpha}\bar{L}_b\|(A^{(n-1)}_{\mu})^{-1}\|\|\tilde{r}^{n+1}_{\mathrm{pr}}\|,$$

due to the new assumptions in (4.27) and (4.28). The first inequality in (4.30) is thus proven. □

Although the bi-Lipschitz continuity is required in Corollary 4.3, no restriction is imposed on the upper Lipschitz constant $\bar{L}_b$. Moreover, the restriction on $\underline{L}_b$ in (4.28) is actually not strong if $\|(A^{(n)}_{\mu})^{-1}\|$ is large. For example, for systems which are nearly noncoercive, $\|(A^{(n)}_{\mu})^{-1}\|$ can be of $\mathcal{O}(10^3)$, or even larger.

Notice that (4.22) and (4.30) hold for $n = 1, \dots, K-1$. When $n = 0$, $\|\tilde{r}^1_{\mathrm{pr}}\|/\|r^1_{\mathrm{pr}}\| = 1$ for zero initial conditions, i.e., $u(0;\mu) \equiv 0$. For problems with nonzero initial conditions, i.e., $u(0;\mu) \neq 0$, one can use the variable transformation $\tilde{u}(\mu) = u(\mu) - u(0;\mu)$ to derive a transformed system with zero initial conditions; then the same conclusion can be similarly obtained.

Remark 4.4. From Corollaries 4.2 and 4.3, we know that the quantity $\rho^{n+1}(\mu)$ in (4.18) is bounded. One may use an upper bound $\bar{\rho}^{n+1}(\mu)$ in (4.22) or (4.30) to derive an output error bound, namely,

$$\|e^{n+1}_{\mathrm{O}}(\mu)\| \leq \Phi^{n+1}_{\mu}\bar{\rho}^{n+1}(\mu)\|r^{n+1}_{\mathrm{pr}}\|. \tag{4.33}$$

However, computing $\bar{\rho}^{n+1}(\mu)$ involves computing the Lipschitz constant L_b (or $\underline{L}_b$) and α (or $\underline{\alpha}$), which are nevertheless not practically computable.

Alternatively, for an efficient computation, one can directly estimate $\rho^{n+1}(\mu)$ by observing the maximal ratio among all the time steps $\max_{k\in\{1,\dots,K\}}\{\rho^k(\mu_\star)\}$, or the average ratio $\frac{1}{K}\sum_{k=1}^{K}\rho^k(\mu_\star)$ when the average of the output errors is estimated. Here, $\mu_\star$ is the parameter selected by the greedy algorithm, to be addressed in the following section. To compute the quantity $\tilde{r}^k_{\mathrm{pr}}(\mu_\star)$ for $\rho^k(\mu_\star)$, the detailed solutions $u^k(\mu_\star)$, $k = 1, \dots, K$, at $\mu_\star$ are required, which cause no additional cost for snapshot-based MOR methods because the detailed solutions at this parameter $\mu_\star$ are already available after the RB extension. Although the parameter $\mu_\star$, which causes the largest error (measured by the error estimation) in the parameter domain, may not be the one that causes the largest ratio $\rho^{n+1}(\mu)$, it makes sense to use the data at $\mu_\star$ to estimate $\rho^{n+1}(\mu)$. It should be pointed out that such an estimation on $\rho^{n+1}(\mu)$ can result in a sharp estimate but may sacrifice the rigorousness. Thus, we may say that the error "bound" obtained by estimating $\rho^{n+1}(\mu)$ is only an output error estimation.

Remark 4.5. When the operator $b(\cdot;\mu)$ is nonlinear w.r.t. the first argument and/or nonaffine w.r.t. the parameter μ, EI [3] can be employed. The ROM can be formulated following (2.4). In such a case, the term $\|r_{\mathrm{pr}}^{n+1}\|$ in (4.18) can be further bounded using the EI error bound,

$$\begin{aligned}
\|r_{\mathrm{pr}}^{n+1}\| &= \|B_\mu^{(n)}\hat{u}^n(\mu) + g(\hat{u}^n(\mu);\mu) - A_\mu^{(n)}\hat{u}^{n+1}(\mu)\| \\
&= \|B_\mu^{(n)}\hat{u}^n(\mu) + \mathcal{I}_M[g(\hat{u}^n(\mu);\mu)] - A_\mu^{(n)}\hat{u}^{n+1}(\mu) \\
&\qquad\qquad + g(\hat{u}^n(\mu);\mu) - \mathcal{I}_M[g(\hat{u}^n(\mu);\mu)]\| \\
&\leq \|r^{n+1}(\mu)\| + \|g(\hat{u}^n(\mu);\mu) - \mathcal{I}_M[g(\hat{u}^n(\mu);\mu)]\| \\
&\leq \|r^{n+1}(\mu)\| + \epsilon_{\mathrm{EI}}^n(\mu),
\end{aligned}$$

where $r^{n+1}(\mu)$ is defined in (3.1), and $\epsilon_{\mathrm{EI}}^n(\mu)$ is the error due to the EI, as defined in (3.5).

To compute the error bound in (4.18), one needs to efficiently compute the norm of the matrix inverse $(A_\mu^{(n)})^{-T}$. The following remark addresses how to evaluate $\|(A_\mu^{(n)})^{-T}\|$. Note that no assumption is imposed to the matrix $A_\mu^{(n)}$.

Remark 4.6. If the vector norm is taken as the standard 2-norm, e.g., when the discrete system is obtained by the finite volume or finite difference discretization, the matrix norm $\|(A_\mu^{(n)})^{-T}\|$ is the spectral norm of $(A_\mu^{(n)})^{-T}$. Therefore,

$$\|(A_\mu^{(n)})^{-T}\|_2 = \|(A_\mu^{(n)})^{-1}\|_2 = \sigma_{\max}\left((A_\mu^{(n)})^{-1}\right) = \frac{1}{\sigma_{\min}(A_\mu^{(n)})}, \tag{4.34}$$

the reciprocal of the smallest singular value of $A_\mu^{(n)}$. For some special cases in which the matrix $A_\mu^{(n)}$ is a constant matrix, the smallest singular value of A is computed once and can be used repeatedly.

For the general vector norm $\|\cdot\|_H$, induced by the inner product $\langle v_1, v_2\rangle := v_1^T H v_2$, $v_1, v_2 \in \mathcal{W}^{\mathcal{N}}$, where H is a symmetric positive definite matrix, e.g., the mass matrix in the FE discretization, the induced matrix norm can be defined as

$$\|Z\|_H := \max_{\|x\|=1} \|Zx\|_H = \max_{\|x\|=1} \sqrt{x^T Z^T H Z x} = \|Z^T H Z\|_2, \quad Z \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}}.$$

This implies that

$$\begin{aligned}
\|Z^T H Z\|_2 &= \sqrt{\lambda_{\max}\left((Z^T H Z)^T Z^T H Z\right)} \\
&= \lambda_{\max}(Z^T H Z) \\
&= \lambda_{\max}\left(Z^T L^T L Z\right) \\
&= \sigma_{\max}^2(LZ).
\end{aligned}$$

Here L is a lower triangular matrix of the Cholesky factorization of H, i.e., $L^T L = H$; $\lambda_{\max}(\cdot)$ refers to the largest eigenvalue of a matrix. Thus, $\|(A_\mu^{(n)})^{-T}\|_H$ can be obtained as

$$\|(A_\mu^{(n)})^{-T}\|_H = \sigma_{\max}^2\left(L(A_\mu^{(n)})^{-T}\right) = \frac{1}{\sigma_{\min}^2((A_\mu^{(n)})^T L^{-1})}.$$

Remark 4.7. The assumptions on the Lipschitz constants (i.e., L_b and $\underline{L}_b$) in Corollaries 4.2 and 4.3 require that α and $\underline{\alpha}$ cannot be too small. This can be achieved if the time step of the detailed simulation is well chosen. In fact, a well chosen time step results in an even distribution of the error of the solution to the FOM over the time interval, and this property can be inherited by the solution of the ROM, see, e.g., [20]. In addition, the values of α, $\underline{\alpha}$, and $\bar{\alpha}$ are $\mathcal{O}(1)$, as will be shown in Figures 2 and 7 in section 6.

Remark 4.8. For the case of multiple outputs, i.e., $N_{\mathrm{o}} > 1$, an error bound for each component of the output vector can be obtained from Theorem 4.1. The final error bound for the whole vector of outputs can be taken as the maximum of all the error bounds.

Notice that the error bound is independent of the projection matrix pairs $(V_{\mathrm{pr}}, W_{\mathrm{pr}})$ and $(V_{\mathrm{du}}, W_{\mathrm{du}})$. It is applicable to any projection-based MOR method. In addition, if one takes $W_{\mathrm{pr}} = V_{\mathrm{pr}}$, then the ROM can be obtained by using Galerkin projection, as is usually implemented by RB MOR methods.

Remark 4.9. For dynamic systems, even Galerkin projection cannot guarantee the stability of the ROM [7, 37]. The stability issue of the ROM is, in general, still an open problem, though many studies have been made and some strategies have been suggested to prevent producing unstable ROMs for certain problems; see, e.g., [1, 19, 31, 33, 37].

5. Construction of the projection matrix.

In this section, we focus on the construction of the projection matrix using the RB method [27]. Usually, Galerkin projection is employed to construct the ROM for the RB method, i.e., $W = V$. For parametrized systems, the projection matrix V is usually generated iteratively through a greedy algorithm, by which the dimension of the reduced space can be kept as small as possible while the accuracy of the ROM is guaranteed. More precisely, a training set $\mathcal{P}_{\mathrm{train}}$ with a finite number of parameter samples is typically chosen a priori in an admissible parameter domain. At each extension step, a parameter $\mu_\star$, which causes the largest error measured by a proper error estimator $\psi(\cdot)$, is chosen from $\mathcal{P}_{\mathrm{train}}$ to enrich the projection matrix. The iteration continues until the error estimator $\psi(\mu_\star)$ goes below the required accuracy $\varepsilon_{\mathrm{ROM}}$. For time dependent problems, the POD-Greedy algorithm [17] is often used to construct the RB. Algorithm 1 shows the basic step of the POD-Greedy algorithm.

Algorithm 1. RB generation using POD-Greedy.

```
Input:  $\mathcal{P}_{\mathrm{train}}$, $\mu_0$, $\varepsilon_{\mathrm{ROM}}(< 1)$.
Output: RB $V = [v_1, \dots, v_N]$.
Initialization: $N = 0$, $\mu_\star = \mu_0$, $\eta_N(\mu_\star) = 1$, $V = [\,]$.
while $\psi_N(\mu_\star) > \varepsilon_{\mathrm{ROM}}$ do
    Compute the trajectory $S_{\max} := \{u^n(\mu_\star)\}_{n=0}^{K}$.
    Enrich the RB, e.g., $V := [V, v_{N+1}]$, where $v_{N+1}$ is the first POD mode of the
    matrix $\bar{U} := [\bar{u}^0, \dots, \bar{u}^K]$ with $\bar{u}^n := u^n(\mu_\star) - \Pi_{\mathcal{W}^N}[u^n(\mu_\star)]$, $n = 0, \dots, K$.
    $\Pi_{\mathcal{W}^N}[u]$ is the projection of $u$ onto the current space $\mathcal{W}^N := \mathrm{span}\{v_1, \dots, v_N\}$.

    $N = N + 1$.
    Find $\mu_\star := \arg \max_{\mu \in \mathcal{P}_{\mathrm{train}}} \psi_N(\mu)$.
end while
```

Remark 5.1. For many problems, like the batch chromatographic model and the SMB model under consideration in this paper, the total number of time steps in the FOM simulation is very large. This implies that the number of snapshots K

in step 3 in Algorithm 1 is large if no appropriate pretreatment for the snapshots is applied. The large number of snapshots will result in expensive computations in step 4. To tackle this problem, the technique of adaptive snapshot selection [5, 40] can be employed to discard the redundant (linearly dependent) information from the trajectory, so that the runtime for the RB construction can be largely reduced.

6. Numerical experiments.

6. Numerical experiments. In this section, four models will be presented to show the performance of the proposed error estimation. The first one is a linear convection-diffusion equation in [13], which is used to compare the performance of the new error estimation and the error bound using the existing primal-dual approach [13, 16]. The second model is the viscous Burgers' equation, which is used to demonstrate that our method is applicable to a large class of nonlinear evolution equations. The other two models arise from chromatographic separation processes, which are a batch chromatographic model and a linear continuous SMB model. For particular applications in chemical engineering, the resulting ROMs are employed to solve the underlying optimization problems.

The RB method [17, 27, 40] is employed to construct the ROMs for all the following examples. More specifically, Algorithm 1 is used to generate the projection matrix V, and the ROMs are constructed by using Galerkin projection. For comparison, we use the newly proposed output error estimation and some existing output error bounds or estimations as error indicators to construct the ROMs, respectively. The error indicator is defined as $\psi_N(\mu) := \frac{1}{K}\sum_{n=1}^{K}\Psi_N^n(\mu)$, where $\Psi_N^n(\mu)$ is the corresponding output error bound/estimation for the parameter μ at the time instance t^n. For example, when the newly proposed output error estimation is employed, $\Psi_N^n(\mu) = \Delta_N^n(\mu)$, where $\Delta_N^n(\mu)$ is defined in (4.18). The error indicator $\psi_N(\mu)$ is used to measure the average output error (i.e., $\frac{1}{K}\sum_{n=1}^{K}\|y(u^n(\mu)) - y(\hat{u}^n(\mu))\|$) over the whole evolution process.

In what follows, EB refers to the existing output error bound or estimation, and ES refers to the newly proposed error estimation. To compute the new output error estimation, the quantity $\rho^{n+1}(\mu)$ in (4.18) needs to be estimated, as discussed in Remark 4.4. In this work, we use the average ratio $\tilde{\rho}_N^\star := \frac{1}{K}\sum_{k=1}^{K}\rho^k(\mu_\star)$ to estimate $\rho^{n+1}(\mu)$, since we measure the average of the output errors over time. More precisely, after each iteration of the greedy algorithm, we compute the average ratio $\tilde{\rho}_N^\star$ at the selected parameter $\mu_\star$ and use it as an estimate of $\rho^{n+1}(\mu)$ for the next iteration. All the computations were done with C++ code on a PC with an Intel Core 2 Quad CPU Q9550 2.83 GHz 4.00 GB RAM unless stated otherwise.

6.1. Linear convection-diffusion equation.

In this section, we consider a linear convection-diffusion equation which models the movement of fluids and other transport phenomena. This model is used as a test case for the primal-dual error bound in [13]. Here we use it to compare the performance of the proposed error estimation and the existing primal-dual error bound in [13, 16].

6.1.1. Model description and reduced-order modeling.

The governing equation for this model is given by

$$u_t = q_1 u_{xx} + q_2 u_x - q_2, \quad x \in \Omega := (0,1), \quad t \in (0,T].$$

The initial and boundary conditions are

$$u(0,x) = -2x^2 + 2x; \quad u(t,0) = u(t,1) = 0, \quad t > 0.$$

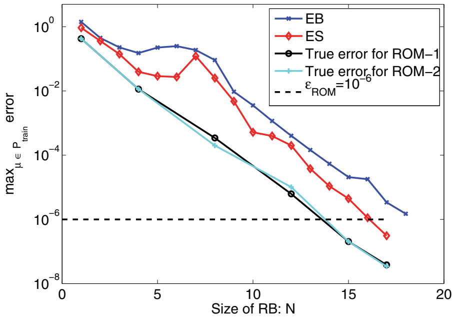


Fig. 1. *Decay of the error estimates and the corresponding true error during the RB construction process for the linear convection-diffusion equation. EB: the error bound in* [16]*; ES: the newly proposed error estimation.*

The output of interest is the average value of u over a small interval Ω_0 as a function of time, i.e., $y(u(t)) := \frac{1}{|\Omega_0|}\int_{\Omega_0} u(t,x)\,dx$, $\Omega_0 := [0.495, 0.505]$.

In this model, we chose the diffusivity q_1 and the velocity q_2 as the parameters, i.e., $\mu := (q_1, q_2)$. The interesting parameter domain is chosen as $\mathcal{P} := [0.1, 1] \times [0.5, 5]$. To construct the FOM, we use the finite volume method for the spatial discretization and the backward Euler scheme for the temporal discretization. We chose the number of degrees of freedom for the FOM $\mathcal{N} = 800$, and an equal time step $\Delta t = T/K$, $T = 1, K = 100$. The FOM is of the general form in (2.2), except that no nonlinear term is involved. Algorithm 1 is employed to construct the ROM with two different output error estimations.

6.1.2. Results. The training set $\mathcal{P}_{\text{train}}$ consists of 200 sample points randomly distributed in the parameter domain $\mathcal{P}$. Figure 1 shows the decay of both error estimates and the corresponding true error for the output during the RB construction process. ROM-1 and ROM-2 are the ROMs constructed by using EB and ES, respectively. It is seen that the new ES outperforms the existing primal-dual EB in [16].

As mentioned above, the estimation of $\rho^n(\mu)$ (by $\tilde{\rho}_N^\star$) is based on the fact that it is bounded, as shown in the two corollaries, i.e., Corollaries 4.2 and 4.3. We have carefully checked the assumptions made in the two corollaries, and the results are detailed as follows. First, we found that all values of $\|(A_\mu^{(n)})^{-1}\|$ are in the range of $\mathcal{O}(1)$; more precisely, they are in the interval $[0.9, 1.0]$. Second, we plot the ratio $\|\tilde{r}_{\text{pr}}^{n+1}\|/\|\tilde{r}_{\text{pr}}^{n}\|$ at the chosen parameter $\mu_\star$ as a function of the time index t^n for different RB dimensions in Figure 2. It can be seen that all the values of the ratio are in the range of $[0.4, 3]$. In fact, the ratio at other dimensions is pretty similar, i.e., it is always in the range in $\mathcal{O}(1)$. This means that the constant α in (4.19) exists, and it is also in $\mathcal{O}(1)$. As a result, $\alpha/\|(A_\mu^{(n)})^{-1}\| \approx \mathcal{O}(1)$, and the condition on the Lipschitz constant in (4.21) becomes $L_b \lesssim 1$, which is reasonable for a linear continuous operator, as here for this example $L_b = 1$. Thus, all the assumptions in Corollary 4.2 are satisfied, so the quantity $\rho^n(\mu)$ is bounded, and in turn, using $\tilde{\rho}_N^\star$ as an estimate of $\rho^n(\mu)$ is practical and meaningful. Figure 3 shows the average ratio $\tilde{\rho}_N^\star$ as a function of the RB size N. We see that it converges to 1 as the RB is extended.

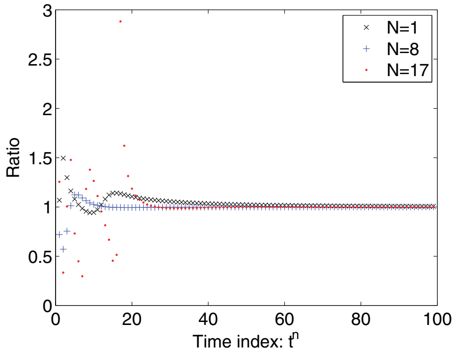


FIG. 2. *Behavior of the ratio* $\frac{\|\tilde{r}^{n+1}\|}{\|\tilde{r}^{n}\|}$ *in the time trajectory corresponding to different RB dimensions for the linear convection-diffusion equation.*

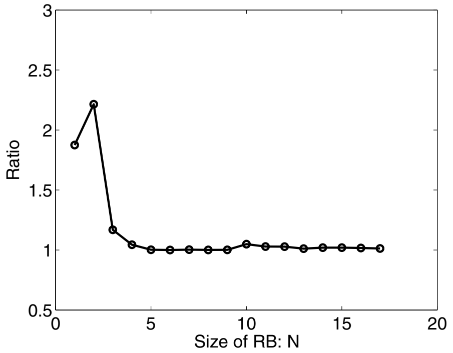


FIG. 3. *Behavior of the average ratio* $\tilde{\rho}_N^{\star}$ *during the RB construction process for the linear convection-diffusion equation.*

6.2. Burgers' equation. The Burgers' equation describes the fundamental nonlinear phenomena in fluid dynamics and is often considered as the starting point to test a new algorithm for nonlinear problems. We now use the unsteady viscous Burgers' equation to show that the proposed error estimation is applicable to MOR for general nonlinear evolution equations.

6.2.1. Reduced-order modeling of Burgers' equation. In this work, we consider the unsteady viscous Burgers' equation as follows:

$$u_t + \left(\frac{u^2}{2}\right)_x = \nu u_{xx} + s(u,x), \quad x \in (0,1), \quad t \in (0,T], \tag{6.1}$$

where $\nu \in \mathcal{P}$ is the viscosity coefficient, and $s(u,x)$ is the source term. The output of interest is the value of u at $x=1$ as a function of t, i.e., $y(t;\nu) := u(t,1;\nu)$.

In this model, the viscosity coefficient ν is considered as the parameter, i.e., $\mu := \nu$. Note that the computation becomes more challenging when ν is smaller, e.g., $\nu \approx \mathcal{O}(10^{-3})$, because the instability grows exponentially with the evolution time [25]. For MOR, it becomes more challenging when a smaller value of ν is involved. To numerically verify this, we chose two parameter domains: $\tilde{\mathcal{P}} = [0.05, 1]$ and $\mathcal{P} = [0.001, 1]$. We will see that the ROM has a better convergence rate and other good properties when $\tilde{\mathcal{P}}$ with a larger value of ν is employed. We take $T = 2$ and $s(u, x) \equiv 1$ in the following computations. For discretization, we use the finite volume method to construct the FOM, in the general form of (2.2).

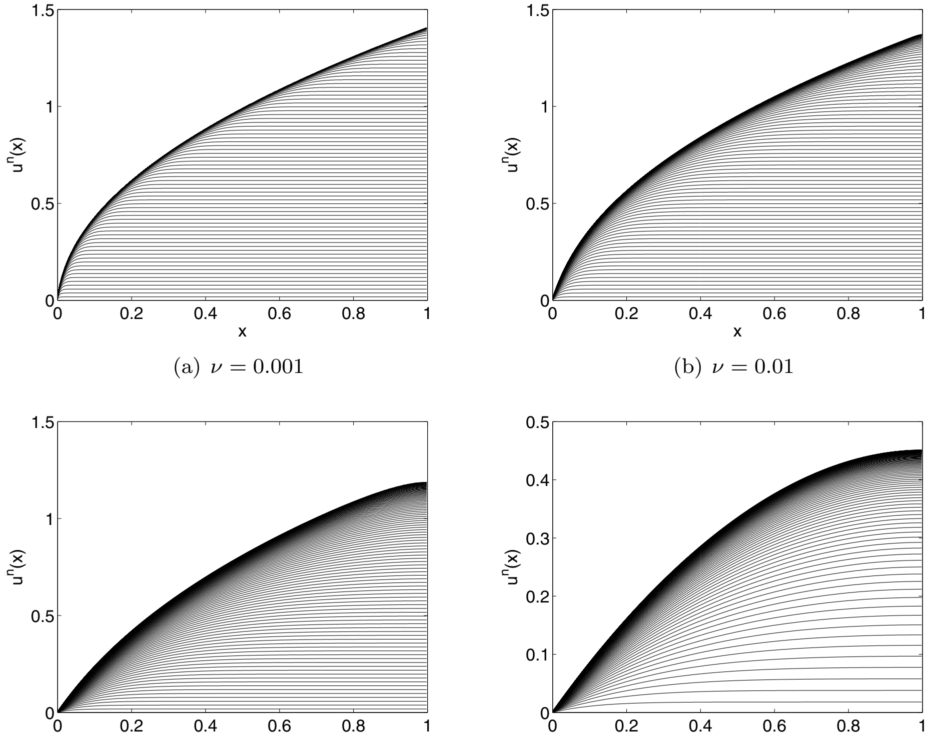
An a posteriori error estimation for the RB method applied to this equation is proposed in [25], where the successive constraint method was used to estimate the lower bound of the stability constant. The error estimation is actually a summation over time of the dual norm of the residual. As pointed out in [25], this error estimation is no longer useful when the viscosity ν is small and the final time T is large. In addition, this error estimation is applicable to problems which are at most quadratically nonlinear. In contrast, the newly proposed error estimation is applicable to MOR for general nonlinear evolution equations.

6.2.2. Results. The following results are obtained by using the following initial and boundary conditions:

$$u(0, x) = 0, \quad x \in [0, 1]; \quad u(t, 0) = 0, \quad u_x(t, x)|_{x=1} = 0.$$

We use a uniform spatial grid with $\mathcal{N} = 500$ cells for the FOM and $\Delta t = T/K, K = 1000$ for both the FOM and ROM simulations.

Figure 4 shows the solutions to the FOM as a function of x and t^n. Each line represents the solution $u(x, t^n)$ at the time instance $t = t^n$, $n = 10j$, $j = 0, \ldots, K/10$.



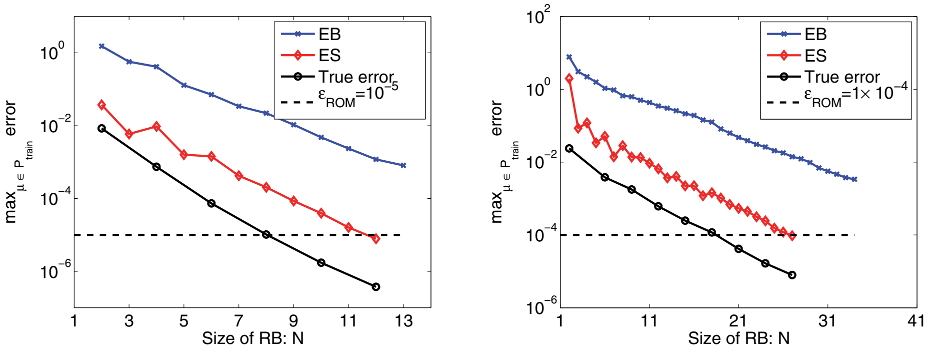
(a) $\nu = 0.001$

(b) $\nu = 0.01$

(c) $\nu = 0.1$

(d) $\nu = 1.0$

Fig. 4. *Solution of the Burgers' equation as a function of x and t^n with different viscosity coefficients ν. Each line represents the solution $u(x, t^n)$ at the time instance $t = t^n$, $n = 10j$, $j = 0, \ldots, K/10$.*



(a) $\nu \in \tilde{\mathcal{P}}$ (b) $\nu \in \mathcal{P}$

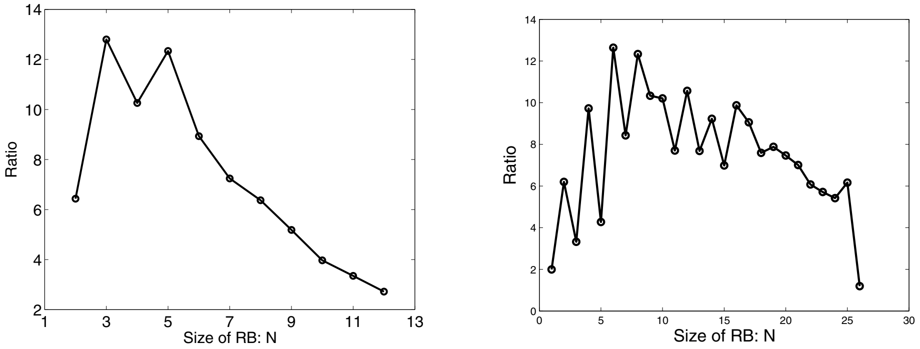
FIG. 5. *Decay of the error bound/estimation and the corresponding true error during the RB construction process for the Burgers' equation. EB: the error bound in Proposition* 3.1*; ES: the newly proposed error estimation.* (a) $\nu \in \tilde{\mathcal{P}} = [0.05, 1]$*;* (b) $\nu \in \mathcal{P} = [0.001, 1]$.

The evolution process tends to be steady at final time. For the ROM construction, we chose a training set with 70 sample points log-uniformly distributed in the parameter domain $\mathcal{P}$ to build the RB and the basis for the EI, respectively.

The behavior of the error bounds and the corresponding true error are illustrated in Figure 5. It is seen that in both cases the new error estimation, denoted as ES, works much better than EB, which is based on Proposition 3.1. Moreover, the new error estimation is fairly sharp in comparison with the true error. Last but not least, the convergence rate becomes relatively slow and many more basis vectors are needed to achieve a certain accuracy, when smaller viscosity coefficient ν is involved. Since this is just an academic numerical example, there is not much computational time reduction from the ROM. For runtime comparison, we will report the computational time for the more challenging problems in the following subsections.

As discussed in Remark 4.4, the constant $\rho^n(\mu)$ can be estimated based on the observation of the average ratio over all the time steps at the selected parameter $\mu_\star$ at each iteration step of the greedy algorithm. Notice that the ratio is changing with the dimension of the RB and with the parameter $\mu_\star$ selected at each iteration step. The behavior of the ratio during the RB extension process is illustrated in Figure 6. From Figure 6(a), we see that the ratio decreases ("almost monotonically") as the RB is extended, which demonstrates that the difference between $\|\tilde{r}_{\rm pr}^{n+1}\|$ and $\|r_{\rm pr}^{n+1}\|$ becomes small as the accuracy of the ROM is increased. However, when smaller viscosity coefficient ν is involved, the ratio oscillates during the basis extension process, as shown in Figure 6(b). This is probably because the instability grows too fast when ν is small. In most cases, the value of the ratio is of the magnitude $\mathcal{O}(1)$, when the accuracy of the ROM achieves a certain degree, which will be further confirmed in the following examples (see Figures 9 and 12).

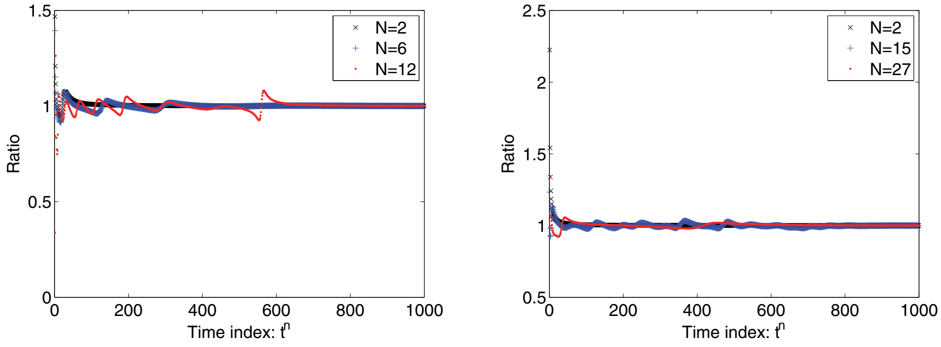
As addressed in Remark 4.7, with well-chosen time steps, the approximation errors $(\hat{u}^n(\mu) - u^n(\mu))$ can be evenly distributed in the time trajectory, so that the norm of the vectors $\tilde{r}_{\rm pr}^{n+1} = A_\mu^{(n)} u^{n+1}(\mu) - A_\mu^{(n)} \hat{u}^{n+1}(\mu)$ is of the same magnitude over time, i.e., the assumptions in (4.21) and (4.29) (in Corollaries 4.2 and 4.3, respectively) are fulfilled. To numerically verify this, we plot the ratio $\frac{\|\tilde{r}_{\rm pr}^{n+1}\|}{\|\tilde{r}_{\rm pr}^{n}\|}$ as a function of time instant t^n for different RB dimensions in Figure 7. It is seen that the ratio



(a) $\nu \in \tilde{\mathcal{P}}$

(b) $\nu \in \mathcal{P}$

FIG. 6. *Behavior of the average ratio $\tilde{\rho}_N^\star$ during the RB construction process for the Burgers' equation.* (a) $\nu \in \tilde{\mathcal{P}} = [0.05, 1]$; (b) $\nu \in \mathcal{P} = [0.001, 1]$.



(a) $\nu \in \tilde{\mathcal{P}}$

(b) $\nu \in \mathcal{P}$

FIG. 7. *Behavior of the ratio $\frac{\|\tilde{r}^{n+1}\|}{\|\tilde{r}^n\|}$ in the time trajectory corresponding to different RB dimensions for the Burgers' equation.* (a) $\nu \in \tilde{\mathcal{P}} = [0.05, 1]$; (b) $\nu \in \mathcal{P} = [0.001, 1]$.

is in the range of $[0.85, 1.5]$, which implies that the constants α, $\underline{\alpha}$, and $\bar{\alpha}$ are all of the magnitude of $\mathcal{O}(1)$. In addition, based on our discretization, the norms of the matrix inverse $\|(A_\mu^{(n)})^{-1}\|$ are all in range of $[0.95, 1]$ for all $\mu \in \mathcal{P}_{\text{train}}$. Thus, $\alpha / \|(A_\mu^{(n)})^{-1}\| \approx \mathcal{O}(1)$, which means that the assumption on the Lipschitz constant L_b in (4.20) in Corollary 4.2 is reasonable. To avoid repetition, we will not show the similar plots for the next two examples.

6.3. Batch chromatographic model.

Batch chromatography is an important chemical process and is widely used for separation and purification in industry. In this subsection, we show the performance of the new error estimation as compared to the error bound in Proposition 3.1.

6.3.1. Model description and optimization.

The governing equations of batch chromatography are as follows:

$$\text{(6.2)} \qquad \begin{cases} \dfrac{\partial c_z}{\partial t} + \dfrac{1-\epsilon}{\epsilon}\dfrac{\partial q_z}{\partial t} = -\dfrac{\partial c_z}{\partial x} + \dfrac{1}{Pe}\dfrac{\partial^2 c_z}{\partial x^2}, & 0 < x < 1, \\ \qquad\qquad\qquad \dfrac{\partial q_z}{\partial t} = \dfrac{L}{Q/(\epsilon A_c)}\kappa_z(q_z^{\mathrm{Eq}} - q_z), & 0 \le x \le 1, \end{cases}$$

where c_z, q_z are the concentrations of the component z $(z = a, b)$ in the liquid and solid phases, respectively, Q the volumetric feed flow rate, A_c the cross-sectional area of the column with the length L, ϵ the column porosity, κ_z the mass-transfer coefficient, and Pe the Péclet number. The adsorption equilibrium q_z^{Eq} is described by the isotherm equations of bi-Langmuir type,

$$q_z^{\mathrm{Eq}} = f_z(c_a, c_b) := \frac{H_{z1}c_z}{1 + K_{a1}c_a^{\mathrm{f}}c_a + K_{b1}c_b^{\mathrm{f}}c_b} + \frac{H_{z2}c_z}{1 + K_{a2}c_a^{\mathrm{f}}c_a + K_{b2}c_b^{\mathrm{f}}c_b},$$

where c_z^{f} is the feed concentration of component z. The initial and boundary conditions are given as follows:

$$\begin{cases} c_z(0,x) = 0, \quad q_z(0,x) = 0, \quad 0 \le x \le 1, \\ \dfrac{\partial c_z}{\partial x}|_{x=0} = Pe\left(c_z(t,0) - \chi_{[0,t_{\mathrm{in}}]}(t)\right), \\ \dfrac{\partial c_z}{\partial x}|_{x=1} = 0, \end{cases}$$

where t_{in} is the injection period, and $\chi_{[0,t_{\mathrm{in}}]}$ is the characteristic function,

$$\chi_{[0,t_{\mathrm{in}}]}(t) = \begin{cases} 1 & \text{if } t \in [0, t_{\mathrm{in}}], \\ 0 & \text{otherwise.} \end{cases}$$

In this paper, we consider the flow feed rate Q and the injection period t_{in} as the operating parameters, i.e., $\mu := (Q, t_{\mathrm{in}})$. We chose the interesting parameter domain as $\mathcal{P} := [0.0667, 0.1667] \times [0.5, 2]$. The outputs of interest are the concentrations at the outlet of the column, i.e., $y(t;\mu) := (c_{a,\mathrm{O}}(t;\mu), c_{b,\mathrm{O}}(t;\mu))$, where $c_{z,\mathrm{O}}(t;\mu) = c_z(t,1;\mu)$ is the concentration of component z at the outlet of the column, $z = a, b$.

The optimization of batch chromatography considered here aims to maximize the production rate Pr while respecting the requirement of the recovery yield Rec, i.e.,

$$\begin{aligned} &\min_{\mu \in \mathcal{P}} \{-Pr(\mu)\}, \\ \text{s.t. } \; & Rec_{\min} - Rec(\mu) \le 0, \quad \mu \in \mathcal{P}, \\ & c_z(\mu),\ q_z(\mu) \text{ are the solutions to the system (6.2)},\ z = a,\ b, \end{aligned}$$

where $Pr(\mu) = \frac{Qp(\mu)}{t_4 - t_1}$, $Rec(\mu) = \frac{p(\mu)}{t_{\mathrm{in}}(c_a^{\mathrm{f}} + c_b^{\mathrm{f}})}$, $p(\mu) = \int_{t_3}^{t_4} c_{a,\mathrm{O}}(t;\mu)\,\mathrm{d}t + \int_{t_1}^{t_2} c_{b,\mathrm{O}}(t;\mu)\,\mathrm{d}t$, and $Rec_{\min}$ is the minimal requirement of the recover yield. The cutting points t_1, t_4 are determined by a minimum concentration threshold that the detector can resolve, and t_2, t_3 are determined by the requirement of the product purity. More details can be found in [40].

6.3.2. Reduced-order modeling of the batch chromatographic model. We use finite volume discretization to construct the FOM in the formulation of (4.1). The training set $\mathcal{P}_{\mathrm{train}}$ consists of 60 sample points uniformly distributed in the parameter domain. The ROM is in the form of (4.4). The coefficient matrix $A_\mu^{(n)}$ in (4.1) is a constant matrix at all time instances t^n, $n = 0, \ldots, K-1$, and is independent

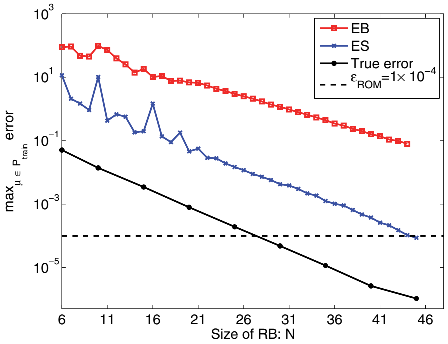


Fig. 8. *Decay of the error bound/estimation and the corresponding true error during the RB construction process for batch chromatography. EB: the error bound in Proposition* 3.1*; ES: the newly proposed error estimation.*

Table 1

Comparison of runtime for the generation of the RB using two error estimates for batch chromatography.

Model	Runtime (h)*
ROM using EB	6.8
ROM using ES	7.6

* Due to memory limitations of the PC, these computations were done on a workstation with 4 Intel Xeon E7-8837 CPUs (8 cores per CPU), 2.67 GHz, 1 TB RAM.

of the parameter μ. This implies that the dual system (4.3) is independent of the parameter and time. Consequently, the related quantities from the dual system for the error estimation are computed only once for all the sample points in the training set at the current iteration step of the greedy algorithm. They are updated only at the next greedy iteration step.

6.3.3. Results. Since we have two outputs $(c_{a,\mathrm{O}}(t;\mu), c_{b,\mathrm{O}}(t;\mu))$ in this model, we define a dual system for each output to compute an output error estimation and take the maximum as the error indicator for the parameter μ, as discussed in Remark 4.8. Figure 8 shows the error estimation decay as the RB is enriched. It is seen that the new output error estimation (ES) works much better than the old one (EB). The new error estimation goes below the prespecified tolerance as the number of the RB increases to 45; in contrast, EB almost stagnates and is still above the tolerance.

To show the efficiency of the new error estimation, we compare the runtime for the generation of the RB. From Table 1, we see that using ES takes slightly more time than using EB. This is because the residual of an additional dual system needs to be computed for ES. However, since ES is much more accurate than EB, it deserves spending a bit more computational time for acquiring a more reliable ROM.

Figure 9 shows the behavior of the average ratio $\tilde{\rho}_N^{\star}$ during the RB extension process. We have the same conclusion as above, i.e., the difference between $\|\tilde{r}_{\mathrm{pr}}^{n+1}\|$ and $\|r_{\mathrm{pr}}^{n+1}\|$ becomes small as the accuracy of the ROM is increased. The ratio stays in the scale of $\mathcal{O}(1)$ when the number of basis vectors is larger than 20.

Before addressing the ROM-based optimization, we assess the validation of the ROM. To this end, we perform detailed and reduced simulations over a test set with 500 random samples of the parameter in the feasible domain. Table 2 shows the

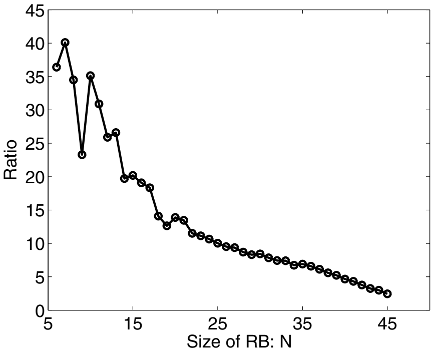


Fig. 9. *Behavior of the average ratio* $\tilde{\rho}_N^\star$ *during the RB construction process for batch chromatography.*

Table 2

Comparison of runtime for the detailed and reduced simulations of batch chromatography over a validation set with 500 *random sample points.* $\varepsilon_{\mathrm{ROM}} = 1 \times 10^{-4}$.

Model	Maximal error	Average runtime (s)/SpF
FOM ($\mathcal{N} = 1500$)	–	339.02(-)
ROM ($N = 45$)	8.16×10^{-7}	5.95/ 57

Table 3

Comparison of the optimization of batch chromatography based on the FOM (FOM-Opt.) and the ROM (ROM-Opt.).

Method	Objective	Optimal solution	#Iterations	Runtime (h)/SpF
FOM-Opt.	0.020264	$(0.07964, 1.05445)$	202	33.88 / -
ROM-Opt.	0.020266	$(0.07964, 1.05445)$	202	0.58 / 58

results. It is seen that the average runtime is sped up by a factor of 58 by using the ROM, and the maximal true output error is 8.16×10^{-7}, which is below the prespecified tolerance. We use the global optimizer NLOPT_GN_DIRECT_L in the open library NLopt [21] to solve the optimization problems. The tolerance for the optimization is taken as $\epsilon_{\mathrm{opt}} = 1.0 \times 10^{-4}$. The optimization results are summarized in Table 3. The optimal solution of the ROM-based optimization converges to that of the FOM-based optimization one, and the runtime is significantly reduced. The speedup factor (SpF) is 58.

6.4. Continuous SMB chromatographic model. SMB chromatography is a continuous multicolumn process and has been widely used as an efficient separation technique in chemical engineering. Recent studies on MOR of the SMB model can be found in [22] and the references therein. In [22] the ROM is generated by a POD-based MOR method and the ROM needs to be updated during the trust-region optimization process. Here, we use the RB method to build a ROM, which is qualified over the whole parameter domain. Notably, for problems like the SMB model under consideration, the evolution process is extraordinarily complicated due to the periodic switching procedure. This makes the existing error estimators, e.g., in [9, 40], hard to compute, because extra errors are introduced due to the switching. In contrast, the new error estimator only considers the residual at the current time instance, i.e., relatively independent of the previous steps.

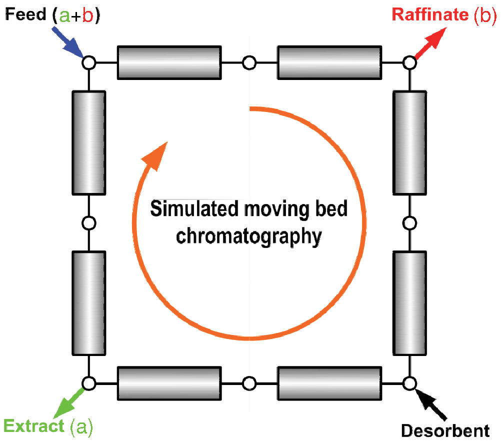


Fig. 10. *Schematic illustration of a SMB chromatographic process with four zones and eight columns.*

6.4.1. Model description and optimization. A classical SMB process with four zones is schematically shown in Figure 10. The SMB model consists of several single column models which are assembled with balance equations at the inlets and outlets. It is assumed that the flow behavior of each column is described as a plug-flow with a limited mass-transfer rate characterized by a linear driving force approximation. The governing equations for each column model are given by

$$\text{(6.3)} \qquad \begin{cases} \dfrac{\partial c_z}{\partial t} + \dfrac{1-\epsilon}{\epsilon}\dfrac{\partial q_z}{\partial t} = -\dfrac{Qt_{\rm s}}{\epsilon A_c L}\left(\dfrac{\partial c_z}{\partial x} - \dfrac{1}{Pe}\dfrac{\partial^2 c_z}{\partial x^2}\right), & 0 < x < 1, \\ \qquad\qquad\qquad \dfrac{\partial q_z}{\partial t} = t_{\rm s}\kappa_z(q_z^{\rm Eq} - q_z), & 0 \le x \le 1, \end{cases}$$

where c_z and q_z are the concentrations of the solute z $(z = a, b)$ in the liquid and solid phases, and other quantities like Q, A_c, ϵ, L, and Pe have the same meanings as those in the batch chromatographic model above, t and x are the dimensionless time and spatial coordinates, and $t_{\rm s}$ is the switching period. The adsorption equilibrium is $q_z^{\rm Eq} := H_z c_z$ with H_z being the Henry constant, which implies that the system of equations (6.3) is linear. It is assumed that $H_a > H_b$. The boundary conditions are

$$\begin{cases} \dfrac{\partial c_z}{\partial x}|_{x=0} = Pe(c_z(t,0) - c_z^{\rm in}), \\ \dfrac{\partial c_z}{\partial x}|_{x=1} = 0, \end{cases}$$

where $c_z^{\rm in}$ is the concentration of component z at the column inlet. More details about the description of the SMB model, e.g., the balanced equations around the inlet and outlet nodes, can be found in [22, 30]. The model parameters are summarized in Table 4.

As a case study, we use an SMB model with four zones and eight columns, as shown in Figure 10. In this model, the flow rate in each zone Q_i, $i = \mathrm{I}, \dots, \mathrm{IV}$, and the switching period $t_{\rm s}$ are the operating parameters. Alternatively, four corresponding dimensionless quantities m_i, $i = \mathrm{I}, \dots, \mathrm{IV}$, and the feed flow rate $Q_{\rm f}$ can also be

TABLE 4
Model parameters and operating conditions for the SMB model.

Column dimensions (cm)	2.6 × 11
Column porosity ϵ (-)	0.4
Péclet number Pe (-)	500
Mass-transfer coefficients κ_z, $z = a, b$ (1/s)	0.1
Feed concentrations c_z^{f}, $z = a, b$ (g/l)	2.9
Henry constants H_a, H_b (-)	3.86, 2.72

chosen as the operating parameters. The four dimensionless quantities introduced by the triangle theory [23] are defined as

$$m_i = \frac{Q_i t_{\mathrm{s}} - \epsilon \mathbf{V}}{\mathbf{V}(1-\epsilon)}, \quad i = \mathrm{I}, \ldots, \mathrm{IV},$$

where $\mathbf{V}$ is the volume of the column. Given a set of parameters $\mu := (m_{\mathrm{I}}, \ldots, m_{\mathrm{IV}}, Q_{\mathrm{f}})$, the SMB process reaches a cyclic steady state (CSS) with a periodic switching along the circularly arranged columns. The CSS condition is defined by

$$\max_{z \in \{a,b\}} \max \left\{ \| c_z(0,\cdot) - P_{\mathrm{s}}[c_z(1,\cdot)] \|, \| q_z(0,\cdot) - P_{\mathrm{s}}[q_z(1,\cdot)] \| \right\} < \varepsilon_{\mathrm{CSS}},$$

where $\varepsilon_{\mathrm{CSS}}$ is the specified CSS tolerance. The switching procedure is expressed as

$$c_{z,T+1}(0,x) = P_{\mathrm{s}}[c_{z,T}(1,x)], \; q_{z,T+1}(0,x) = P_{\mathrm{s}}[q_{z,T}(1,x)], \quad T = 1, 2, \ldots,$$

where $P_{\mathrm{s}}[\cdot]$ is a columnwise switching operator, and T refers to the Tth period. In this model, the outputs of interest are the concentrations at the extract and the raffinate outlets in the CSS period, i.e.,

$$y(t,\mu) := (c^{\mathrm{E}}_{a,\mathrm{CSS}}(t;\mu), c^{\mathrm{R}}_{a,\mathrm{CSS}}(t;\mu), c^{\mathrm{E}}_{b,\mathrm{CSS}}(t;\mu), c^{\mathrm{R}}_{b,\mathrm{CSS}}(t;\mu)),$$

where $c^{\mathrm{E}}_{z,\mathrm{CSS}}(t;\mu)$, $c^{\mathrm{R}}_{z,\mathrm{CSS}}(t;\mu)$ are the CSS concentrations of c_z at the extract and the raffinate outlets, respectively, $z = a, b$. The interesting parameter domain is taken as $\mathcal{P} := [4.2, 4.7] \times [2.5, 3.0] \times [3.5, 4.0] \times [2.2, 2.7] \times [0.05, 0.1]$.

In this work, we consider an optimization problem of the SMB model as follows:

$$\begin{aligned} &\min_{\mu \in \mathcal{P}} f(\mu), \; f(\mu) = -Q_{\mathrm{f}}, \\ \text{s.t. } & Pu_{a,\min} - Pu_a(\mu) \le 0, \\ & Pu_{b,\min} - Pu_b(\mu) \le 0, \\ & Q_{\mathrm{I}} - Q_{\max} \le 0, \end{aligned}$$

where $Pu_a(\mu) = \frac{\int_0^1 c^{\mathrm{E}}_{a,\mathrm{CSS}}(t;\mu)\mathrm{d}t}{\int_0^1 c^{\mathrm{E}}_{a,\mathrm{CSS}}(t;\mu)\mathrm{d}t + \int_0^1 c^{\mathrm{E}}_{b,\mathrm{CSS}}(t;\mu)\mathrm{d}t}$, $Pu_b(\mu) = \frac{\int_0^1 c^{\mathrm{R}}_{b,\mathrm{CSS}}(t;\mu)\mathrm{d}t}{\int_0^1 c^{\mathrm{R}}_{a,\mathrm{CSS}}(t;\mu)\mathrm{d}t + \int_0^1 c^{\mathrm{R}}_{b,\mathrm{CSS}}(t;\mu)\mathrm{d}t}$ are the product purities at the extract and the raffinate outlets. Solving such an optimization problem is time-consuming because it takes many iterations to converge and each iteration needs to simulate the original FOM till the CSS. We now use MOR to tackle this problem.

6.4.2. Reduced-order modeling of the SMB model. We use the finite volume discretization to construct the FOM as follows:

$$\begin{cases} A_{\mu,z} c_z^{n+1} = B_{\mu,z} c_z^n + r_z^n + t_{\mathrm{s}} \kappa_z q_z^n, \\ \quad q_z^{n+1} = (1 - t_{\mathrm{s}} \kappa_z \Delta t) q_z^n + t_{\mathrm{s}} \kappa_z H_z \Delta t c_z^n. \end{cases}$$

Note that the coefficient matrices $A_{\mu,z}$ and $B_{\mu,z}$ are time independent compared to the general form in (2.2), r_z^n comes from the feed conditions, and it does not dependent on the field variables. Let $V_{c_z} \in \mathbb{R}^{\mathcal{N}\times N_{c_z}}, V_{q_z} \in \mathbb{R}^{\mathcal{N}\times N_{q_z}}$ be the RB matrices for the field variables c_z, q_z, respectively, and $\hat{c}_z^n := V_{c_z} a_{c_z}^n$, $\hat{q}_z^n := V_{q_z} a_{q_z}^n$ be the reduced approximations of c_z^n and q_z^n, accordingly. Here $\mathcal{N}$ is the number of degrees of freedom of the FOM for every field variable, and N_{c_z}, N_{q_z} are the column numbers of the projection matrices for c_z, q_z, respectively, $z = a, b$. By using Galerkin projection, the ROM is formulated as

$$\begin{cases} \hat{A}_{\mu,z} a_{c_z}^{n+1} = \hat{B}_{\mu,z} a_{c_z}^n + \hat{r}_z + t_{\mathrm{s}} \kappa_z \hat{D}_z a_{q_z}^n, \\ \quad\;\; a_{q_z}^{n+1} = (1 - t_{\mathrm{s}} \kappa_z \Delta t) a_{q_z}^n + t_{\mathrm{s}} \kappa_z H_z \Delta t \hat{D}_z^T a_{c_z}^n, \end{cases}$$

where $\hat{A}_{\mu,z} = V_{c_z}^T A_{\mu,z} V_{c_z}$, $\hat{B}_{\mu,z} = V_{c_z}^T B_{\mu,z} V_{c_z}$, $\hat{r}_z = V_{c_z}^T r_z^n$, and $\hat{D}_z = V_{c_z}^T V_{q_z}$ are the reduced matrices, and $a_{c_z}^n$, $a_{q_z}^n$ are the unknowns of the ROM.

6.4.3. Results. The training set $\mathcal{P}_{\mathrm{train}}$ consists of 150 sample points randomly distributed in the parameter domain. When using the POD-Greedy algorithm to generate the projection matrices V_{c_z} and V_{q_z}, the snapshots are taken from one CSS period rather than the transient process, since only the products in the CSS period are of interest. The number of time steps in one period is still large ($\mathcal{O}(10^3)$), which is larger than the dimension of the spatial discretization. To efficiently construct the RB, the technique of adaptive snapshot selection [40] is employed. There are four outputs in this model. The similar strategy as we took for the batch chromatographic model is employed, i.e., compute an error estimation for each output and take the maximum as the error indicator for every parameter.

The column numbers of the projection matrices ($V_{c_z}, V_{q_z},\ z = a, b$) are $82, 83, 83$, 83, respectively, when the tolerance $\varepsilon_{\mathrm{ROM}}$ is taken as 1.0×10^{-3}. Figure 11 shows the behavior of the output error estimation and the corresponding true error during the extension of the RB. The output error estimation goes below the prespecified tolerance when the maximal number of the RB reaches 83. However, it does not decay smoothly, unlike the previous examples. This is because the average ratio $\tilde{\rho}_N^\star$ oscillates during the RB extension process, as illustrated in Figure 12. In fact, the oscillation in $\tilde{\rho}_N^\star$ will result in the oscillation in the output error estimation, since we use $\tilde{\rho}_N^\star$ as the estimate of $\rho^{n+1}(\mu)$ in (4.18). This is probably due to the multiswitching procedure, which causes extreme difficulty for MOR, because some error might be introduced after each switch and this error is hard to measure. In addition, the Péclet number in this model is 500, which is challenging for MOR, as we have observed in section 6.2.

Before the ROM is used to solve the underlying optimization problem, we validate its accuracy by performing full and reduced simulations over a test set with 200 random samples of parameters in the parameter domain. The maximal error and average runtime are shown in Table 5. It is seen that the maximal true error is 1.1×10^{-4} and is smaller than the prespecified tolerance. The average runtime is largely reduced and the speedup factor is 7.

Finally, we study the performance of the ROM-based optimization. Table 6 shows the results using the constraints $Pu_{a,\min} = 99.0\%$, $Pu_{b,\min} = 99.0\%$, $Q_{\max} = 0.50$ ml/s. We use the gradient-free optimizer, the subroutine NLOPT_LN_COBYLA [21], to solve the underlying optimization problem. It is a local optimizer. To verify the robustness of the optimizer to this problem, different initial guesses have been employed and almost the same optimal solution is obtained. The tolerance for the optimization is taken as $\epsilon_{\mathrm{opt}} = 1.0 \times 10^{-4}$. From Table 6, it is seen that the ROM-based

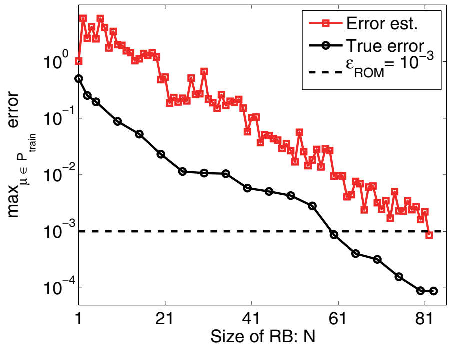


Fig. 11. *Decay of the error estimation and the corresponding true error during the RB construction process for the SMB model.*

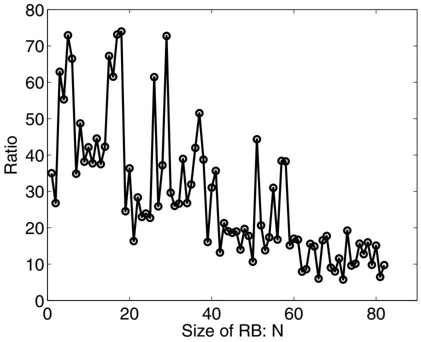


Fig. 12. *Behavior of the average ratio $\tilde{\rho}_N^{\star}$ during the RB construction process for the SMB model.*

Table 5

Comparison of runtime for the full and reduced simulations of the SMB model over a validation set $\mathcal{P}_{\mathrm{val}}$ with 200 random sample points. $\varepsilon_{\mathrm{ROM}} = 1.0 \times 10^{-3}$.

Model	Maximal error	Average runtime (s)/SpF
FOM ($\mathcal{N} = 800$)	–	349.5 / -
ROM	1.1×10^{-4}	46.7 / 7

Table 6

Comparison of the optimization of SMB chromatography based on the FOM and the ROM.

		Initial guess	FOM-Opt.	ROM-Opt.
Objective	Q_{f} (ml/s)	0.07	0.0745	0.0745
Optimal solution	m_{I}	4.50	4.3269	4.3271
	m_{II}	2.90	2.8599	2.8603
	m_{III}	3.50	3.6036	3.6039
	m_{IV}	2.30	2.3468	2.3685
	Q_{f} (ml/s)	0.07	0.0745	0.0745
Constraints	Pu_a	98.9%	99.0%	99.0%
	Pu_b	99.5%	99.0%	99.0%
	Q_{I} (ml/s)	0.4161	0.4997	0.4998
#Iterations			71	79
Runtime (h) / SpF			5.13 / -	0.82 / 6

optimization is very successful. The runtime for solving the optimization problem is largely reduced while the optimal solutions are almost the same as those of the FOM-based optimization. The speedup factor is 6.

7. Conclusions. We have presented an efficient a posteriori output error estimation for MOR of parametrized nonlinear evolution equations. The new error estimation is suitable for projection-based MOR methods. Certainly, it can be applied to the linear evolution problem as well, as shown in sections 6.1 and 6.4.

The proposed output error estimation is sharp and computationally efficient and can be applied to a broad class of evolution equations. In particular, it is applicable to problems with a long time evolution process. In contrast, the existing error bound often fails due to its continuous accumulation over time. Numerical results have illustrated the behaviors of the error estimations and have shown the efficiency of the newly proposed error estimation. However, since the quantity $\rho^n(\mu)$ in the error bound needs to be estimated, the rigorousness of the error estimate cannot be guaranteed, though loss of the "upper bound" property was not observed for the examples tested in this manuscript. A convincing and more reliable estimation of $\rho^n(\mu)$ in the error bound deserves further investigation.

REFERENCES

[1] D. Amsallem and C. Farhat, *Stabilization of projection-based reduced-order models*, Internat. J. Numer. Methods Engrg., 91 (2012), pp. 358–377.

[2] Z. Bai, *Krylov subspace techniques for reduced-order modeling of large-scale dynamical systems*, Appl. Numer. Math., 43 (2002), pp. 9–44.

[3] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera, *An 'empirical interpolation' method: Application to efficient reduced-basis discretization of partial differential equations*, C. R. Math. Acad. Sci. Paris, 339 (2004), pp. 667–672.

[4] P. Benner and L. Feng, *A robust algorithm for parametric model order reduction based on implicit moment matching*, in Reduced Order Methods for Modeling and Computational Reduction, A. Quarteroni and G. Rozza, eds., MS&A Model. Simul. Appl. 9, Springer, Cham, Switzerland, 2014, pp. 159–185.

[5] P. Benner, L. Feng, S. Li, and Y. Zhang, *Reduced-order modeling and ROM-based optimization of batch chromatography*, in Numerical Mathematics and Advanced Applications, ENUMATH 2013, A. Abdulle, S. Deparis, D. Kressner, F. Nobile, and M. Picasso, eds., Lect. Notes Comput. Sci. Eng. 103, Springer, New York, 2015, pp. 427–435.

[6] T. Bui-Thanh, M. Damodaran, and K. Willcox, *Proper Orthogonal Decomposition Extensions for Parametric Applications in Compressible Aerodynamics*, AIAA paper 4213, AIAA, Reston, VA, 2003.

[7] T. Bui-Thanh, K. Willcox, O. Ghattas, and B. van Bloemen Waanders, *Goal-oriented, model-constrained optimization for reduction of large-scale systems*, J. Comput. Phys., 224 (2007), pp. 880–896.

[8] S. Chaturantabut and D. C. Sorensen, *Nonlinear model reduction via discrete empirical interpolation*, SIAM J. Sci. Comput., 32 (2010), pp. 2737–2764.

[9] M. Drohmann, B. Haasdonk, and M. Ohlberger, *Reduced basis approximation for nonlinear parametrized evolution equations based on empirical operator interpolation*, SIAM J. Sci. Comput., 34 (2012), pp. 937–969.

[10] L. Feng, A. C. Antoulas, and P. Benner, *Some a Posteriori Error Bounds for Reduced Order Modelling of (Non-)Parametrized Linear Systems*, Preprint MPIMD/15-17, Max Planck Institute Magdeburg, Germany, 2015.

[11] L. Feng, P. Benner, and A. C. Antoulas, *Automatic generation of reduced order models for linear parametric systems*, in Proceedings of European Consortium for Mathematics in Industry (ECMI 2014), to appear.

[12] R. W. Freund, *Model reduction methods based on Krylov subspaces*, Acta Numer., 12 (2003), pp. 267–319.

[13] M. A. Grepl, *Reduced-Basis Approximation A Posteriori Error Estimation for Parabolic Partial Differential Equations*, Ph.D. thesis, MIT, Cambridge, MA, 2005.

[14] M. A. Grepl, *Certified reduced basis methods for nonaffine linear time-varying and nonlinear parabolic partial differential equations*, Math. Models Methods Appl. Sci., 22 (2012).

[15] M. A. Grepl, Y. Maday, N. C. Nguyen, and A. T. Patera, *Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations*, M2AN Math. Model. Numer. Anal., 41 (2007), pp. 575–605.

[16] M. A. Grepl and A. T. Patera, *A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations*, M2AN Math. Model. Numer. Anal., 39 (2005), pp. 157–181.

[17] B. Haasdonk and M. Ohlberger, *Reduced basis method for finite volume approximations of parametrized linear evolution equations*, M2AN Math. Model. Numer. Anal., 42 (2008), pp. 277–302.

[18] B. Haasdonk and M. Ohlberger, *Efficient reduced models and a posteriori error estimation for parametrized dynamical systems by offline/online decomposition*, Math. Comput. Model. Dyn. Syst., 17 (2011), pp. 145–161.

[19] M. W. Hess and P. Benner, *A reduced basis method for microwave semiconductor devices with geometric variations*, COMPEL, 33 (2014), pp. 1071–1081.

[20] R. H. Hoppe and Z. Liu, *Snapshot location by error equilibration in proper orthogonal decomposition for linear and semilinear parabolic partial differential equations*, J. Numer. Math., 22 (2014), pp. 1–32.

[21] S. G. Johnson, *The NLopt Nonlinear-Optimization Package*, http://ab-initio.mit.edu/nlopt.

[22] S. Li, L. Feng, P. Benner, and A. Seidel-Morgenstern, *Using surrogate models for efficient optimization of simulated moving bed chromatography*, Comput. Chem. Eng., 67 (2014), pp. 121–132.

[23] M. Mazzotti, G. Storti, and M. Morbidelli, *Optimal operation of simulated moving bed units for nonlinear chromatographic separations*, J. Chromatogr. A, 769 (1997), pp. 3–24.

[24] K. W. Morton and D. F. Mayers, *Numerical Solution of Partial Differential Equations: An Introduction*, Cambridge University Press, Cambridge, UK, 2005.

[25] N. C. Nguyen, G. Rozza, and A. T. Patera, *Reduced basis approximation and a posteriori error estimation for the time-dependent viscous Burgers' equation*, Calcolo, 46 (2009), pp. 157–185.

[26] A. K. Noor and J. M. Peters, *Reduced basis technique for nonlinear analysis of structures*, AIAA J., 18 (1980), pp. 145–161.

[27] A. T. Patera and G. Rozza, *Reduced Basis Approximation and a Posteriori Error Estimation for Parametrized Partial Differential Equations*, MIT, Cambridge, MA, 2007; also available from http://augustine.mit.edu/methodology/methodology_book.htm.

[28] T. A. Porsching, *Estimation of the error in the reduced basis method solution of nonlinear equations*, Math. Comput., 45 (1985), pp. 487–496.

[29] C. Prud'homme, D. V. Rovas, K. Veroy, L. Machiels, Y. Maday, A. T. Patera, and G. Turinici, *Reliable real-time solution of parametrized partial differential equations: Reduced-basis output bound methods*, J. Fluids Engrg., 124 (2002), pp. 70–80.

[30] A. Rajendran, G. Paredes, and M. Mazzotti, *Simulated moving bed chromatography for the separation of enantiomers*, J. Chromatogr. A, 1216 (2009), pp. 709–738.

[31] T. Reis and T. Stykel, *Stability analysis and model order reduction of coupled systems*, Math. Comput. Model. Dyn. Syst., 13 (2007), pp. 413–436.

[32] D. V. Rovas, *Reduced-Basis Output Bound Methods for Parametrized Partial Differential Equations*, Ph.D. thesis, MIT, Cambridge, MA, 2003.

[33] G. Rozza and K. Veroy, *On the stability of the reduced basis method for Stokes equations in parametrized domains*, Comput. Methods Appl. Mech. Engrg., 196 (2007), pp. 1244–1260.

[34] K. Urban and A. T. Patera, *An improved error bound for reduced basis approximation of linear parabolic problems*, Math. Comp., 83 (2014), pp. 1599–1615.

[35] K. Willcox, *Unsteady flow sensing and estimation via the gappy proper orthogonal decomposition*, Comput. Fluids, 35 (2006), pp. 208–226.

[36] K. Willcox and J. Peraire, *Balanced model reduction via the proper orthogonal decomposition*, AIAA J., 40 (2002).

[37] M. Yano, *A space-time Petrov-Galerkin certified reduced basis method: Application to the Boussinesq equations*, SIAM J. Sci. Comput., 36 (2014), pp. 232–266.

[38] M. Yano and A. T. Patera, *A space-time variational approach to hydrodynamic stability theory*, Proc. R. Soc. Lond. Ser. A Math., Phys. Eng. Sci., 469 (2013), 0036.

[39] M. Yano, A. T. Patera, and K. Urban, *A space-time hp-interpolation-based certified reduced basis method for Burgers' equation*, Math. Models Methods Appl. Sci., 24 (2014), pp. 1903–1935.

[40] Y. Zhang, L. Feng, S. Li, and P. Benner, *Accelerating PDE constrained optimization by the reduced basis method: Application to batch chromatography*, Internat. J. Numer. Methods Engrg., 104 (2015), pp. 983–1007.