Max Planck Institute Magdeburg
Preprints

Yongjin Zhang, Lihong Feng, Suzhou Li and Peter Benner

An efficient output error bound for model order reduction of parametrized evolution equations

MPIMD/14-22 **December 2, 2014**

Imprint:

Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg

Publisher:
Max Planck Institute for
Dynamics of Complex Technical Systems

Address:
Max Planck Institute for
Dynamics of Complex Technical Systems
Sandtorstr. 1
39106 Magdeburg

http://www.mpi-magdeburg.mpg.de/preprints/

Abstract

In this work we present an efficient *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, the output error bound can be estimated sharply. Such an error bound successfully avoids the accumulation of the residual over time, which is a common drawback in the existing error estimation for time-stepping schemes. The proposed error bound is applied to three kinds of problems. The first one is the unsteady viscous Burgers' equation, an academic benchmark of nonlinear evolution equations in fluid dynamics often used as first test case to validate nonlinear model order reduction methods. The other two problems arise from chromatographic separation processes. They are batch chromatography with (nonlinear) bi-Langmuir isotherm equations, and continuous simulated moving bed chromatography with linear isotherm equations, where periodic switching is involved. Numerical experiments demonstrate the performance and efficiency of the proposed error bound. Optimization based on the resulting reduced-order models is successful in term of accuracy and the runtime for getting the optimal solution.

Keywords: model order reduction, output error bound, empirical interpolation, dual system, evolution equations, optimization, chromatography

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 etc. 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 the 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 the *a posteriori* error estimation for the reduced basis (RB) method started from [20], and has been followed by many others [10, 11, 12, 18, 19, 21, 23]. Notably, these error estimations are all derived in the functional space in the framework of the finite element (FE) discretization except for [11]. 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 [11] is derived in the framework of the finite volume (FV) discretization and it is 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 [16], and error estimations for projection based MOR methods have been studied in recent years, see e.g. [10, 11, 24]. The error estimator, however, may loose 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, the original system. The output error for the primal system can 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 resulting from certain assumptions. These assumptions can be reasonably fulfilled, which will be shown in the numerical experiments. The proposed error bound for the output is independent of the choice of the projection matrix or projection subspace, so it is independent of the chosen MOR method.

The idea for the proposed error bound originates from the recent study in [7, 8], where some error bounds are derived for linear time-invariant systems. The main difference of the proposed error estimation from that in [7, 8] 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 reduced basis method [11, 19, 24], and is valid for nonlinear parametric systems, whereas, the error bound in [7, 8] is an error estimation for the transfer function of the reduced model, 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 [1, 3, 9].

The proposed error estimation is applied to three different evolution problems. The first one is the unsteady viscous Burgers' equation, which is used to illustrate that the proposed error estimation is applicable to MOR of a broad class of evolution problems. The second one is batch chromatography with bi-Langmuir isotherm equations, which is a nonlinear parametric evolution problem. The last one is continuous chromatography with linear isotherm equations, which is a linear parametric periodic switching system. Note that the latter two problems arise from chromatographic separation processes in chemical engineering. The resulting reduced models are employed to solve the underlying optimization problems.

The paper is organized as follows. Section 2 shows the projection based model order reduction 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 [11] used in reduced basis methods, where the proposed error estimation will be used to generate the reduced basis (projection matrix). Numerical examples are given in Section 6. Conclusions are drawn in Section 7.

2 Model order reduction 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 $\mathcal{P} \subset \mathbb{R}^p$,

$$\partial_t u(t, x; \mu) + \mathcal{L}[u(t, x; \mu)] = 0, \quad t \in [0, T], \quad x \in \Omega, \quad \mu \in \mathcal{P}, \tag{1}$$

where $\mathcal{L}[\cdot]$ 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 equation (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 following form,

$$A_\mu^{(n)} u^{n+1}(\mu) = B_\mu^{(n)} u^n(\mu) + g(u^n(\mu), \mu), \tag{2}$$

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 $g(\cdot)$ 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 model oder reduction, 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) is called full order model (FOM).

Solving such a FOM repeatedly under parameter variations is time consuming or even prohibitive in a multi-query context, e.g. optimization, real time control. Besides the improvements of the computing resources, MOR has been developed as a useful tool to handle this kind of problems and plays an important role in an efficient solution process for parametric systems. In the following subsections, we address the projection based MOR method and the simulation of the reduced-order model.

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 $u^n(\mu) \approx V a^n(\mu)$, and employing Petrov-Galerkin projection with W,

$$\hat{A}_\mu^{(n)} a^{n+1}(\mu) = \hat{B}_\mu^{(n)} a^n(\mu) + W^T g(V a^n(\mu)), \tag{3}$$

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 in the ROM.

Notably, the number of degrees of freedom of the ROM in (3) is usually much less than that of the FOM in (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 (3); it is required that the evaluation of $\hat{A}_\mu^{(n)}, \hat{B}_\mu^{(n)}$ and $W^T g(Va^n(\mu))$ is achieved 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 reduced basis MOR method, an offline-online decomposition strategy is often employed to achieve this goal. Similar to related studies, assume that the matrices $A_\mu^{(n)}$ and $B_\mu^{(n)}$ in (2) can be written in a separable way, the so-called affine form, i.e.,

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

where A_j, B_k are constant matrices, ξ_μ^n, ζ_μ^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

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

where $\hat{A}_j = W^T A_j V$ and $\hat{B}_k = W^T B_k V$, $j = 1, \ldots, n_a, k = 1, \ldots, n_b$. 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 evaluations of $\hat{A}_\mu^{(n)}$ and $\hat{B}_\mu^{(n)}$ at μ are independent of the full dimension $\mathcal{N}$ when simulating the ROM. However, the computation of the last term of (3), $W^T g(Va^n(\mu))$, cannot be done analogously because of the nonlinearity or non-affinity of g. To achieve an efficient offline-online computation, empirical (operator) interpolation [2, 6] or the discrete empirical interpolation method [5] can be employed. For example, $g(\hat{u}^n(\mu)) \approx \hat{g}^n(\mu) := \mathcal{I}_M[g(\hat{u}^n(\mu))] = S\beta^n(\mu)$, where $S \in \mathbb{R}^{\mathcal{N}\times M}$ $(M \ll \mathcal{N})$ is the precomputed parameter-independent basis, and $\beta^n(\mu) \in \mathbb{R}^M$ is the corresponding vector of coefficients. As a result, a low dimensional ROM is obtained as below,

$$\hat{A}_\mu^{(n)} a^{n+1}(\mu) = \hat{B}_\mu^{(n)} a^n(\mu) + \hat{G}\beta^n(\mu), \tag{4}$$

where $\hat{G} = W^T S$ is precomputed. With this reduced model (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,

$$\|Z\| := \sup_{Z\in\mathbb{R}^{\mathcal{N}},v\neq 0} \frac{\|Zv\|}{\|v\|} = \max_{\|v\|=1} \|Zv\|.$$

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 [6, 10, 11, 19, 24]. Motivated by the error estimation for the field variable in [6], an error estimation for the field variable in the vector space and a corresponding output error bound were proposed in [24]. In [24], the matrices $A_\mu^{(n)}$ and $B_\mu^{(n)}$ are independent of the parameter μ, i.e., they are constant matrices $A^{(n)}$ and $B^{(n)}$. The error estimations, however, can be easily extended to the case that the coefficient matrices $A_\mu^{(n)}$ and $B_\mu^{(n)}$ are parameter dependent. In addition, the nonlinear term $g(\cdot)$ was tackled by using the empirical interpolation (EI) [2]. By defining the residual

$$r^{n+1}(\mu) := B_\mu^{(n)}\hat{u}^n(\mu) + \mathcal{I}_M[g(\hat{u}^n(\mu))] - A_\mu^{(n)}\hat{u}^{n+1}(\mu) \tag{5}$$

for the ROM in (4), the error estimations in [24] are summarized as below.

Proposition 3.1. *Assume that the operator $g\colon \mathbb{R}^{\mathcal{N}} \to \mathbb{R}^{\mathcal{N}}$ is Lipschitz continuous, i.e., there exists a positive constant L_g, such that*

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

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

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

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

$$y(u^n(\mu)) = Pu^n(\mu), \tag{6}$$

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

$$\|e^n(\mu)\| \le \eta_{N,M}^n(\mu) := R^{n-1} + \sum_{k=0}^{n-2}\left(\prod_{j=k+1}^{n-1} \mathtt{G}_{\mathrm{F},\mu}^{(j)}\right) R^k,\ n = 1, \ldots, K, \tag{7}$$

$$\begin{aligned}\left\|e_{\mathrm{O}}^{n+1}(\mu)\right\| &\le \tilde{\eta}_{N,M}^{n+1}(\mu)\\ &:= \mathtt{G}_{\mathrm{O},\mu}^{(n)}\eta_{N,M}^n(\mu) + \|P(A_\mu^{(n)})^{-1}\|\epsilon_{\mathrm{EI}}^n(\mu) + \|P\|\|(A_\mu^{(n)})^{-1}r^{n+1}(\mu)\|,\end{aligned} \tag{8}$$

where

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

and

$$
\epsilon_{\mathrm{EI}}^n(\mu) = \sum_{m=M+1}^{M+M'} \|s_m\beta_m^n(\mu)\|\,, \tag{9}
$$

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

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 caused by the interpolation over all the previous time steps. This implies that both error bounds are accumulated over time. As a result, they may loose sharpness when a large number of time steps are needed, e.g. in the simulation of batch chromatography [24]. The same phenomenon also exists in the error estimation in [6]. Similar observations are also reported in [17]. To circumvent the problem, we propose a new output error bound for the reduced model 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 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

$$
A_\mu^{(n)}u^{n+1}(\mu) = b(u^n(\mu), \mu), \tag{10}
$$

where $A_\mu^{(n)}$ is assumed to be nonsingular for all $\mu \in \mathcal{P}$, $u^n(\mu) \in \mathbb{R}^{\mathcal{N}}$ is the numerical solution at time $t = t^n$, $b : \mathbb{R}^{\mathcal{N}} \to \mathbb{R}^{\mathcal{N}}$ can be linear or nonlinear, e.g. the right hand side of the equation in (2). The output of interest is expressed as in (6). Here, we temporally assume $N_{\mathrm{o}} = 1$ for simplicity. The extension to the multiple output case is possible, see Remark 4.7.

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

$$
\begin{cases}
A_\mu^{(n)}u^{n+1}(\mu) = b(u^n(\mu), \mu),\\
\quad y^{n+1}(\mu) = Pu^{n+1}(\mu),
\end{cases} \tag{11}
$$

and introduce a corresponding dual system as follows:

$$
\begin{cases}
(A_\mu^{(n)})^T u_{\mathrm{du}}^{n+1}(\mu) = -P^T,\\
\quad y_{\mathrm{du}}^{n+1}(\mu) = P_{\mathrm{du}}u_{\mathrm{du}}^{n+1}(\mu).
\end{cases} \tag{12}
$$

Here, $P_{\mathrm{du}} \in \mathbb{R}^{N_o \times \mathcal{N}}$ can be freely defined, as the output of the dual system does not contribute to the derivation of the error bound. 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 system, respectively. Using Petrov-Galerkin projection, we have the reduced models for the primal and dual systems,

$$\begin{cases} \hat{A}^{(n)}_{\mu,\mathrm{pr}} a^{n+1}_{\mathrm{pr}}(\mu) = W^T_{\mathrm{pr}} b(\hat{u}^n(\mu), \mu), \\ \hat{y}^{n+1}(\mu) = P\hat{u}^{n+1}(\mu), \end{cases} \tag{13}$$

$$\begin{cases} \hat{A}^{(n)}_{\mu,\mathrm{du}} a^{n+1}_{\mathrm{du}}(\mu) = -W^T_{\mathrm{du}} P^T, \\ \hat{y}^{n+1}_{\mathrm{du}}(\mu) = P_{\mathrm{du}} \hat{u}^{n+1}_{\mathrm{du}}(\mu), \end{cases} \tag{14}$$

where $\hat{A}^{(n)}_{\mu,\mathrm{pr}} = W^T_{\mathrm{pr}} A^{(n)}_{\mu} V_{\mathrm{pr}}$, $\hat{A}^{(n)}_{\mu,\mathrm{du}} = W^T_{\mathrm{du}} (A^{(n)}_{\mu})^T V_{\mathrm{du}}$, and $\hat{u}^n(\mu) = V_{\mathrm{pr}} a^n_{\mathrm{pr}}(\mu)$, $\hat{u}^n_{\mathrm{du}}(\mu) = V_{\mathrm{du}} a^n_{\mathrm{du}}(\mu)$ are the approximations to $u^n(\mu)$ and $u^n_{\mathrm{du}}(\mu)$, respectively. The vectors $a^n_{\mathrm{pr}}(\mu)$ and $a^n_{\mathrm{du}}(\mu)$ are the unknowns in the reduced systems above. The residuals for both systems read,

$$r^{n+1}_{\mathrm{pr}} := r^{n+1}_{\mathrm{pr}}(\mu) = b(\hat{u}^n(\mu), \mu) - A^{(n)}_{\mu} \hat{u}^{n+1}(\mu), \tag{15}$$

$$r^{n+1}_{\mathrm{du}} := r^{n+1}_{\mathrm{du}}(\mu) = -P^T - (A^{(n)}_{\mu})^T \hat{u}^{n+1}_{\mathrm{du}}(\mu), \tag{16}$$

respectively. Define an auxiliary vector

$$\tilde{r}^{n+1}_{\mathrm{pr}} := b(u^n(\mu), \mu) - A^{(n)}_{\mu} \hat{u}^{n+1}(\mu) = A^{(n)}_{\mu} u^{n+1}(\mu) - A^{(n)}_{\mu} \hat{u}^{n+1}(\mu). \tag{17}$$

Notice that the only difference of $\tilde{r}^{n+1}_{\mathrm{pr}}$ from r^{n+1}_{pr} is that $b(\hat{u}^n(\mu), \mu)$ in (15) is replaced by $b(u^n(\mu), \mu)$ in (17), so that we have a direct relation between $\tilde{r}^{n+1}_{\mathrm{pr}}$ 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.

Theorem 4.1. *For the systems (11) and (13), assume that $A^{(n)}_{\mu}$ is invertible for any $\mu \in \mathcal{P}$, and the residual r^{n+1}_{pr} is "close" to $\tilde{r}^{n+1}_{\mathrm{pr}}$ in the sense that for each well chosen $V_{\mathrm{pr}} \in \mathbb{R}^{\mathcal{N} \times N}$, there exists a constant $\tilde{\epsilon} := \tilde{\epsilon}(N) > 0$, such that*

$$\|r^{n+1}_{\mathrm{pr}} - \tilde{r}^{n+1}_{\mathrm{pr}}\| \leq \tilde{\epsilon}. \tag{18}$$

Then the output error $e^{n+1}_{\mathrm{O}}(\mu) = y^{n+1}(\mu) - \hat{y}^{n+1}(\mu)$ at the time instance t^{n+1} satisfies

$$\|e^{n+1}_{\mathrm{O}}(\mu)\| \leq \|(A^{(n)}_{\mu})^{-T}\| \|r^{n+1}_{\mathrm{du}}\| \|r^{n+1}_{\mathrm{pr}}\| + \|\hat{u}^{n+1}_{\mathrm{du}}(\mu)\| \|r^{n+1}_{\mathrm{pr}}\| + \epsilon, \tag{19}$$

where $\epsilon = \left(\|(A^{(n)}_{\mu})^{-T}\| \|r^{n+1}_{\mathrm{du}}\| + \|\hat{u}^{n+1}_{\mathrm{du}}(\mu)\| \right) \tilde{\epsilon}$.

Proof. By the dual system (12), we have

$$\left(u^{n+1}(\mu) - \hat{u}^{n+1}(\mu)\right)^T (A^{(n)}_{\mu})^T u^{n+1}_{\mathrm{du}}(\mu) = -\left(u^{n+1}(\mu) - \hat{u}^{n+1}(\mu)\right)^T P^T.$$

Transposing this equation, we obtain

$$\left(u^{n+1}_{\mathrm{du}}(\mu)\right)^T A^{(n)}_{\mu} \left(u^{n+1}(\mu) - \hat{u}^{n+1}(\mu)\right) = -P\left(\left(u^{n+1}(\mu) - \hat{u}^{n+1}(\mu)\right)\right). \tag{20}$$

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

$$\tilde{r}_{\text{pr}}^{n+1} = A_{\mu}^{(n)} \left(u^{n+1}(\mu) - \hat{u}^{n+1}(\mu) \right). \tag{21}$$

Left-multiplying both sides of (21) by $\left(u_{\text{du}}^{n+1}(\mu)\right)^T$ yields

$$\left(u_{\text{du}}^{n+1}(\mu)\right)^T \tilde{r}_{\text{pr}}^{n+1} = \left(u_{\text{du}}^{n+1}(\mu)\right)^T A_{\mu}^{(n)} \left(u^{n+1}(\mu) - \hat{u}^{n+1}(\mu) \right). \tag{22}$$

Combining (20) and (22), 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

$$\begin{aligned} |y^{n+1}(\mu) - \tilde{y}^{n+1}(\mu)| &= |Pu^{n+1}(\mu) - P\hat{u}^{n+1}(\mu) + \left(\hat{u}_{\text{du}}^{n+1}(\mu)\right)^T \tilde{r}_{\text{pr}}^{n+1}| \\ &= | - \left(u_{\text{du}}^{n+1}(\mu)\right)^T \tilde{r}_{\text{pr}}^{n+1} + \left(\hat{u}_{\text{du}}^{n+1}(\mu)\right)^T \tilde{r}_{\text{pr}}^{n+1}| \\ &= | - \left(u_{\text{du}}^{n+1}(\mu) - \hat{u}_{\text{du}}^{n+1}(\mu)\right)^T \tilde{r}_{\text{pr}}^{n+1}| \\ &\leq \|u_{\text{du}}^{n+1}(\mu) - \hat{u}_{\text{du}}^{n+1}(\mu)\| \|\tilde{r}_{\text{pr}}^{n+1}\|. \end{aligned} \tag{23}$$

By the definition of the residual in (16) and the dual system in (12), we have

$$\begin{aligned} r_{\text{du}}^{n+1} &= -P^T - (A_{\mu}^{(n)})^T \hat{u}_{\text{du}}^{n+1}(\mu) \\ &= (A_{\mu}^{(n)})^T u_{\text{du}}^{n+1}(\mu) - (A_{\mu}^{(n)})^T \hat{u}_{\text{du}}^{n+1}(\mu) \\ &= (A_{\mu}^{(n)})^T (u_{\text{du}}^{n+1}(\mu) - \hat{u}_{\text{du}}^{n+1}(\mu)). \end{aligned} \tag{24}$$

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

$$u_{\text{du}}^{n+1}(\mu) - \hat{u}_{\text{du}}^{n+1}(\mu) = (A_{\mu}^{(n)})^{-T} r_{\text{du}}^{n+1}. \tag{25}$$

Combining (23) and (25), we obtain

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

Thus

$$\begin{aligned} |y^{n+1}(\mu) - \hat{y}^{n+1}(\mu)| &= |y^{n+1}(\mu) - \tilde{y}^{n+1}(\mu) - \left(\hat{u}_{\text{du}}^{n+1}(\mu)\right)^T \tilde{r}_{\text{pr}}^{n+1}| \\ &\leq |y^{n+1}(\mu) - \tilde{y}^{n+1}(\mu)| + |\left(\hat{u}_{\text{du}}^{n+1}(\mu)\right)^T \tilde{r}_{\text{pr}}^{n+1}| \\ &\leq \|(A_{\mu}^{(n)})^{-T}\| \|r_{\text{du}}^{n+1}\| \|\tilde{r}_{\text{pr}}^{n+1}\| + \|\left(\hat{u}_{\text{du}}^{n+1}(\mu)\right)^T\| \|\tilde{r}_{\text{pr}}^{n+1}\|. \end{aligned} \tag{26}$$

Due to the assumption (18), we have

$$\|\tilde{r}_{\text{pr}}^{n+1}\| \leq \|r_{\text{pr}}^{n+1}\| + \tilde{\epsilon}. \tag{27}$$

Substituting (27) into (26) yields the proposed error bound (19). □

With suitable assumptions, we have the following corollary, where a ratio between $\|\tilde{r}_{\mathrm{pr}}^{n+1}\|$ and $\|r_{\mathrm{pr}}^{n+1}\|$ is given, so that the quantity ϵ is removed from the error bound in (19).

Corollary 4.2. *Under the assumptions of Theorem 4.1, for* $\{\|\tilde{r}_{\mathrm{pr}}^{n+1}\|\}$ *and* $\{\|r_{\mathrm{pr}}^{n+1}\|\}$, *assume that there exist positive constants* $\underline{\alpha}, \bar{\alpha}, \underline{\beta}, \bar{\beta}$, *such that*

$$\underline{\alpha} \leq \frac{\|\tilde{r}_{\mathrm{pr}}^{n+1}\|}{\|\tilde{r}_{\mathrm{pr}}^{n}\|} \leq \bar{\alpha}, \tag{28a}$$

$$\underline{\beta} \leq \frac{\|r_{\mathrm{pr}}^{n+1}\|}{\|r_{\mathrm{pr}}^{n}\|} \leq \bar{\beta}, \tag{28b}$$

for $n = 1, \ldots, K-1$. *Assume the operator* $b(\cdot)$ *is Lipschitz continuous, i.e., there exist a positive constant* L_b, *such that*

$$\|b(u_1) - b(u_2)\| \leq L_b \|u_1 - u_2\|, \quad u_1, u_2 \in \mathcal{W}^{\mathcal{N}}, \tag{29}$$

and L_b *satisfies the inequality* $L_b \|(A_\mu^{(n)})^{-1}\| < \underline{\alpha}$, *for all* $\mu \in \mathcal{P}$. *Then*

$$\|\tilde{r}_{\mathrm{pr}}^{n+1}\| \leq \rho \|r_{\mathrm{pr}}^{n+1}\|, \tag{30}$$

and the error bound in (19) can be given as

$$\|e_{\mathrm{O}}^{n+1}(\mu)\| \leq \Delta^{n+1}(\mu) := \rho \left(\|(A_\mu^{(n)})^{-T}\| \|r_{\mathrm{du}}^{n+1}\| + \|\hat{u}_{\mathrm{du}}^{n+1}(\mu)\| \right) \|r_{\mathrm{pr}}^{n+1}\|, \tag{31}$$

where $\rho = \bar{\beta}/(\underline{\alpha} - L_b \|(A_\mu^{(n)})^{-1}\|)$.

Proof. By the first inequality in (28a) and the definition of $\tilde{r}_{\mathrm{pr}}^{n+1}$ in (17), we have

$$\begin{aligned}
\underline{\alpha}\|\tilde{r}_{\mathrm{pr}}^{n}\| - \|r_{\mathrm{pr}}^{n+1}\| &\leq \|\tilde{r}_{\mathrm{pr}}^{n+1}\| - \|r_{\mathrm{pr}}^{n+1}\| \\
&\leq \|r_{\mathrm{pr}}^{n+1} - \tilde{r}_{\mathrm{pr}}^{n+1}\| = \|b(\hat{u}^n(\mu)) - b(u^n(\mu))\| \\
&\leq L_b \|\hat{u}^n(\mu) - u^n(\mu)\| = L_b \|(A_\mu^{(n)})^{-1} \tilde{r}_{\mathrm{pr}}^{n}\| \\
&\leq L_b \|(A_\mu^{(n)})^{-1}\| \|\tilde{r}_{\mathrm{pr}}^{n}\|.
\end{aligned}$$

Taking the left hand side and the last term in the right hand side of the inequalities above, and using the inequality (28b), we obtain

$$\left(\underline{\alpha} - L_b \|(A_\mu^{(n)})^{-1}\| \right) \|\tilde{r}_{\mathrm{pr}}^{n}\| \leq \|r_{\mathrm{pr}}^{n+1}\| \leq \bar{\beta} \|r_{\mathrm{pr}}^{n}\|,$$

i.e.,

$$\|\tilde{r}_{\mathrm{pr}}^{n}\| \leq \frac{\bar{\beta}}{\underline{\alpha} - L_b \|(A_\mu^{(n)})^{-1}\|} \|r_{\mathrm{pr}}^{n}\| = \rho \|r_{\mathrm{pr}}^{n}\|, \quad n = 1, \ldots, K. \tag{32}$$

Substituting (32) into (26) yields the error estimation in (31). □

Remark 4.3. *When the operator $b(\cdot)$ is nonlinear, the empirical interpolation [2] can be employed. The ROM can be formulated following (4). In such a case, the term $\|r_{\text{pr}}^{n+1}\|$ in (19) or (31) can be further bounded using the EI error bound,*

$$\begin{aligned}\|r_{\text{pr}}^{n+1}\| &= \|B_\mu^{(n)}\hat{u}^n(\mu) + g(\hat{u}^n(\mu),\mu) - A_\mu^{(n)}\hat{u}^{n+1}(\mu)\| \\ &= \|B_\mu^{(n)}\hat{u}^n(\mu) + \mathcal{I}_M[g(\hat{u}^n(\mu),\mu)] - A_\mu^{(n)}\hat{u}^{n+1}(\mu) \\ &\qquad + g(\hat{u}^n(\mu),\mu) - \mathcal{I}_M[g(\hat{u}^n(\mu),\mu)]\| \\ &\le \|r^{n+1}(\mu)\| + \|g(\hat{u}^n(\mu),\mu) - \mathcal{I}_M[g(\hat{u}^n(\mu),\mu)]\| \\ &\le \|r^{n+1}(\mu)\| + \epsilon_{\text{EI}}^n(\mu),\end{aligned} \tag{33}$$

where $r^{n+1}(\mu)$ is defined in (5), and $\epsilon_{\text{EI}}^n(\mu)$ is the error due to the EI, as defined in (9).

Remark 4.4. *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,*

$$\|(A_\mu^{(n)})^{-T}\|_2 = \|(A_\mu^{(n)})^{-1}\|_2 = \sigma_{\max}\left((A_\mu^{(n)})^{-1}\right) = \frac{1}{\sigma_{\min}(A_\mu^{(n)})}, \tag{34}$$

the reciprocal of the smallest singular value of $A_\mu^{(n)}$. For many problems, the matrix $A_\mu^{(n)}$ is a constant matrix. Consequently, 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 finite element discretization, the induced matrix norm can be defined as

$$\|Z\|_H := \max_{\|x\|=1} \|Zx\|_H = \max_{\|x\|=1} \sqrt{x^T Z^T H Z x} = \|Z^T H Z\|_2. \tag{35}$$

In fact, the matrix $Z^T H Z$ is symmetric positive semidefinite because H is symmetric positive definite. This implies that the following equalities hold:

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

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 the matrix. Thus, $\|(A_\mu^{(n)})^{-T}\|_H$ can be obtained as,

$$\|(A_\mu^{(n)})^{-T}\|_H = \sigma_{\max}^2\left(L(A_\mu^{(n)})^{-T}\right) = \frac{1}{\sigma_{\min}^2((A_\mu^{(n)})^T L^{-1})}. \tag{36}$$

Remark 4.5. *The assumptions* (28a) *and* (28b) *in Corollary 4.2 require that the approximation error in the time trajectory are of the same magnitude. This can be achieved if the time step of the detailed simulation is well chosen. In fact, the 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. [13].*

Remark 4.6. *As for the constant ρ in the error bound* (31) *in Corollary 4.2, it is usually not easy to get an estimation of it directly from the property of the operator $b(\cdot)$ and the assumptions there. However, one may heuristically estimate it by observing the average ratio over all the time steps $(\frac{1}{K}\sum_{k=1}^{K}\|\tilde{r}_{\mathrm{pr}}^{k}(\mu_\star)\|)/(\frac{1}{K}\sum_{k=1}^{K}\|r_{\mathrm{pr}}^{k}(\mu_\star)\|)$, or the maximal ratio among all the time steps $\max_{k\in\{1,\dots,K\}}\|\tilde{r}_{\mathrm{pr}}^{k}(\mu_\star)\|/\|r_{\mathrm{pr}}^{k}(\mu_\star)\|$. Here, $\mu_\star$ is the parameter selected by the greedy algorithm. To compute the residual $\tilde{r}_{\mathrm{pr}}^{k}(\mu_\star)$, the detailed solutions $u^k(\mu_\star)$, $k=1,\dots,K$, at $\mu_\star$ are required, which causes no additional cost for snapshot based MOR methods because the detailed solutions at this parameter $\mu_\star$ are already available after the RB extension.*

Remark 4.7. *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 reduced basis MOR methods.

5 Construction of the projection matrix

In this section, we focus on the construction of the projection matrix using the reduced basis method [19]. Usually, Galerkin projection is employed to construct the ROM for the reduced basis 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 [11] is often used to construct the reduced basis. Algorithm 1 shows the basic step of the POD-Greedy algorithm.

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*

Algorithm 1 RB generation using POD-Greedy

Input: $\mathcal{P}_{\text{train}}$, μ_0, $\varepsilon_{\text{ROM}}(< 1)$.
Output: RB $V = [V_1, \ldots, V_N]$.

1: Initialization: $N = 0$, $\mu_\star = \mu_0$, $\eta_N(\mu_\star) = 1$, $V = [\,]$.
2: **while** $\psi_N(\mu_\star) > \varepsilon_{\text{ROM}}$ **do**
3: Compute the trajectory $S_{\max} := \{u^n(\mu_\star)\}_{n=0}^K$.
4: Enrich the RB, e.g. $V := [V, V_{N+1}]$, where V_{N+1} is the first POD mode of the matrix $\bar{U} := [\bar{u}^0, \ldots, \bar{u}^K]$ with $\bar{u}^n := u^n(\mu_\star) - \Pi_{\mathcal{W}^N}[u^n(\mu_\star)], n = 0, \ldots, K$. $\Pi_{\mathcal{W}^N}[u]$ is the projection of u onto the current space $\mathcal{W}^N := \text{span}\{V_1, \ldots, V_N\}$.
5: $N = N + 1$.
6: Find $\mu_\star := \arg \max_{\mu \in \mathcal{P}_{\text{train}}} \psi_N(\mu)$.
7: **end while**

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 [4, 24] 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

In this section, three models will be presented to show the performance of the proposed error estimation. The first 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. Algorithm 1 is used to generate the projection matrix V, and the ROMs are constructed by using the Galerkin projection. For particular applications in chemical engineering, the resulting reduced models are employed to solve the underlying optimization problems. All the computations were done on a PC with Intel(R) Core(TM)2 Quad CPU Q9550 2.83GHz RAM 4.00GB unless stated otherwise.

6.1 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 for MOR of general nonlinear evolution equations.

6.1.1 Reduced-order modeling of Burgers' equation

In this work, we consider the unsteady viscous Burgers' equation as follows,

$$u_t + (\frac{u^2}{2})_x = \nu u_{xx} + s(u,x), \quad x \in [0,1], \quad t \in (0,T], \tag{37}$$

where $\nu \in \mathcal{P}$ is the viscosity coefficient, and $s(u,x)$ is the source term.

In this model, the viscosity coefficient ν is considered as the parameter, i.e., $\mu := \nu$, and we chose $\mathcal{P} = [0.001, 1]$ as the parameter domain. 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 [17]. 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 full order model, in the general form of (2). The reduced basis is constructed by using Algorithm 1. Two output error bounds are used as the indicator for the greedy sampling process. The results are detailed in the next subsection.

An *a posteriori* error estimation for the reduced basis method applied to this equation is proposed in [17], 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 [17], this error estimation is no longer useful, when the viscosity ν is small and the final time T is large. In comparison with the early work in [17], we additionally use the EIM [2, 6] to treat the nonlinear flux for an efficient offline-online computation. Therefore, the derived error estimation in this paper is applicable to MOR of general nonlinear evolution equations.

6.1.2 Results

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

$$u(0,x) = 0, \quad x \in [0,1]; \quad u(t,0) = 0, \quad u_x(t,x)|_{x=1} = 0. \tag{38}$$

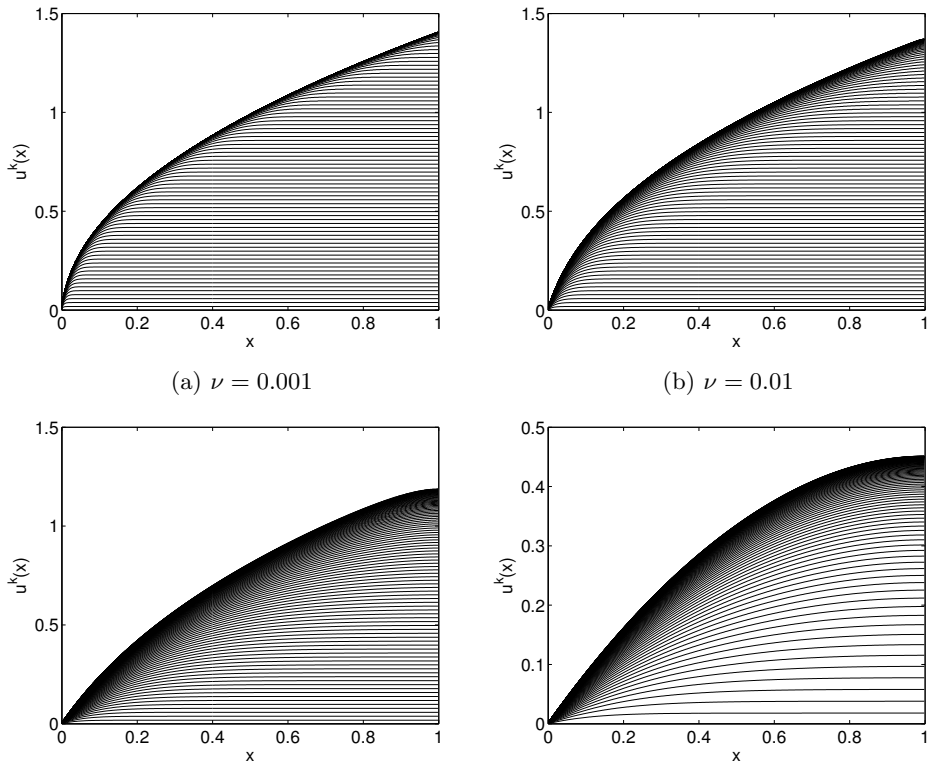
We use a uniform spatial grid with $\mathcal{N} = 500$ cells for the FOM, and $\Delta t = 2/K, K = 1000$ for both the FOM and ROM simulations.

Figure 1 shows the solutions to the FOM as a function of x and t^k. Each line represents the solution $u(x,t^k)$ at the time instance $t = t^k$, $k = 10j$, $j = 0, \ldots, K/10$. 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 built the RB and the CRB, respectively. We present the output error bound for the average error of the outputs during the whole evolution process, i.e., $\psi(\mu) = \frac{1}{K}\sum_{k=0}^{K} |Pu^k(\mu) - P\hat{u}^k(\mu)|$. Here, $P^T = [0, \ldots, 0, 1]^T \in \mathbb{R}^{\mathcal{N}}$, and $u^k(\mu), \hat{u}^k(\mu)$ are the solutions at the time instance t^k to the FOM and the ROM, respectively.

The behavior of the error bounds and the corresponding true error are illustrated in Figure 2. It is seen that the new error bound, which is derived from Theorem 4.1 and denoted as ErrorBound-2, works much better than ErrorBound-1, which is based on Proposition 3.1. Moreover, the new error bound is fairly sharp in comparison with the true error. Since this is just an academic numerical example, there is not much time

reduction in this model. For runtime comparison, we will report the computational times for the more challenging problems in the following subsections.

As mentioned in Remark 4.6, the constant ρ in (30) 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, i.e., $\rho \approx \frac{\frac{1}{K}\sum_{n=0}^{K-1} \|\tilde{r}_{\mathrm{pr}}^{n+1}(\mu_\star)\|}{\frac{1}{K}\sum_{n=0}^{K-1} \|r_{\mathrm{pr}}^{n+1}(\mu_\star)\|}$. Notice that ρ is changing with the dimension of the reduced basis 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 3. It is seen that the ratio decreases as the reduced basis is extended, which implies that the difference between $\tilde{r}_{\mathrm{pr}}^{n+1}$ and r_{pr}^{n+1} becomes small as the accuracy of the ROM is increased. 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 demonstrated in the next example, see Figure 5.



(a) $\nu = 0.001$ (b) $\nu = 0.01$

(c) $\nu = 0.1$ (d) $\nu = 1.0$

Figure 1: Solution of the Burgers' equation as a function of x and t^k with different viscosity coefficients ν. Each line represents the solution $u(x, t^k)$ at the time instance $t = t^k$, $k = 10j$, $j = 0, \ldots, K/10$.

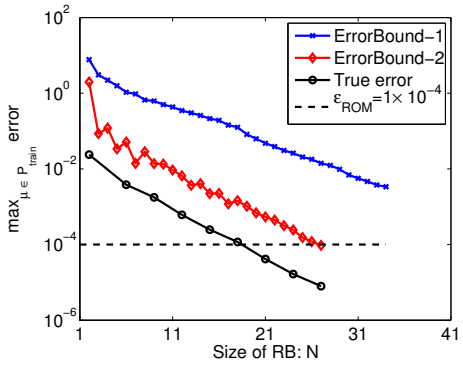


Figure 2: Illustration of the decay of the error bound and the corresponding true error during the reduced basis construction for Burgers' equation.

6.2 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 bound, named as ErrorBound-2. For comparison, the performance of the error bound in Proposition 3.1, named as ErrorBound-1, is also presented.

6.2.1 Model description and optimization

The governing equations of batch chromatography are as follows:

$$\begin{cases} \dfrac{\partial c_z}{\partial t} + \dfrac{1-\epsilon}{\epsilon}\dfrac{\partial q_z}{\partial t} = -\dfrac{\partial c_z}{\partial x} + \dfrac{1}{Pe}\dfrac{\partial^2 c_z}{\partial x^2}, & 0 < x < 1, \\ \dfrac{\partial q_z}{\partial t} = \dfrac{L}{Q/(\epsilon A_c)}\kappa_z(q_z^{\mathrm{Eq}} - q_z), & 0 \le x \le 1, \end{cases} \tag{39}$$

where c_z, q_z are the concentrations of the component z ($z = a, b$) in the liquid and solid phase, 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,

$$q_z^{\mathrm{Eq}} = f_z(c_a, c_b) := \frac{H_{z1}c_z}{1 + K_{a1}c_a^{\mathrm{f}}c_a + K_{b1}c_b^{\mathrm{f}}c_b} + \frac{H_{z2}c_z}{1 + K_{a2}c_a^{\mathrm{f}}c_a + K_{b2}c_b^{\mathrm{f}}c_b}, \tag{40}$$

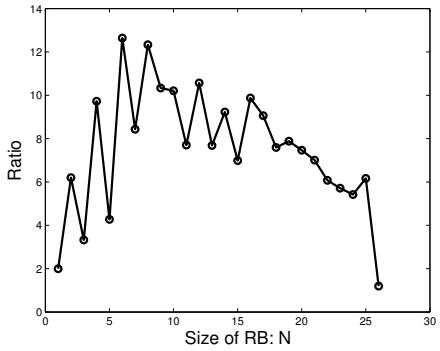


Figure 3: Illustration of the behavior of the ratio $\frac{\frac{1}{K}\sum_{n=0}^{K-1}\|\tilde{r}_{\mathrm{pr}}^{n+1}(\mu_\star)\|}{\frac{1}{K}\sum_{n=0}^{K-1}\|r_{\mathrm{pr}}^{n+1}(\mu_\star)\|}$, where $\mu_\star$ is the parameter selected by the greedy algorithm during the RB extension for MOR of the Burgers' equation.

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

$$\begin{cases} c_z(0,x) = 0, \quad q_z(0,x) = 0, \quad 0 \le x \le 1, \\ \dfrac{\partial c_z}{\partial x}|_{x=0} = Pe\left(c_z(t,0) - \chi_{[0,t_{\mathrm{in}}]}(t)\right), \\ \dfrac{\partial c_z}{\partial x}|_{x=1} = 0, \end{cases} \tag{41}$$

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., $\mu := (Q, t_{\mathrm{in}})$.

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.,

$$\begin{aligned} &\min_{\mu\in\mathcal{P}}\{-Pr(\mu)\}, \\ \text{s.t.}\quad &Rec_{\min} - Rec(\mu) \le 0, \quad \mu \in \mathcal{P}, \\ &c_z(\mu), q_z(\mu) \text{ are the solutions to the system (39)}, \ z = a, b, \end{aligned} \tag{42}$$

where $Pr(\mu) = \frac{Qp(\mu)}{t_4 - t_1}$, $Rec(\mu) = \frac{p(\mu)}{t_{\mathrm{in}}(c_a^{\mathrm{f}} + c_b^{\mathrm{f}})}$, $p(\mu) = \int_{t_3}^{t_4} c_{a,\mathrm{O}}(t,\mu)\,\mathrm{d}t + \int_{t_1}^{t_2} c_{b,\mathrm{O}}(t,\mu)\,\mathrm{d}t$, $c_{z,\mathrm{O}}(t,\mu) = c_z(t,1;\mu)$ is the concentration of component z at the outlet of the column, 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 [24].

6.2.2 Reduced-order modeling of the batch chromatographic model

We use the finite volume discretization to construct the FOM in the formulation of (10). The ROM is in the form of (13). The coefficient matrix $A_\mu^{(n)}$ in (10) is a constant matrix at all time instances t^n, $n = 0, \ldots, K-1$, and is independent of the parameter μ. This implies that the first equation in the dual system (12) 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.

6.2.3 Results

To compare the sharpness of two error bounds mentioned above, i.e., ErrorBound-1 and ErrorBound-2, we perform the POD-Greedy algorithm using the two error bounds, respectively. Let $\bar{\Psi}_{c_z}(\mu) := \frac{1}{K}\sum_{n=1}^{K} \Psi_{c_z}^n(\mu)$ be the average of the error bound for the output of c_z over the whole evolution process at a given parameter $\mu \in \mathcal{P}$, where $\Psi_{c_z}^n(\mu) = \tilde{\eta}_{N,M,c_z}^n(\mu)$ or $\Delta_{c_z}^n(\mu)$, indicating that they are computed by using Proposition 3.1 or Corollary 4.2, respectively. ErrorBound-1 and ErrorBound-2 are defined as $\eta_N(\mu_\star) := \max\limits_{\mu\in\mathcal{P}_{\text{train}}} \max\limits_{z\in\{a,b\}} \bar{\tilde{\eta}}_{N,M,c_z}(\mu)$ and $\Delta_N(\mu_\star) := \max\limits_{\mu\in\mathcal{P}_{\text{train}}} \max\limits_{z\in\{a,b\}} \bar{\Delta}_{c_z}(\mu)$, respectively. Here, a and b represent two components to be separated. In accordance, the reference true output error is defined as $e_N^{\max} := \max\limits_{\mu\in\mathcal{P}_{\text{train}}} \max\limits_{z\in\{a,b\}} \bar{e}_{N,c_z}(\mu)$, where $\bar{e}_{N,c_z}(\mu) := \frac{1}{K}\sum_{n=1}^{K} \|c_{z,\mathrm{O}}^n(\mu) - \hat{c}_{z,\mathrm{O}}^n(\mu)\|$, and $c_{z,\mathrm{O}}^n(\mu)$, $\hat{c}_{z,\mathrm{O}}^n(\mu)$ are the output response computed by using the FOM and ROM, respectively.

Figure 4 shows the error bound decay as the reduced basis is enriched. It is seen that the new output error bound (ErrorBound-2) works much better than the old one (ErrorBound-1). Since ErrorBound-1 decreases very slowly after certain extension steps, we use the early-stop criterion proposed in [24] to reasonably stop the iteration, although it does not go below the pre-specified tolerance. In contrast, the new error bound goes below the pre-specified tolerance as the number of the reduced basis increases to 45.

To show the efficiency of the new error bound, we compare the runtime for the generation of the reduced basis using the two error bounds. From Table 1, we see that using ErrorBound-2 takes slightly more time than using ErrorBound-1. This is because the residual of the additional dual system needs to be computed for ErrorBound-2. However, since ErrorBound-2 is much more accurate than ErrorBound-1, it deserves spending a bit more computational time for getting a more reliable ROM.

In addition, like we did for the Burgers' equation, we estimate the constant ρ in (30) by computing the average ratio over all the time steps at the chosen parameter $\mu_\star$ at each iteration step of the greedy algorithm, i.e., $\rho \approx \frac{\frac{1}{K}\sum_{n=0}^{K-1} \|\tilde{r}_{\text{pr}}^{n+1}(\mu_\star)\|}{\frac{1}{K}\sum_{n=0}^{K-1} \|r_{\text{pr}}^{n+1}(\mu_\star)\|}$. Figure 5

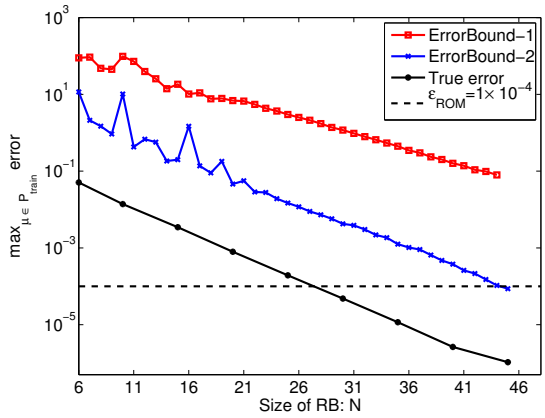


Figure 4: Illustration of the decay of the error bound and the corresponding true error during the reduced basis construction for batch chromatography.

Table 1: Comparison of runtime for the generation of the RB using two error bounds.

Model	Runtime [h] [1]
ROM using ErrorBound-1	6.8
ROM using ErrorBound-2	7.6

[1] 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 RAM 1TB.

shows the behavior of the ratio during the RB extension process. We have the same conclusion as above, i.e., the difference between $\tilde{r}_{\text{pr}}^{n+1}$ and r_{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. These numerical results confirm that our assumption (18) in Theorem 4.1 is reasonable.

Before addressing the ROM based optimization, we assess the validation of the ROM. To this end, we perform the detailed and reduced simulation over a test set with 500 random samples of the parameter in the feasible domain. Table 2 shows the results. It is seen that the average runtime is reduced by 98% using the ROM, and the maximal true output error is 8.16×10^{-7}, which is below the pre-specified tolerance. The optimization results are summarized in Table 3. The optimal solution of the ROM based optimization converges to that of the FOM based one, and the runtime is significantly reduced. The speedup factor (SpF) is 58.

6.3 Continuous SMB chromatographic model

Simulated moving bed (SMB) chromatography is a continuous multi-column process and has been widely used as an efficient separation technique in chemical industries.

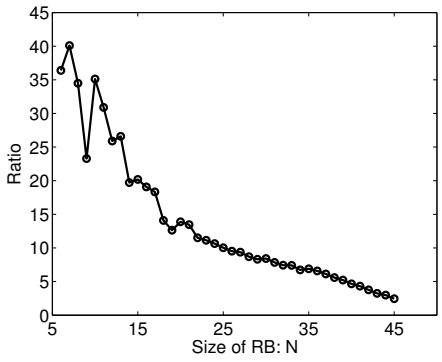


Figure 5: Illustration of the behavior of the ratio $\frac{\frac{1}{K}\sum_{n=0}^{K-1}\|\tilde{r}_{\mathrm{pr}}^{n+1}(\mu_\star)\|}{\frac{1}{K}\sum_{n=0}^{K-1}\|r_{\mathrm{pr}}^{n+1}(\mu_\star)\|}$, where $\mu_\star$ is the parameter selected by the greedy algorithm during the RB extension for MOR of batch chromatography.

Table 2: Comparison of runtime for the detailed and reduced simulation over a validation set $\mathcal{P}_{\mathrm{val}}$ 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

Recent studies on model order reduction of the SMB model can be found in [14] and the references therein. In [14] the reduced-order model 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 reduced basis method to build a ROM, which is qualified in 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 [6, 24], 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.

Table 3: Optimization based on the FOM $\mathcal{N} = 1500$ and the ROM $N = 45$.

Simulations	Obj. (Pr)	Opt. solution (μ)	#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

6.3.1 Model description and optimization

A classical SMB process with 4 zones is schematically shown in Figure 6. 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

$$\begin{cases} \dfrac{\partial c_z}{\partial t} + \dfrac{1-\epsilon}{\epsilon}\dfrac{\partial q_z}{\partial t} = -\dfrac{Qt_\mathrm{s}}{\epsilon A_c L}\left(\dfrac{\partial c_z}{\partial x} - \dfrac{1}{Pe}\dfrac{\partial^2 c_z}{\partial x^2}\right), & 0 < x < 1, \\ \dfrac{\partial q_z}{\partial t} = t_\mathrm{s}\kappa_z(q_z^{\mathrm{Eq}} - q_z), & 0 \le x \le 1, \end{cases} \tag{43}$$

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 coordinate, and t_s is the switching period. The adsorption equilibrium is $q_z^{\mathrm{Eq}} := H_z c_z$ with H_z being the Henry constant, which implies that the system of equations (43) is linear. It is assumed that $H_a > H_b$. The boundary conditions are

$$\begin{cases} \dfrac{\partial c_z}{\partial x}|_{x=0} = Pe(c_z(t,0) - c_z^{\mathrm{in}}), \\ \dfrac{\partial c_z}{\partial x}|_{x=1} = 0, \end{cases} \tag{44}$$

where c_z^{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 [14, 22]. The model parameters are summarized in Table 4.

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

As a case study, we use an SMB model with 4 zones and 8 columns, as is shown in Figure 6. In this model, the flow rate in each zone $Q_i, i = \mathrm{I}, \ldots, \mathrm{IV}$, and the switching period t_s are the operating parameters. Alternatively, four corresponding dimensionless quantities $m_i, i = \mathrm{I}, \ldots, \mathrm{IV}$, and the feed flow rate Q_f can also be chosen

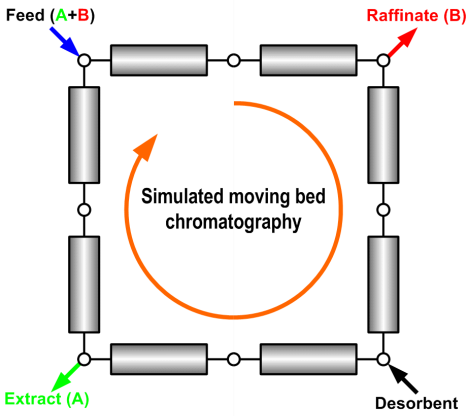


Figure 6: Schematic illustration of a simulated moving bed (SMB) chromatographic process with 4 zones and 8 columns.

as the operating parameters. The four dimensionless quantities introduced by the triangle theory [15] are defined as

$$m_i = \frac{Q_i t_\mathrm{s} - \epsilon \mathbf{V}}{\mathbf{V}(1-\epsilon)}, \quad i = \mathrm{I}, \ldots, \mathrm{IV}, \tag{45}$$

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

$$\max\left\{\|c_z(0,\cdot) - P_\mathrm{s}[c_z(1,\cdot)]\|, \|q_z(0,\cdot) - P_\mathrm{s}[q_z(1,\cdot)]\|\right\} < \varepsilon_\mathrm{CSS}, \tag{46}$$

where ε_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 column-wise switching operator, and T refers to the T-th period.

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

$$\begin{aligned} &\min_{\mu\in\mathcal{P}} f(\mu), \;\; f(\mu) = -Q_\mathrm{f}, \\ \text{s.t. } &Pu_{a,\min} - Pu_a(\mu) \le 0, \\ &Pu_{b,\min} - Pu_b(\mu) \le 0, \\ &Q_\mathrm{I} - Q_\max \le 0, \end{aligned} \tag{47}$$

where $Pu_a(\mu) := \frac{\int_0^1 c^\mathrm{E}_{a,\mathrm{CSS}}(t)\mathrm{d}t}{\int_0^1 c^\mathrm{E}_{a,\mathrm{CSS}}(t)\mathrm{d}t + \int_0^1 c^\mathrm{E}_{b,\mathrm{CSS}}(t)\mathrm{d}t}$, $Pu_b(\mu) := \frac{\int_0^1 c^\mathrm{R}_{b,\mathrm{CSS}}(t)\mathrm{d}t}{\int_0^1 c^\mathrm{R}_{a,\mathrm{CSS}}(t)\mathrm{d}t + \int_0^1 c^\mathrm{R}_{b,\mathrm{CSS}}(t)\mathrm{d}t}$ are the product purities at the extract and the raffinate outlets, and $c^\mathrm{E}_{z,\mathrm{CSS}}(t)$, $c^\mathrm{R}_{z,\mathrm{CSS}}(t)$

are the CSS concentrations of c_z at the extract and the raffinate outlets, respectively. 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 model order reduction to tackle this problem.

6.3.2 Reduced-order modeling of the SMB model

We use the FV discretization to construct the FOM as follows,

$$\begin{cases} A_{\mu,z} c_z^{n+1} = B_{\mu,z} c_z^n + r_z^n + t_s \kappa_z q_z^n, \\ q_z^{n+1} = (1 - t_s \kappa_z \Delta t) q_z^n + t_s \kappa_z H_z \Delta t c_z^n. \end{cases} \tag{48}$$

Note that the coefficient matrices $A_{\mu,z}$ and $B_{\mu,z}$ are time independent compared to the general form in (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 reduced basis 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

$$\begin{cases} \hat{A}_{\mu,z} a_{c_z}^{n+1} = \hat{B}_{\mu,z} a_{c_z}^n + \hat{r}_z + t_s \kappa_z \hat{D}_z a_{q_z}^n, \\ a_{q_z}^{n+1} = (1 - t_s \kappa_z \Delta t) a_{q_z}^n + t_s \kappa_z H_z \Delta t \hat{D}_z^T a_{c_z}^n, \end{cases} \tag{49}$$

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.

6.3.3 Results

The projection matrices V_{c_z} and V_{q_z} are constructed by using the POD-Greedy algorithm, Algorithm 1 in Section 5. Specifically, 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 ($\mathcal{O}(1000)$), which is larger than the dimension of the spatial discretization. To efficiently construct the reduced basis, the technique of adaptive snapshot selection [24] is employed. The column number of the projection matrices are $81, 82, 82, 82$, respectively, when the tolerance ε_{ROM} is taken 5.0×10^{-4}. Since the purities Pu_a and Pu_b in (47) are determined by the concentrations of c_a and c_b at the extract and the raffinate outlets in the CSS period, we defined the error bound as the maximal error bound for the output at the extract and the raffinate outlets, i.e., $\psi(\mu_\star) := \max\limits_{\mu \in \mathcal{P}_{\text{train}}} \max\limits_{z=\{a,b\}} \{\bar{\Delta}_{c_z}^{\text{E}}(\mu), \bar{\Delta}_{c_z}^{\text{R}}(\mu)\}$, where $\bar{\Delta}_{c_z}^{\text{O}}(\mu) := \frac{1}{K} \sum_{k=1}^{K} \Delta_{c_z}^{\text{O},n}(\mu), \text{O} = \text{E}, \text{R}$ indicating the average of the error bound over all time steps in the CSS period at the extract and the raffinate outlets, respectively. In accordance, the true output error is defined as $e(\mu_\star) := \max\limits_{\mu \in \mathcal{P}_{\text{train}}} \max\limits_{z=\{a,b\}} \{\bar{e}_{c_z}^{\text{E}}(\mu), \bar{e}_{c_z}^{\text{R}}(\mu)\}$, where $\bar{e}_{c_z}^{\text{O}}(\mu) := \frac{1}{K} \sum_{k=1}^{K} \|c_z^{\text{O},n}(\mu) - \hat{c}_z^{\text{O},n}(\mu)\|, \text{O} = \text{E}, \text{R}$. Here, $\hat{c}_z^{\text{E},n}(\mu), \hat{c}_z^{\text{R},n}(\mu)$ are the approximate concentrations at the extract and the raffinate outlets, respectively.

Figure 7 shows the error decay during the extension of the reduced basis. In this model, we simply estimate the constant $\rho = 3$. Note that the error bound is not rigorous in the first several steps. This is not surprising, because the RB approximation is not good enough when N is small and the constant ρ is taken aggressively. However, as the reduced basis is enriched, the new error bound is rigorous and fairly sharp in comparison with the true error. The error bound goes below the pre-specified tolerance when the maximal number of the reduced basis reaches 82.

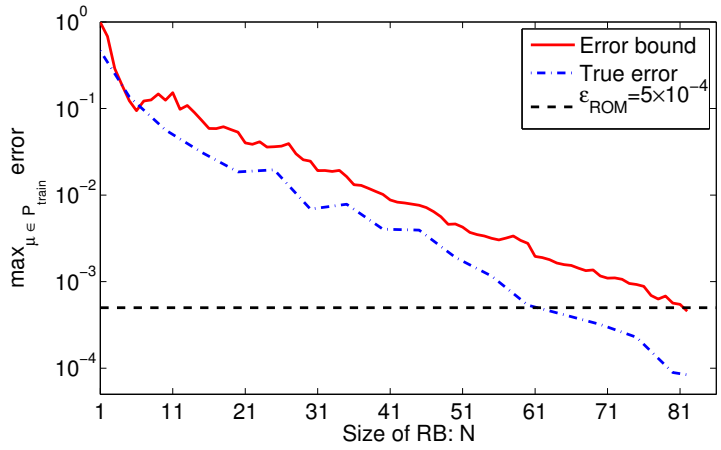


Figure 7: Error decay during the RB extension.

Before the ROM is used to solve the underlying optimization problem, we validate its accuracy by performing the full and reduced simulation 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 pre-specified tolerance. The average runtime is largely reduced and the speedup factor is 8.

Table 5: Comparison of runtime for the full and reduced simulations over a validation set $\mathcal{P}_{\text{val}}$ with 200 random sample points. $\varepsilon_{\text{ROM}} = 5.0 \times 10^{-4}$.

Model	Maximal error	Average runtime [s]/SpF
FOM ($\mathcal{N} = 800$)	–	349.5 / -
ROM	1.1×10^{-4}	45.2 / 8

Finally, we study the performance of the ROM based optimization. Table 6 shows the results using the constants $Pu_{a,\min} = 99\%, Pu_{b,\min} = 99\%, Q_{\max} = 0.50$ ml/s. It is seen that the ROM based optimization is very successful. The runtime of 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 9.

Table 6: Comparison of the results for the FOM based and the ROM based optimization.

		Initial guess	FOM-Opt.	ROM-Opt.
Objective	Q_{f} [ml/s]	0.07	0.0745	0.0744
Opt. solut.	m_1	4.50	4.3269	4.3218
	m_2	2.90	2.8599	2.8600
	m_3	3.50	3.6036	3.6025
	m_4	2.30	2.3468	2.3086
	Q_{f} [ml/s]	0.07	0.0745	0.0744
Constraints	Pu_a	98.9%	99.0%	99.0%
	Pu_b	99.5%	99.0%	99.0%
	Q_1 [ml/s]	0.4161	0.4997	0.4991
#Iterations			71	58
Runtime [h] / SpF			5.13 / -	0.59 / 9

7 Conclusions

We have presented an efficient *a posteriori* output error estimation for model order reduction 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 is shown in Section 6.3.

The proposed output error bound 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 bound.

References

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

[2] 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.

[3] 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., vol. 9 of Modeling, Simulation & Applications Series, Springer, 2014, pp. 159–185.

[4] 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., vol. 103 of Lecture Notes in Computational Science and Engineering, Springer, 2015, pp. 427–435.

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

[6] 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.

[7] L. Feng, A. C. Antoulas, and P. Benner, *Some a posteriori error bounds for model order reduction of parametrized linear systems*, in Preparation.

[8] L. Feng, P. Benner, and A. C. Antoulas, *Automatic generation of reduced order models for linear parametric systems*, submitted.

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

[10] M. A. Grepl, *Reduced-basis approximation a posteriori error estimation for parabolic partial differential equations*, PhD thesis, Massachusetts Institute of Technology, 2005.

[11] B. Haasdonk and M. Ohlberger, *Reduced basis method for finite volume approximations of parametrized linear evolution equations*, Mathematical Modelling and Numerical Analysis, 42 (2008), pp. 277–302.

[12] B. Haasdonk and M. Ohlberger, *Efficient reduced models and a posteriori error estimation for parametrized dynamical systems by offline/online decomposition*, Mathematical and Computer Modelling of Dynamical Systems, 17 (2011), pp. 145–161.

[13] R. H. Hoppe and Z. Liu, *Snapshot location by error equilibration in proper orthogonal decomposition for linear and semilinear parabolic partial differential equations*, Journal of Numerical Mathematics, 22 (2014), pp. 1–32.

[14] S. Li, L. Feng, P. Benner, and A. Seidel-Morgenstern, *Using surrogate models for efficient optimization of simulated moving bed chromatography*, Computers & Chemical Engineering, 67 (2014), pp. 121–132.

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

[16] K. W. Morton and D. F. Mayers, *Numerical solution of partial differential equations: an introduction*, Cambridge University Press, 2005.

[17] 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.

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

[19] A. T. Patera and G. Rozza, *Reduced Basis Approximation and a Posteriori Error Estimation for Parametrized Partial Differential Equations*, MIT, 2007. MIT Pappalardo Graduate Monographs in Mechanical Engineering, Available from http://augustine.mit.edu/methodology/methodology_book.htm.

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

[21] 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*, Journal of Fluids Engineering, 124 (2002), pp. 70–80.

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

[23] D. V. Rovas, *Reduced-basis output bound methods for parametrized partial differential equations*, PhD thesis, Massachusetts Institute of Technology, 2003.

[24] Y. Zhang, L. Feng, S. Li, and P. Benner, *Accelerating PDE constrained optimization by the reduced basis method: application to batch chromatography*, MPI Magdeburg Preprint MPIMD/14-09, May 2014. Available from http://www.mpi-magdeburg.mpg.de/preprints/.