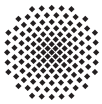

Universität Stuttgart

SimTech
Cluster of Excellence

B. Haasdonk[a] · **M. Ohlberger**[b]

Efficient Reduced Models and A-Posteriori Error Estimation for Parametrized Dynamical Systems by Offline/Online Decomposition

Stuttgart, November 2009

[a] Institute of Applied Analysis and Numerical Simulation, University of Stuttgart
Pfaffenwaldring 57, 70569 Stuttgart/ Germany
haasdonk@mathematik.uni-stuttgart.de
http://www.ians.uni-stuttgart.de/agh

[b] Institute of Numerical and Applied Mathematics University of Münster
Einsteinstr. 62, 48149 Münster/ Germany
mario.ohlberger@math.uni-muenster.de
http://wwwmath.uni-muenster.de/u/ohlberger

Abstract Reduced basis (RB) methods are an effective approach for model reduction of parametrized partial differential equations. In the field of dynamical systems' order reduction, these methods are not very established, but the interest in reduction of parametrized systems is increasing. In the current presentation, we show that some characteristic components of RB-methods can be transfered to model reduction of parametrized linear dynamical systems. We assume an affine parameter dependence of the system components, which allows an offline/online decomposition and is the basis for efficient reduced simulation. Additionally, error control is possible by a-posteriori error estimators for the state and output, based on residual analysis and primal-dual techniques. Experiments demonstrate the applicability of the reduced parametrized systems, the reliability of the error estimators and the runtime gain by the reduction technique. The a-posteriori error estimation technique can straightforwardly be applied to all traditional projection-based reduction techniques of non-parametric linear systems, such as modal reduction, balanced truncation, moment matching, POD, etc.

Keywords Parametrized Dynamical Systems · Model Order Reduction · Reduced Basis Methods · A-Posteriori Error Estimation · Offline/Online Decomposition

Preprint Series **Issue No. 2009-23**
Stuttgart Research Centre for Simulation Technology (SRC SimTech)
SimTech – Cluster of Excellence
Pfaffenwaldring 7a
70569 Stuttgart
publications@simtech.uni-stuttgart.de
www.simtech.uni-stuttgart.de

1 Introduction

Reduced basis (RB) methods are an effective approach for model reduction of parametrized partial differential equations. These are partial differential equations, where parameters characterize the system, e.g. geometry, material, boundary-value, initial-value or control parameters. The need for parametrized reduced models can result from multi-query scenarios, where many simulations have to be performed for varying parameters such as parameter studies, parameter optimization, inverse problems, etc. Similarly, real-time requirements can be a motivation for parametrized model reduction, e.g. control, interactive simulation environments, etc. In addition to parametrized reduced models, a fast rigorous parameter-dependent quantification of the model error is required. These demands are fulfilled by RB-methods. During the last years various types of stationary and time-dependent, linear and nonlinear parametrized partial differential equations have been treated with this technique [19, 10,11,13,23,20]. An overview with many further recent references is given by [18,21]. In the field of model order reduction of dynamical systems, these methods are not very established, but the interest in reduction of parametrized systems is increasing. Early work in [2] already considers the solution of parametrized systems by concatenating projection bases of special parameter choices. Parametrized systems are further tackled with interpolation [26] or moment matching techniques [7,22,9]. Polynomially parametrized systems are treated in [8], nonlinear systems are considered in [5]. Some recent approaches comprise interpolatory schemes with sparse grids [4] and superposition of locally reduced models [15]. A-posteriori error analysis for reduced non-parametric dynamical systems is considered in [14] and error analysis of POD reduction schemes for optimal control can be found in [24].

In the current presentation, we show that some characteristic components of RB-methods can be transfered to model reduction of parametrized dynamical systems. We exemplify this for linear systems with output estimation. A so called offline/online decomposition is the key for efficient simulation. In the offline phase the reduced basis and auxiliary parameter-independent quantities are precomputed. In the online phase, these preparations allow rapid assembly of the reduced parameter dependent system and online simulations for varying parameters with a complexity independent of the original state dimension. The possibly extensive offline phase pays off in case of a multi-query context, where a sufficient number of reduced simulations with different parameter constellations are expected. In addition to the effective reduced simulation schemes, error control is possible by a-posteriori error estimators. These are based on residual analysis and can also be effectively decomposed in an offline/online fashion and hence allow fast and rigorous error guarantees. In contrast to global-in-time a-priori estimates in classical MOR of dynamical systems, these a-posteriori error estimates provide error bounds for the state variable or output variable point-wise in time. By considering an additional dual problem, output estimates and error-estimators can be improved.

In the following section, we introduce the reduced basis simulation scheme for parametrized problems. Sec. 3 is devoted to the algorithmical decomposition into an offline and online phase. For the current formulation a-posteriori error estimation is demonstrated in Sec. 4 including a full offline/online decomposition. Then, we illustrate, how a parametrized dual problem can be applied to provide improved output estimation. Experiments in Sec. 6 demonstrate the applicability of the model reduction technique, the quality of the a-posteriori error estimators and the runtime-gain of the reduced scheme. We conclude in Sec. 7.

2 Parametrized Reduced Simulation Scheme

We assume the following parametrized linear dynamical system for a state variable $x(t) \in \mathbb{R}^n$, input $u(t) \in \mathbb{R}^m$ and output variable $y(t) \in \mathbb{R}^p$ for $t \in [0, \infty)$

$$\begin{aligned} \frac{d}{dt}x(t) &= \mathbf{A}(t,\boldsymbol{\mu})x(t) + \mathbf{B}(t,\boldsymbol{\mu})u(t) \\ y(t) &= \mathbf{C}(t,\boldsymbol{\mu})x(t) + \mathbf{D}(t,\boldsymbol{\mu})u(t). \end{aligned}$$

The system matrices $\mathbf{A}(t,\boldsymbol{\mu}) \in \mathbb{R}^{n\times n}, \mathbf{B}(t,\boldsymbol{\mu}) \in \mathbb{R}^{n\times m}, \mathbf{C}(t,\boldsymbol{\mu}) \in \mathbb{R}^{p\times n}, \mathbf{D}(t,\boldsymbol{\mu}) \in \mathbb{R}^{p\times m}$ depend on a parameter $\boldsymbol{\mu} \in \mathcal{P} \subset \mathbb{R}^d$ from a bounded parameter domain $\mathcal{P}$ and may be time-varying. The parameter $\boldsymbol{\mu}$ is assumed to be fixed during a single simulation of the dynamical system. Occasionally, the solution and output will be denoted as $x(t,\boldsymbol{\mu}), y(t,\boldsymbol{\mu})$ to emphasize their parameter dependence.

Given a projection matrix $\mathbf{V} \in \mathbb{R}^{n\times k}$ with reduced model order $k << n$ and biorthogonal $\mathbf{W} \in \mathbb{R}^{n\times k}$, i.e. $\mathbf{W}^T\mathbf{V} = \mathbf{I}_{k\times k}$, the reduced system reads as usual [1]

$$\begin{aligned} \frac{d}{dt}\hat{x}(t) &= \hat{\mathbf{A}}(t,\boldsymbol{\mu})\hat{x}(t) + \hat{\mathbf{B}}(t,\boldsymbol{\mu})u(t) \\ \hat{y}(t) &= \hat{\mathbf{C}}(t,\boldsymbol{\mu})\hat{x}(t) + \hat{\mathbf{D}}(t,\boldsymbol{\mu})u(t) \end{aligned} \tag{1}$$

with reduced system matrices

$$\begin{aligned} \hat{\mathbf{A}}(t,\boldsymbol{\mu}) &:= \mathbf{W}^T\mathbf{A}(t,\boldsymbol{\mu})\mathbf{V}, \quad & \hat{\mathbf{B}}(t,\boldsymbol{\mu}) &:= \mathbf{W}^T\mathbf{B}(t,\boldsymbol{\mu}) \\ \hat{\mathbf{C}}(t,\boldsymbol{\mu}) &:= \mathbf{C}(t,\boldsymbol{\mu})\mathbf{V}, \quad & \hat{\mathbf{D}}(t,\boldsymbol{\mu}) &:= \mathbf{D}(t,\boldsymbol{\mu}) \end{aligned}$$

and initial condition

$$\hat{x}(0) = \hat{x}_0 := \mathbf{W}^T x(0).$$

In reduced basis methods, the projection basis $\mathbf{V}$ is constructed in a simulation-based way such that $\mathrm{colspan}\mathbf{V} \subset \mathrm{span}\{x(t_i,\boldsymbol{\mu}_i)\}_{i\in I}$. Here $t_i, \boldsymbol{\mu}_i$ are suitably selected time instants and parameters and $x(t_i,\boldsymbol{\mu}_i)$ are so called *snapshots* of the solution. The basis matrix $\mathbf{V}$ is commonly orthonormalized with respect to a certain problem specific inner product, such that the reduced system is numerically more stable. For more details on basis generation in RB-methods, we refer to [18,12,13,11]. In the following, however, we do not put any assumption on the reduced basis apart from biorthogonality of $\mathbf{V}$ and $\mathbf{W}$. Hence in particular, the method and results are as well valid for Krylov-subspace bases, bases obtained from modal analysis, balanced truncation, POD, etc. If we omit the parameter dependence, our error estimators are directly applicable to all existing standard linear projection techniques.

3 Offline-Online Decomposition

For efficient computation, we pose some assumptions on the matrices and initial data. We require that the system matrices can be decomposed as a weighted sum of parameter-independent parts with parameter-dependent weights. Note that this basic idea of linear superposition of systems has also been used in [2]. Here we perform a refined argumentation, which results in an online simulation scheme, the complexity of which is completely independent of the dimension n. To be precise, we assume the separable parameter dependence for the system matrices

$$\mathbf{A}(t,\boldsymbol{\mu}) = \sum_{q=1}^{Q_A} \sigma_A^q(t,\boldsymbol{\mu})\mathbf{A}^q, \quad \mathbf{B}(t,\boldsymbol{\mu}) = \sum_{q=1}^{Q_B} \sigma_B^q(t,\boldsymbol{\mu})\mathbf{B}^q, \quad \mathbf{C}(t,\boldsymbol{\mu}) = \sum_{q=1}^{Q_C} \sigma_C^q(t,\boldsymbol{\mu})\mathbf{C}^q \tag{2}$$

with scalar parameter- and time-dependent coefficient functions $\sigma_A^q, \sigma_B^q, \sigma_C^q$ and parameter- and time-independent matrices $\mathbf{A}^q, \mathbf{B}^q, \mathbf{C}^q$ of suitable dimensions and small number of components Q_A, Q_B, Q_C. Note that the superscript q does not denote a power, but simply is a counting index. We assume, that the initial data variations of the system are not arbitrary, but can similarly be described by parameter variations, i.e. $x(0) = x_0(\boldsymbol{\mu})$ with

$$x_0(\boldsymbol{\mu}) = \sum_{q=1}^{Q_{x_0}} \sigma_{x_0}^q(\boldsymbol{\mu}) x_0^q. \tag{3}$$

Making use of the assumed parameter dependence, the reduced simulation can be performed rapidly in a complexity, which is completely independent of n. This is obtained by an offline/online decomposition.

In the *offline phase*, the parameter-independent quantities of the reduction scheme are computed. This phase may be arbitrary time consuming, as it will pay off in view of sufficiently many online simulations. First, the biorthogonal projection matrices $\mathbf{V}$ and $\mathbf{W}$ may be generated by any algorithm, then the following parameter independent components are computed:

$$\hat{\mathbf{A}}^q := \mathbf{W}^T\mathbf{A}^q\mathbf{V}, \qquad \hat{\mathbf{B}}^q := \mathbf{W}^T\mathbf{B}^q, \qquad \hat{\mathbf{C}}^q := \mathbf{A}^q\mathbf{V}, \qquad \hat{x}_0^q := \mathbf{W}^T x_0^q. \tag{4}$$

In the *online phase*, the parameter $\boldsymbol{\mu}$ is known and the reduced simulation matrices can be assembled in complexity *independent of* n. In particular we obtain from (2)–(3) and (4):

$$\hat{\mathbf{A}}(t,\boldsymbol{\mu}) = \sum_{q=1}^{Q_A} \sigma_A^q(t,\boldsymbol{\mu})\hat{\mathbf{A}}^q, \qquad \hat{\mathbf{B}}(t,\boldsymbol{\mu}) = \sum_{q=1}^{Q_B} \sigma_B^q(t,\boldsymbol{\mu})\hat{\mathbf{B}}^q,$$

$$\hat{\mathbf{C}}(t,\boldsymbol{\mu}) = \sum_{q=1}^{Q_C} \sigma_C^q(t,\boldsymbol{\mu})\hat{\mathbf{C}}^q, \qquad \hat{\mathbf{D}}(t,\boldsymbol{\mu}) = \mathbf{D}(t,\boldsymbol{\mu}), \qquad \hat{x}_0(\boldsymbol{\mu}) = \sum_{q=1}^{Q_{x_0}} \sigma_{x_0}^q(\boldsymbol{\mu})\hat{x}_0^q.$$

The separable parameter dependence of the components is not a strong assumption, as there are several methods to obtain such exact or approximate decompositions. First, if the dynamical system results from a discretization of a physical problem, the physical parameters can frequently be tracked through the discretization and hereby explicitly give such a desired decomposition, as e.g. done for finite volume discretizations [13]. This clearly requires full control over the discretization, which is realizable (and a good argument) for own development of discretization code instead of using black-box discretization packages. If there is some algebraic model knowledge about the parameter dependence, e.g. the number of components and the coefficient functions are known, but the component matrices not, these matrices can be constructed by setting up matrix equations from sample matrices and solving for the matrix components [17]. If the matrices are given as explicit functions $\mathbf{A}(\mu,t)$ or can be obtained from a black-box discretization software-package, approximation methods can be used to produce finite-sum representations, e.g. polynomial, modal or empirical interpolation [3].

4 A-Posteriori Error Estimation

A further attractive aspect of RB-methods is rigorous error analysis. In particular a-posteriori error estimates can be obtained. These allow to assess the quality of a reduced model during the reduced simulation. In particular these error-estimators provide a theoretical foundation for empirical basis generation procedures, such as POD-Greedy [13] or other snapshot-based approaches. As the error-estimators are not restricted to a special choice of reduced basis, they are as well applicable to non-parametric problems and traditional reduction schemes, such as modal reduction, balanced truncation, moment matching, etc.

The error analysis is residual-based, hence we define the error and residual

$$e(t,\boldsymbol{\mu}) := x(t,\boldsymbol{\mu}) - \mathbf{V}\hat{x}(t,\boldsymbol{\mu}), \qquad R(t,\boldsymbol{\mu}) := \mathbf{A}(t,\boldsymbol{\mu})\mathbf{V}\hat{x}(t) + \mathbf{B}(t,\boldsymbol{\mu})u(t) - \mathbf{V}\frac{d}{dt}\hat{x}(t). \tag{5}$$

This residual has the notable property, that it is zero, if the exact solution $x(t)$ evolves in the column-span of $\mathbf{V}$, i.e. the reduced system reproduces the exact system's solution without approximation error. Further, it satisfies a so called *Galerkin orthogonality* $\mathbf{W}^T R(t,\boldsymbol{\mu}) = 0$ due to (1) and the biorthogonality of $\mathbf{W}$ and $\mathbf{V}$. The error in particular satisfies

$$e(0,\boldsymbol{\mu}) = x_0(\boldsymbol{\mu}) - \mathbf{V}\hat{x}_0(\boldsymbol{\mu}) = (\mathbf{I}_{n\times n} - \mathbf{V}\mathbf{W}^T)x_0(\boldsymbol{\mu}). \tag{6}$$

For deriving a-posteriori error estimators, suitable norms must be chosen. We assume, that some symmetric positive definite inner matrix $\mathbf{G} \in \mathbb{R}^{n\times n}$ is given and denote $\langle x, x'\rangle_{\mathbf{G}} := x^T\mathbf{G}x'$ as the corresponding inner product. This induces a vector norm $\|x\|_{\mathbf{G}} := \sqrt{\langle x,x\rangle_{\mathbf{G}}}$ on $\mathbb{R}^n$ and a matrix-norm $\|\mathbf{A}\|_{\mathbf{G}} := \sup_{\|x\|_{\mathbf{G}}=1} \|\mathbf{A}x\|_{\mathbf{G}}$ for $\mathbf{A} \in \mathbb{R}^{n\times n}$. For the output, we will consider the simple 2-norm, hence define the induced matrix norm $\|\mathbf{C}\|_{\mathbf{G}} := \sup_{\|x\|_{\mathbf{G}}=1} \|\mathbf{C}x\|$ for $\mathbf{C} \in \mathbb{R}^{p\times n}$. The matrix $\mathbf{G}$ could be chosen trivially $\mathbf{G} = \mathbf{I}_{n\times n}$ in the following, which then reproduces the usual 2-norm $\|\cdot\|$ for vectors and matrices. However, other choices of $\mathbf{G}$ are possible in a problem-dependent way, which may improve the error estimator. For demonstrating the error estimation technique, we propose the following a-posteriori error estimators for the state variable and output:

Proposition 1 (A-Posteriori Error Estimate) *We assume that* $\mathbf{A}(t,\boldsymbol{\mu}) = \mathbf{A}(\boldsymbol{\mu})$ *is time-invariant and has eigenvalues with nonpositive real part for all* $\boldsymbol{\mu}$*. Hence, the solutions are bounded and we assume to have a computable constant* $C_1(\boldsymbol{\mu})$ *with*

$$C_1(\boldsymbol{\mu}) \geq \sup_t \left\|\exp(\mathbf{A}(\boldsymbol{\mu})t)\right\|_{\mathbf{G}}.$$

Then for the state variable the following error estimate holds:

$$\left\|x(t,\boldsymbol{\mu}) - \mathbf{V}\hat{x}(t,\boldsymbol{\mu})\right\|_{\mathbf{G}} \leq \Delta_x(t,\boldsymbol{\mu}) := C_1(\boldsymbol{\mu})\left\|e(0,\boldsymbol{\mu})\right\|_{\mathbf{G}} + C_1(\boldsymbol{\mu})\int_0^t \left\|R(\tau,\boldsymbol{\mu})\right\|_{\mathbf{G}} d\tau. \tag{7}$$

If we additionally assume to have an upper bound $C_2(\boldsymbol{\mu}) \geq \sup_t \left\|\mathbf{C}(t,\boldsymbol{\mu})\right\|_{\mathbf{G}}$*, then the following output error estimate holds:*

$$\left\|y(t,\boldsymbol{\mu}) - \hat{y}(t,\boldsymbol{\mu})\right\| \leq \Delta_y(t,\boldsymbol{\mu}) := C_2(\boldsymbol{\mu})\Delta_x(t,\boldsymbol{\mu}). \tag{8}$$

Proof From the definition of the residual (5) we see that

$$\mathbf{V}\frac{d}{dt}\hat{x}(t) = \mathbf{A}(\boldsymbol{\mu})\mathbf{V}\hat{x}(t) + \mathbf{B}(t,\boldsymbol{\mu})u(t) - R(t,\boldsymbol{\mu}).$$

Subtracting this from the original system yields the error evolution equation

$$\frac{d}{dt}e(t) = \mathbf{A}(\boldsymbol{\mu})e(t) + R(t,\boldsymbol{\mu})$$

with initial condition (6). This linear system has the explicit solution

$$e(t,\boldsymbol{\mu}) = \exp(\mathbf{A}(\boldsymbol{\mu})t)e(0) + \int_0^t \exp(\mathbf{A}(\boldsymbol{\mu})(t-\tau))R(\tau)d\tau.$$

Due to the assumed boundedness of $\left\|\exp(\mathbf{A}(\boldsymbol{\mu})s)\right\|_{\mathbf{G}} \leq C_1(\boldsymbol{\mu})$ for $s \in \mathbb{R}^+$ we obtain the claimed bound Eqn. (7). For the second statement, we observe that

$$\begin{aligned} y(t,\boldsymbol{\mu}) - \hat{y}(t,\boldsymbol{\mu}) &= \mathbf{C}(t,\boldsymbol{\mu})x(t,\boldsymbol{\mu}) + \mathbf{D}(t,\boldsymbol{\mu})u(t) - \mathbf{C}(t,\boldsymbol{\mu})\mathbf{V}\hat{x}(t,\boldsymbol{\mu}) - \mathbf{D}(t,\boldsymbol{\mu})u(t) \\ &= \mathbf{C}(t,\boldsymbol{\mu})(x(t,\boldsymbol{\mu}) - \mathbf{V}\hat{x}(t,\boldsymbol{\mu})). \end{aligned}$$

which yields Eqn. (8) and concludes the proof.

Note, that similar statements for non-stable systems are possible, if only finite times $t \in [0,T]$ are considered. Similarly, systems with time-varying matrix $\mathbf{A}(t,\boldsymbol{\mu})$ can be treated by suitable modifications of the constant C_1 and/or additional terms in the error bound. We remark, that the matrix $\mathbf{G}$ in the above formulation is a degree of freedom, which can be used to keep the constants C_1 and C_2 small. Let $\mathbf{A} = \mathbf{U}\mathbf{J}\mathbf{U}^{-1}$ be an eigendecomposition of $\mathbf{A}$ with unitary $\mathbf{U} \in \mathbb{C}^{n\times n}$ and Jordan-block matrix $\mathbf{J}$. Hence (after extending the $\left\|\cdot\right\|_{\mathbf{G}}$ on complex matrices) we have

$$\left\|\exp(\mathbf{A}t)\right\|_{\mathbf{G}} = \left\|\mathbf{U}\exp(\mathbf{J}t)\mathbf{U}^{-1}\right\|_{\mathbf{G}} \leq \left\|\mathbf{U}\right\|_{\mathbf{G}}\left\|\exp(\mathbf{J}t)\right\|_{\mathbf{G}}\left\|\mathbf{U}^{-1}\right\|_{\mathbf{G}}.$$

If, for example, $\mathbf{A}$ is symmetric, $\mathbf{J}$ is a real diagonal matrix with negative entries. By choosing $\mathbf{G}$ as (a multiple of) the identity matrix, the product remains upper bounded by 1 and $C_1 = 1$ is a proper choice. In dynamical systems obtained from PDE discretizations, the matrix $\mathbf{G}$ is usually chosen as the Gram-Matrix of the finite element / finite volume basis. Then, $\left\|e\right\|_{\mathbf{G}}$ is the function space norm of the error in the full finite element / finite volume function space.

The above simple result is practically relevant, as a full offline/online decomposition of the error bound is possible which enables fast and rigorous error estimation during the reduced simulation. This

is based on the fact, that residual norms can be determined exactly based on (5). We omit time and parameter dependency in the following notation:

$$\|R\|^2_{\mathbf{G}} = R^T\mathbf{G}R = \hat{x}^T\mathbf{V}^T\mathbf{A}^T\mathbf{G}\mathbf{A}\mathbf{V}\hat{x} + u^T\mathbf{B}^T\mathbf{G}\mathbf{B}u + (\frac{d}{dt}\hat{x})^T\mathbf{V}^T\mathbf{G}\mathbf{V}(\frac{d}{dt}\hat{x}) \tag{9}$$

$$+2u^T\mathbf{B}^T\mathbf{G}\mathbf{A}\mathbf{V}\hat{x} - 2(\frac{d}{dt}\hat{x})^T\mathbf{V}^T\mathbf{G}\mathbf{A}\mathbf{V}\hat{x} - 2(\frac{d}{dt}\hat{x})^T\mathbf{V}^T\mathbf{G}\mathbf{B}u \tag{10}$$

$$= \hat{x}^T\mathbf{M}_1\hat{x} + u^T\mathbf{M}_2u + (\frac{d}{dt}\hat{x})^T\mathbf{M}_3(\frac{d}{dt}\hat{x}) + \tag{11}$$

$$2u^T\mathbf{M}_4\hat{x} - 2(\frac{d}{dt}\hat{x})^T\mathbf{M}_5\hat{x} - 2(\frac{d}{dt}\hat{x})^T\mathbf{M}_6u$$

where the matrices $\mathbf{M}_1$–$\mathbf{M}_6$ are introduced as abbreviations for the matrices in (9)–(10).

$$\mathbf{M}_1(t,\boldsymbol{\mu}) = \mathbf{V}^T\mathbf{A}(t,\boldsymbol{\mu})^T\mathbf{G}\mathbf{A}(t,\boldsymbol{\mu})\mathbf{V}, \quad \mathbf{M}_2(t,\boldsymbol{\mu}) = \mathbf{B}(t,\boldsymbol{\mu})^T\mathbf{G}\mathbf{B}(t,\boldsymbol{\mu}), \tag{12}$$

$$\mathbf{M}_3 := \mathbf{V}^T\mathbf{G}\mathbf{V}, \; \mathbf{M}_4(t,\boldsymbol{\mu}) = \mathbf{B}(t,\boldsymbol{\mu})^T\mathbf{G}\mathbf{A}(t,\boldsymbol{\mu})\mathbf{V},$$

$$\mathbf{M}_5(t,\boldsymbol{\mu}) = \mathbf{V}^T\mathbf{G}\mathbf{A}(t,\boldsymbol{\mu})\mathbf{V}, \qquad \mathbf{M}_6(t,\boldsymbol{\mu}) = \mathbf{V}^T\mathbf{G}\mathbf{B}(t,\boldsymbol{\mu}). \tag{13}$$

The offline/online decomposition is then obvious: In the offline phase, we can compute time- and parameter-independent component matrices, i.e.

$$\mathbf{M}_1^{q,q'} := \mathbf{V}^T(\mathbf{A}^q)^T\mathbf{G}\mathbf{A}^{q'}\mathbf{V}$$

for $q, q' = 1, \ldots, Q_A$ and similarly for $\mathbf{M}_2^{q,q'}$, $\mathbf{M}_3$ (not parameter dependent), $\mathbf{M}_4^{q,q'}$, $\mathbf{M}_5^q$ and $\mathbf{M}_6^q$. In the online phase, these matrices can be combined and we assemble the matrices of (12)–(13), i.e.

$$\mathbf{M}_1(t,\boldsymbol{\mu}) := \sum_{q,q'}^{Q_A} \sigma_A^q(t,\boldsymbol{\mu})\sigma_A^{q'}(t,\boldsymbol{\mu})\mathbf{M}_1^{q,q'}$$

and similarly $\mathbf{M}_2, \mathbf{M}_4, \mathbf{M}_5, \mathbf{M}_6$. Note again, that $\mathbf{M}_3$ is already available from the offline phase. The quantities $\hat{x}(t)$, $\frac{d}{dt}\hat{x}(t)$ and $u(t)$ are available during the reduced simulation, hence the squared residual norm can be computed online.

The initial error in the above estimate (7) can easily be set to 0 by ensuring, that the components of the initial data are in the reduced space, i.e. $x_0^q \in \mathrm{colspan}\mathbf{V}$. For more general basis $\mathbf{V}$, the norm of the initial error is required for the error estimator in (7)

$$\|e(0,\boldsymbol{\mu})\|^2_{\mathbf{G}} = x_0(\boldsymbol{\mu})^T(\mathbf{I}_{n\times n} - \mathbf{V}\mathbf{W}^T)^T\mathbf{G}(\mathbf{I}_{n\times n} - \mathbf{V}\mathbf{W}^T)x_0(\boldsymbol{\mu}). \tag{14}$$

This can similarly be decomposed in an offline/online fashion. In the offline phase we compute the parameter independent components

$$m^{q,q'} := (x_0^q)^T(\mathbf{I}_{n\times n} - \mathbf{V}\mathbf{W}^T)^T\mathbf{G}(\mathbf{I}_{n\times n} - \mathbf{V}\mathbf{W}^T)(x_0^{q'})$$

for $q, q' = 1, \ldots, Q_{x_0}$. In the online phase the error norm is assembled by

$$\|e(0,\boldsymbol{\mu})\|^2_{\mathbf{G}} = \sum_{q,q'=1}^{Q_{x_0}} \sigma_{x_0}^q(\boldsymbol{\mu})\sigma_{x_0}^{q'}(\boldsymbol{\mu})m^{q,q'}.$$

Note, that this error estimator was chosen for simple presentation of the approach. For computation of the above error bound, the exact integral cannot be determined but must be approximated by some quadrature rule. Further, the presented RB-method still is continuous in time and results in a real simulation scheme after suitable time discretization by ordinary differential equation solvers. These steps introduce additional (but controllable) numerical errors in the above analysis. To prevent these additional approximation issues, a-posteriori error bounds can be derived, which are suited to specific time integration schemes [13,11].

5 Improved Output Estimation by Dual Problem

In view of (8) we may face a problem in practice: The constant $C_2(\boldsymbol{\mu})$ may be very large such that the output bound $\Delta_y(t,\boldsymbol{\mu}) = C_2(\boldsymbol{\mu})\Delta_x(t,\boldsymbol{\mu})$ is not very useful. This may for instance happen, if the output function is a point-evaluation of the field variable. In such cases, better output treatment is possible by considering a suitable dual problem [16,11]. In particular, the dual problem can be used to i) add a correction term to the reduced output and ii) improve the a-posteriori error-bound. Not surprisingly, these improved estimates are obtained for the price of increased computational cost. In the following we detail this procedure.

For simplicity of presentation, we restrict here to a single scalar output $y(t) \in \mathbb{R}^p, p = 1$, hence $\mathbf{C}(t,\boldsymbol{\mu}) = \mathbf{c}^T(t,\boldsymbol{\mu})$ for $\mathbf{c}(t,\boldsymbol{\mu}) \in \mathbb{R}^n$ and $\mathbf{D}(t,\boldsymbol{\mu}) = \mathbf{d}^T(t,\boldsymbol{\mu})$ for $\mathbf{d} \in \mathbb{R}^m$. Multiple outputs for $p > 1$ can simply be treated by p separate dual problems. We also assume that the output is only to be estimated at a single certain finite time $T > 0$. In the time-variant case, output estimation at multiple time instants requires the computation of multiple dual problems. In the linear time-invariant case, a single dual problem is sufficient and can then be appropriately "shifted" [11].

We start with the formulation of the parametrized dual problem for the dual variable $x^{du}(t) \in \mathbb{R}^n$, $t \in [0,T]$ by

$$\frac{d}{dt}x^{du}(t) = \mathbf{A}^{du}(t,\boldsymbol{\mu})x^{du}(t) \quad \text{with} \quad \mathbf{A}^{du}(t,\boldsymbol{\mu}) := -\mathbf{G}^{-1}\mathbf{A}^T(t,\boldsymbol{\mu})\mathbf{G}$$

and *final time* condition at time T

$$x^{du}(T) = \mathbf{G}^{-1}\mathbf{c}(T,\boldsymbol{\mu})$$

which is well defined as the inner product matrix $\mathbf{G}$ is assumed to be positive definite. We assume to have a second pair of biorthogonal projection bases $\mathbf{V}^{du}, \mathbf{W}^{du} \in \mathbb{R}^{n\times k^{du}}$ with $(\mathbf{W}^{du})^T\mathbf{V}^{du} = I_{k^{du}\times k^{du}}$. These define the reduced state equation

$$\frac{d}{dt}\hat{x}^{du}(t) = \hat{\mathbf{A}}^{du}(t,\boldsymbol{\mu})x^{du}(t) \quad \text{with} \quad \hat{\mathbf{A}}^{du}(t,\boldsymbol{\mu}) := (\mathbf{W}^{du})^T\mathbf{A}^{du}(t,\boldsymbol{\mu})\mathbf{V}^{du}$$

and final time condition

$$\hat{x}^{du}(T) = (\mathbf{W}^{du})^T\mathbf{G}^{-1}\mathbf{c}(T,\boldsymbol{\mu}).$$

Similarly to the original (primal) problem, we introduce the error

$$e^{du}(t) := x^{du}(t) - \mathbf{V}^{du}\hat{x}^{du}(t)$$

and residual

$$R^{du}(t) := -\mathbf{V}^{du}\frac{d}{dt}\hat{x}^{du}(t) + \mathbf{A}^{du}(t,\boldsymbol{\mu})\mathbf{V}^{du}\hat{x}^{du}(t)$$

and easily verify the dual error evolution equation

$$\frac{d}{dt}e^{du}(t) = \mathbf{A}^{du}(t,\boldsymbol{\mu})e^{du}(t) + R^{du}(t) \tag{15}$$

with final condition

$$e^{du}(T) = (I_{n\times n} - \mathbf{V}^{du}(\mathbf{W}^{du})^T)\mathbf{G}^{-1}\mathbf{c}(t,\boldsymbol{\mu}).$$

We then can formulate an a-posteriori error estimator for the dual problem analogous to Prop. 1

Proposition 2 (Dual A-Posteriori Error Estimate) *We assume that $-\mathbf{A}^{du}(t,\boldsymbol{\mu}) = -\mathbf{A}^{du}(\boldsymbol{\mu})$ is time-invariant. We assume to have a computable bound $C_1^{du}(\boldsymbol{\mu}) \in \mathbb{R}$ with*

$$C_1^{du}(\boldsymbol{\mu}) \geq \sup_{t\in[0,T]} \left\|\exp(-\mathbf{A}^{du}(\boldsymbol{\mu})t)\right\|_{\mathbf{G}}.$$

Then for the dual state variable the following error estimate holds

$$\left\|e^{du}(t,\boldsymbol{\mu})\right\|_{\mathbf{G}} \leq \Delta_x^{du}(t,\boldsymbol{\mu}) := C_1^{du}(\boldsymbol{\mu})\left(\left\|e^{du}(T,\boldsymbol{\mu})\right\|_{\mathbf{G}} + \int_t^T \left\|R^{du}(\tau,\boldsymbol{\mu})\right\|_{\mathbf{G}} d\tau\right).$$

Proof Similar to the proof of Prop. 1 we can write the explicit solution of the error evolution equation (15)

$$e^{du}(t) = \exp(-\mathbf{A}^{du}(T-t))e^{du}(T) - \int_t^T \exp(-\mathbf{A}^{du}(\tau - t))R^{du}(\tau)d\tau.$$

Bounding the corresponding terms yields the claim.

The improved output estimation procedure will be motivated by the following observation. For notational convenience, we suppress the parameter dependence of all quantities, but we still treat a parametrized problem.

Lemma 1 (Output Error Contributions) *For the reduced output error at time T holds*

$$\begin{aligned} y(T) - \hat{y}(T) \;&=\; \left\langle e^{du}(T), e(T)\right\rangle_{\mathbf{G}} + \left\langle \mathbf{V}^{du}\hat{x}^{du}(0), e(0)\right\rangle_{\mathbf{G}} \\ &\quad - \int_0^T \left\langle R^{du(t)}, e(t)\right\rangle_{\mathbf{G}} dt + \int_0^T \left\langle \mathbf{V}^{du}\hat{x}^{du}(t), R(t)\right\rangle_{\mathbf{G}} dt \\ &=: T_1 + T_2 - T_3 + T_4. \end{aligned}$$

Proof Starting from the definitions we obtain

$$\begin{aligned} y(T) - \hat{y}(T) &= \mathbf{c}^T x(T) + \mathbf{d}^T u(T) - (\mathbf{c}^T \mathbf{V}\hat{x}(T) - \mathbf{d}^T u(T)) \\ &= \mathbf{c}^T e(T) = \left\langle \mathbf{G}^{-1}\mathbf{c}, e(T)\right\rangle_{\mathbf{G}} = \left\langle x^{du}(T), e(T)\right\rangle_{\mathbf{G}} \\ &= \left\langle e^{du}(T), e(T)\right\rangle_{\mathbf{G}} + \left\langle \mathbf{V}^{du}\hat{x}^{du}(T), e(T)\right\rangle_{\mathbf{G}}. \end{aligned}$$

As the first term is T_1 it remains to show that the last term equals $T_2 - T_3 + T_4$:

$$\left\langle \mathbf{V}^{du}\hat{x}^{du}(T), e(T)\right\rangle_{\mathbf{G}} = \int_0^T \frac{d}{dt}\left\langle \mathbf{V}^{du}\hat{x}^{du}(t), e(t)\right\rangle_{\mathbf{G}} dt + \left\langle \mathbf{V}^{du}\hat{x}^{du}(0), e(0)\right\rangle_{\mathbf{G}} \tag{16}$$

The last term is T_2 so we remain to show that the first term on the right hand side of (16) is $-T_3 + T_4$. For this we rewrite

$$\begin{aligned} \int_0^T \frac{d}{dt}\left\langle \mathbf{V}^{du}\hat{x}^{du}(t), e(t)\right\rangle_{\mathbf{G}} dt &= \int_0^T \left\langle \frac{d}{dt}\mathbf{V}^{du}\hat{x}^{du}(t), e(t)\right\rangle_{\mathbf{G}} dt \\ &\quad + \int_0^T \left\langle \mathbf{V}^{du}\hat{x}^{du}(t), \frac{d}{dt}e(t)\right\rangle_{\mathbf{G}} dt \end{aligned} \tag{17}$$

The last term can be rewritten by the error evolution equation

$$\begin{aligned} \int_0^T \left\langle \mathbf{V}^{du}\hat{x}^{du}(t), \frac{d}{dt}e(t)\right\rangle_{\mathbf{G}} dt &= \int_0^T \left\langle \mathbf{V}^{du}\hat{x}^{du}(t), \mathbf{A}e(t)\right\rangle_{\mathbf{G}} dt \\ &\quad + \int_0^T \left\langle \mathbf{V}^{du}\hat{x}^{du}(t), R(t)\right\rangle_{\mathbf{G}} dt \end{aligned} \tag{18}$$

The last term is T_4. As $\langle u, \mathbf{A}v\rangle_{\mathbf{G}} = \left\langle \mathbf{G}^{-1}\mathbf{A}^T\mathbf{G}u, v\right\rangle_{\mathbf{G}}$ we can rewrite the first term

$$\int_0^T \left\langle \mathbf{V}^{du}\hat{x}^{du}(t), \mathbf{A}e(t)\right\rangle_{\mathbf{G}} dt = \int_0^T \left\langle -A^{du}\mathbf{V}^{du}\hat{x}^{du}(t), e(t)\right\rangle_{\mathbf{G}} dt$$

So, the sum of the first term on the right hand side of (17) and the first term of (18) give $-T_3$:

$$\begin{aligned} &\int_0^T \left\langle \frac{d}{dt}\mathbf{V}^{du}\hat{x}^{du}(t), e(t)\right\rangle_{\mathbf{G}} dt + \int_0^T \left\langle -A^{du}\mathbf{V}^{du}\hat{x}^{du}(t), e(t)\right\rangle_{\mathbf{G}} dt \\ &\qquad = \int_0^T \left\langle \frac{d}{dt}\mathbf{V}^{du} - A^{du}\mathbf{V}^{du}\hat{x}^{du}(t), e(t)\right\rangle_{\mathbf{G}} dt \\ &\qquad = \int_0^T \left\langle -R^{du}(t), e(t)\right\rangle_{\mathbf{G}} dt. \end{aligned} \tag{19}$$

The relevance of the terms is as follows: T_1 can be made arbitrary small by choosing a good dual reduced space $\mathbf{V}^{du}$, which accurately approximates all $\mathbf{G}^{-1}\mathbf{c}(\boldsymbol{\mu})$, hence the final-time-projection error is small. The term T_2 can similarly be made arbitrary small by ensuring a small primal initial error, e.g. by approximating $x_0(\boldsymbol{\mu})$ well by $\mathbf{V}$. In case of affine parameter dependence of x_0 and $\mathbf{c}$ one can simply include the corresponding affine components in the reduced spaces and hereby eliminate these terms. The contribution T_3 is expected to be "small" by good choice of reduced spaces. More precisely, if we assume that $\left\|R^{du(t)}\right\|_{\mathbf{G}}, \|e(t)\|_{\mathbf{G}} \leq \epsilon$, then $|T_3| \leq T\epsilon^2$. However, the term T_3 cannot be computed precisely during the reduced simulation as $e(t)$ in unavailable. The last term T_4 is of order $O(\epsilon)$ if we assume $\|R(t)\|_{\mathbf{G}} \leq \epsilon$, as we cannot directly relate $\hat{x}^{du}$ to ϵ. So, T_4 contributes most to the overall error. The nice observation is, that this term is computable as it only depends on reduced quantities. So this term can be used as a correction term for the output estimation, overall improving the error bound. Similar argument holds for T_2 if it happens to be nonzero: it is completely computable hence can be used as output correction term. Overall we use T_2 and T_4 as output correction term and bound T_1 and T_3 for the output error estimate. The proof of the following is then clear and omitted here.

Proposition 3 (Improved Output Estimation) *We define the improved output estimate as*

$$\hat{y}^*(T) := \mathbf{c}^T\mathbf{V}\hat{x}(T) + \mathbf{d}^T u(T) + \left\langle \mathbf{V}^{du}\hat{x}^{du}(0), e(0)\right\rangle_{\mathbf{G}} + \int_0^T \left\langle \mathbf{V}^{du}\hat{x}^{du}(t), R(t)\right\rangle_{\mathbf{G}} dt.$$

Then, the following a-posteriori error bound holds:

$$|y(T) - \hat{y}^*(T)| \leq \Delta_y^*(\boldsymbol{\mu}) := \left\|e^{du}(T)\right\|_{\mathbf{G}} \Delta_x(T) + \int_0^T \left\|R^{du}(t)\right\|_{\mathbf{G}} \Delta_x(t)dt, \tag{20}$$

which can be further bounded by

$$\Delta_y^*(\boldsymbol{\mu}) \leq \frac{1}{C_1^{du}} \Delta_x(T, \boldsymbol{\mu}) \Delta_x^{du}(0). \tag{21}$$

Note, that for time-variant systems similar bounds $\Delta_x(t, \boldsymbol{\mu}), \Delta_x^{du}(t, \boldsymbol{\mu})$ as in Prop. 1 and Prop. 2 can be derived and used in the above statement. Equation (21) reveals a typical "quadratical" error effect by the output correction terms. This compact expression is similar to the version for time-discrete parabolic systems [11]. We emphasize, that (20) is a more refined but still rigorous error estimator, which gives more tight error predictions than (21). Assuming an affine parameter dependence as in the previous section, all required quantities can similarly be decomposed in an offline/online fashion. Hence, the scheme again enables rapid evaluation for varying parameters/inputs. Here, we omit these details.

6 Experiments

As a parametrized dynamical system we consider a discretization of a convection problem on a rectangular domain $\Omega = [0, 2] \times [0, 1]$ with end time $T = 1$. The initial and boundary conditions are based on nonnegative radial functions with centers on the top edge and different x_1-positions, all of them are linearly decaying in time and vanishing for $t = T = 1$. Based on these components, the initial/boundary value function then is a convex combination of these depending on a parameter $\mu_0 \in [0, 1]$, such that $\mu_0 = 0$ and $\mu_0 = 1$ recover the leftmost and rightmost cone. The velocity field is chosen as a weighted superposition of two divergence free parabolic velocity fields in x_1 and x_2 direction, which linearly decay in time to 0 at $t = T = 1$. The free weighting parameters are $\mu_1, \mu_2 \in [0, 1]$, which allows to chose no-flow, pure x_1- and x_2-direction flows and their mixtures by suitable parameter choice. The overall parametrization is given by $\boldsymbol{\mu} = (\mu_0, \mu_1, \mu_2)^T \in [0, 1]^3$. Fig. 1 illustrates some solution snapshots, the upper row for $\boldsymbol{\mu} = (1, 1, 0.5)^T$ the lower row for $\boldsymbol{\mu} = (0, 0.5, 1)^T$.

As output we choose the concentration average in the lower right quarter $\Omega_o = [1, 2] \times [0, 0.5]$. The space discretization is obtained by a regular triangular grid of 65536 triangles and an upwind finite volume discretization of the convective flux.

The resulting dynamical system has dimension $n = 65536$, single input $m = 1$ and single output $p = 1$. The parameter and time variation is obtained by fixing the input $u(t) \equiv 1$ and varying the

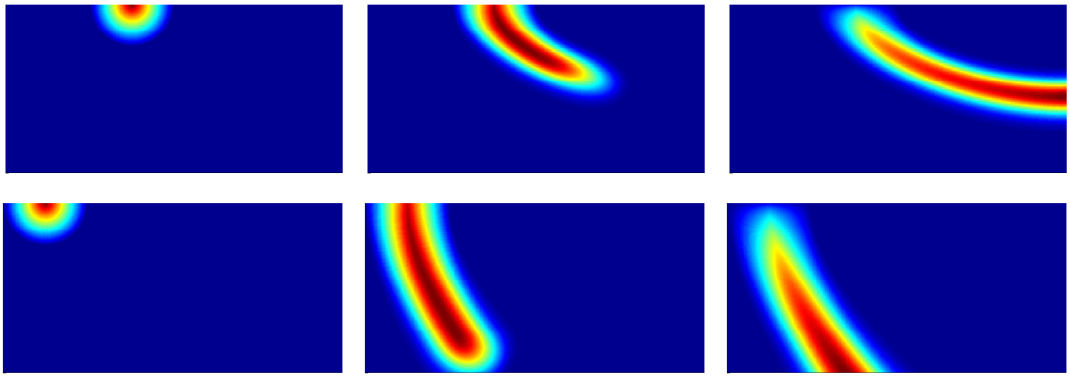

Fig. 1 Illustration of solution variation. Upper row: $\boldsymbol{\mu} = (1, 1, 0.5)^T$ at times $t = 0.0, 0.25, 1.0$, lower row: $\boldsymbol{\mu} = (0, 0.5, 1)^T$ at times $t = 0.0, 0.25, 1.0$.

matrices. The matrices $\mathbf{A}(t, \boldsymbol{\mu})$ and $\mathbf{B}(t, \boldsymbol{\mu})$ are both parametric and time-dependent with $Q_A = 2$ and $Q_B = 6$ components. The output matrix $\mathbf{C}$ is constant and $\mathbf{D} = 0$. The initial data consists of $Q_{x_0} = 3$ components and the matrix $\mathbf{G}$ is given by the finite volume mass-matrix of the discretization, which is diagonal. For the time-discretization we use the first order Euler-forward scheme with 2048 timesteps for both the detailed and the reduced system.

Here we construct a simple basis by computing the trajectories of $\boldsymbol{\mu} = (0, 0, 0)^T$ and $\boldsymbol{\mu} = (1, 1, 1)^T$, storing 129 equally distributed snapshots in time for each of them and performing a POD resulting in overall 127 basis vectors $\mathbf{V}$. As a POD-basis is orthogonal with respect to $\langle \cdot, \cdot \rangle_{\mathbf{G}}$, we can simply choose $\mathbf{W} = \mathbf{GV}$ and ensure biorthogonality. This basis choice is not expected to be well approximating the global parameter space, but certain regions around the training parameters. There are several possibilities for better global basis generation of parametrized problems, e.g. by optimization [6], [25], or by accumulative Greedy algorithms, [18] combined with POD approaches [13], and adaptivity [12]. For simplicity we chose the above trajectory-based approach, as the framework is independent of the basis choice.

The typical output estimation results of the scheme are illustrated in Fig. 2 for $\boldsymbol{\mu} = (1, 1, 1)^T$ and varying reduced dimensions $k = 60, 40, 20$ from left to right. Each plot indicates the exact and reduced output estimate, which are indiscriminable due to the accuracy of the reduced model. Additionally the rigorous upper and lower output bounds $\hat{y}(t, \boldsymbol{\mu}) \pm \Delta_y(t, \boldsymbol{\mu})$ are plotted. The development of the output reflects the underlying physics: It takes some time for the concentration flare to reach the lower right corner (output 0 at initial times), then the concentration in the output domain rises and at the end of the time-interval decreases again, as relevant parts of the concentration have left the domain. Obviously, the true output is located within the error bounds in all cases which confirms the theoretical rigor of the a-posteriori error-estimators. As expected from the integral formula (7), the error-bound width is increasing in time, which makes the estimates useful for finite-time error statements rather than for long or infinite-time statements. Comparing the plots, the bounds are getting less accurate with reduced model dimension. Extrapolating the rightmost plot, there is a critical lower dimension bound, at which the error bar width is in the order of the reduced output itself. In these cases, the rigor of the error bars obviously is of no practical use. But for well-approximating reduced models, the error bars can indeed be very small. Ideally, the error bars can even be evaluated to be 0, if the reduced model is exact. Hence, exact approximation can be verified and identified a-posteriori in a fast way without any expensive operations of order $O(n)$. Note, that the effectivity indices of the a-posteriori error estimators of Sec. 4, i.e. the ratio of error to error-estimator still ranges over several orders. This motivates to use the estimator-improvements by the dual problem of Sec. 5. We refrain from presenting such results here, these will follow in a subsequent report.

The quality of the reduced model in parameter domain can now be quantified during the online-phase, as the reduced model enables parameter sweeps. Fig. 3 indicates the output error bound for linearly varying the parameter vector from $\boldsymbol{\mu} = (0, 0, 0)^T$ to $\boldsymbol{\mu} = (1, 1, 1)^T$. These curves are plotted for different reduced model dimensions k. As expected and accepted from the basis construction, the error

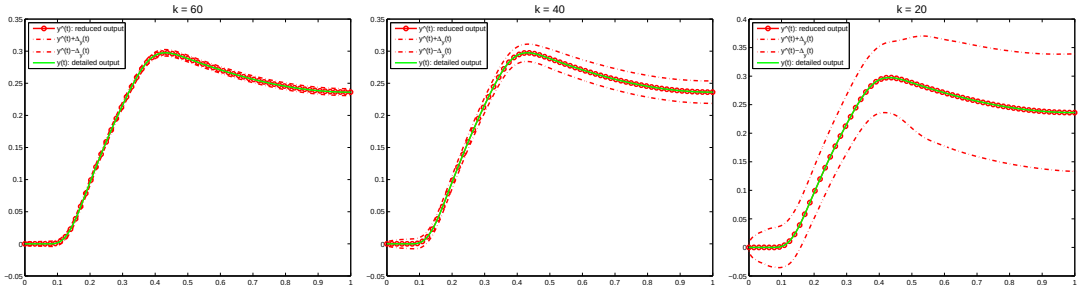

Fig. 2 Output estimation over time for $\boldsymbol{\mu} = (1,1,1)^T$: reduced output with guaranteed error bars and exact solution. From left to right we use reduced model dimensions $k = 60, 40, 20$.

estimators confirm that the resulting basis is good for the boundary parameters but not suitable for intermediate parameters. Interestingly, the basis for $k = 1$ is already excellent for the lowest parameter $\boldsymbol{\mu} = (0,0,0)^T$, which is due to the fact, that the process is stationary in this case, as the velocities are zero. Hence, a single vector is sufficient to exactly(!) reproduce this solution trajectory which is confirmed by $\Delta_x(t, \boldsymbol{\mu}) = 0$. For the right-limit parameter $\boldsymbol{\mu} = (1,1,1)^T$ the model is also becoming accurate with few 16–32 basis vectors.

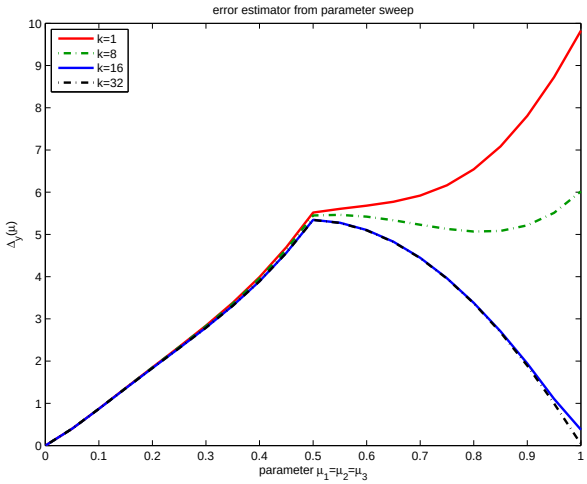

Fig. 3 Output error bound for parameter sweeps along the diagonal of the parameter domain $[0,1]^3$ for different reduced dimensions $k = 1, 8, 16, 32$.

We now want to address the claimed "efficiency" of the resulting models in terms of computation time. For this, we present the runtimes of a single detailed simulation and the online-phase of the reduced simulations with different dimensionalities, with and without error estimation. Table 6 summarizes these results obtained on a standard computer (Intel-Centrino Duo, 2 GHz, 1 GB RAM). We clearly see, how the reduced models realize an acceleration of factor 10-25. This is quite remarkable, due to the simplicity of the time-discretization scheme. The explicit time discretization of the detailed system has complexity $O(n^\alpha)$ for $\alpha = 1$ in each step, dominated by the single sparse matrix-vector multiplication. So currently we even can accelerate these simple matrix-vector multiplications with the reduced system. Therefore, we expect to obtain even higher acceleration factors if we use implicit discretization schemes, which typically scale with $\alpha \in [1,3]$. We further see, that the error-estimation procedure roughly doubles the reduced simulation time, which is understandable in view of the additionally required quantities, cf. Sec. 4, which must be assembled and integrated. Overall, the runtime-dependency on the reduced model dimension k is not so expressed as might be expected. Instead, the runtimes indicate a relevant dominating k-independent overhead. In the current case, this

Table 1 Runtime comparison of detailed and reduced schemes with/without error-estimation and varying k, results averaged over 10 runs.

Simulation	Dimension	Runtime [s]	
		with err.-est.	without err.-est.
detailed	$n = 65536$	74.3	
reduced	$k = 120$	6.4	3.4
reduced	$k = 100$	5.5	3.2
reduced	$k = 80$	4.9	3.0
reduced	$k = 60$	4.3	2.8
reduced	$k = 40$	3.8	2.8
reduced	$k = 20$	3.6	2.8

is the evaluation of the model's online-coefficients and the linear-combination loops for assembling the matrices, which are required at each of the 2048 time-steps.

7 Conclusions

We presented a method for obtaining fast reduced models for parametrized dynamical systems, which is motivated by RB-methods. In view of a multi-query context the resulting offline/online decomposition is an effective algorithmical approach. In particular, the reduced basis is constructed in the offline phase, where time-extensive trajectory-based algorithms are well accepted. We presented an example of a-posteriori error estimation for state variable and system outputs. The error estimators also allow a full offline/online decomposition. This means, that the reduced simulation not only produces the fast reduced solution, but also a fast and rigorous estimate of the error pointwise in time. These are provided in an online complexity, which is completely independent of the dimensionality n.

These a-posteriori error estimators of course are also applicable for non-parametrized problems. In particular the estimators can be used for several standard MOR techniques, such as modal reduction, moment-matching, balanced truncation, etc. As these methods are frequently claimed to lack efficient a-posteriori error control, our estimators are a contribution towards certification of these traditional reduction techniques.

Perspectives are treatment of high-dimensional parameter or input spaces. For example, if the dimension m of the input variable $u(t)$ is too large, the reduced scheme still may be expensive to simulate. An additional parametrization and assumption of separable decomposition of u may be beneficial, i.e. $u(t, \boldsymbol{\mu}) = \sum_{q=1}^{Q_u} \sigma_u^q(t, \boldsymbol{\mu}) u^q$. By choosing 'typical' input signals u^q, which may be available in practice, this allows to model arbitrary linear combinations of the components u^q over time by suitable choice of coefficient function σ_u^q.

8 Acknowledgement

The first author acknowledges funding by the Landesstiftung Baden Württemberg gGmbH and would like to thank the German Research Foundation (DFG) for financial support within the Cluster of Excellence in Simulation Technology (EXC 310/1) at the University of Stuttgart.

References

1. A.C. Antoulas. *Approximation of Large–Scale Dynamical Systems*. SIAM Publications, Philadelphia, PA, 2005.
2. E. Balmes. Parametric families of reduced finite element models, theory and applications. *Mechanical Systems and Signal Processing*, 10(4):381–394, 1996.
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 Series I*, 339:667–672, 2004.
4. U. Baur and P. Benner. Parametrische modellreduktion mit dünnen gittern. In *GMA-Fachausschuss 1.30, Modellbildung, Identifizierung und Simulation in der Automatisierungstechnik, Salzburg ISBN 978-3-9502451-3-4*, 2008.

5. B. Bond and L. Daniel. Parameterized model order reduction of nonlinear dynamical systems. In *Proceedings of the IEEE Conference on Computer-Aided Design*, 2005.
6. T. Bui-Thanh. *Model-Constrained Optimization Methods for Reduction of Parameterized Large-Scale Systems*. PhD thesis, Massachusetts Institute of Technology, 2007.
7. L. Daniel, O. Siong, L. Chay, K. Lee, and J. White. Multi-parameter moment-matching model-reduction approach for generating geometrically parameterized interconnect performance models. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 23(5):678–693, 2004.
8. O. Farle, V. Hill, P. Ingelström, and R. Dyczij-Edlinger. Ordnungsreduktion linearer zeitinvarianter Finite-Elemente-Modelle mit multivariater polynomieller Parametrisierung (model order reduction of linear finite element models parameterized by polynomials in several variables). *at - Automatisierungstechnik*, 54:161–169, 2006.
9. L. Feng and P. Benner. A robust algorithm for parametric model order reduction. In *Proceedings in Applied Mathematics and Mechanics*, 2007.
10. 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(3):575–605, 2007.
11. 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(1):157–181, 2005.
12. B. Haasdonk and M. Ohlberger. Adaptive basis enrichment for the reduced basis method applied to finite volume schemes. In *Proc. 5th International Symposium on Finite Volumes for Complex Applications*, pages 471–478, 2008.
13. B. Haasdonk and M. Ohlberger. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *M2AN, Math. Model. Numer. Anal.*, 42(2):277–302, 2008.
14. N. Jung, O. Rain, K. Schumacher, R. Eid, and B. Lohmann. Error estimators for projection-based order reduction methods. In *MoRePaS 09 Book of Abstracts*, 2009.
15. B. Lohmann and R. Eid. Efficient order reduction of parametric and nonlinear models by superposition of locally reduced models. In *Methoden und Anwendungen der Regelungstechnik. Erlangen-Münchener Workshops 2007 und 2008*. Shaker Verlag, Aachen, 2009.
16. M. Meyer and H.G. Matthies. Efficient model reduction in non-linear dynamics using the Karhunen-Loève expansion and dual-weighted-residual methods. *Computational Mechanics*, 31:179–191, 2003.
17. C. Moosmann. *ParaMOR - Model Order Reduction for parameterized MEMS applications*. PhD thesis, IMTEK, University of Freiburg, 2008.
18. A.T. Patera and G. Rozza. *Reduced Basis Approximation and a Posteriori Error Estimation for Parametrized Partial Differential Equations*. MIT, 2007. Version 1.0, Copyright MIT 2006-2007, to appear in (tentative rubric) MIT Pappalardo Graduate Monographs in Mechanical Engineering.
19. 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 Engineering*, 124:70–80, 2002.
20. D.V. Rovas, L. Machiels, and Y. Maday. Reduced basis output bound methods for parabolic problems. *IMA J. Numer. Anal.*, 26(3):423–445, 2006.
21. G. Rozza, D.B.P. Huynh, and A.T. Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: application to transport and continuum mechanics. *Arch. Comput. Meth. Eng.*, 15(3):229–275, 2008.
22. E. B. Rudnyi, C. Moosmann, A. Greiner, T. Bechtold, and J. G. Korvink. Parameter preserving model reduction for MEMS system-level simulation and design. In *Proc. of MATHMOD 2006*, 2006.
23. T. Tonn and K. Urban. A reduced-basis method for solving parameter-dependent convection-diffusion problems around rigid bodies. In P. Wesseling, E. Onate, and J. Periaux, editors, *ECCOMAS CFD 2006 Proceedings*, 2006.
24. S. Volkwein. Model reduction using proper orthogonal decomposition, 2008. Lecture notes, University of Graz.
25. S. Volkwein and K. Kunisch. Optimal snapshot location for computing POD basis functions. Technical report, University of Graz, 2008.
26. S. Weile, E. Michielssen, E. Grimme, and K. Gallivan. A method for generating rational interpolant reduced order models of two- parameter linear systems. *Appl. Math. Lett.*, 12(5):93–102, 1999.