Universität Konstanz

A-posteriori error analysis for lithium-ion concentrations in batteries utilizing the reduced basis method

Laura Iapichino
Stefan Volkwein
Andrea Wesche

Konstanzer Schriften in Mathematik

Nr. 342, Oktober 2015

ISSN 1430-3558

A-posteriori error analysis for lithium-ion concentrations in batteries utilizing the reduced basis method

L. Iapichino[a], S. Volkwein[b*], and A. Wesche[b]

[a]*Department of Precision and Microsystems Engineering, Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands;* [b]*Department of Mathematics and Statistics, University of Konstanz, Germany*

(*October 16, 2015*)

In the present paper the authors consider a parametrized nonlinear parabolic differential equation, which is motivated by lithium-ion battery models. A standard finite volume (FV) discretization leads to a high-dimensional discrete nonlinear problem so that simulation of the parametrized problem for various different parameters is very costly. Therefore, the reduced-basis method is applied, so that the number of degrees of freedom is reduced significantly and a fast numerical simulation of the model is possible. To control the error, an a-posteriori error estimator is derived. Numerical experiments show the efficiency of the approach.

Keywords: finite volume schemes, reduced basis method, pod-greedy algorithm, a-posteriori error analysis, parameter estimation

AMS Subject Classification: 35K58, 65M08, 65K10, 90C53

1. Introduction

The interest in lithium-ion batteries has been increased in the recent years. Several companies worldwide are developing such batteries for consumer electronic applications, in particular, for electric-vehicle applications. To achieve the performance and lifetime demands in this area exact mathematical models of the battery are required. In the present work we consider a parametrized partial differential equation ($\boldsymbol{\mu}$PDE) that occurs in lithium-ion battery models (see [20, 21]) as an equation for the concentration of lithium-ions. This equation describes the mass transport in the (positive) electrode of a battery. Let us refer to [6, 32], for instance, for a different system of $\boldsymbol{\mu}$PDEs describing lithium-ion batteries.

The discretization of the nonlinear $\boldsymbol{\mu}$PDE using the (cell-centered) finite volume (FV) techniques (see, e.g., [3]), leads to very large systems that are expensive to solve. The goal is to develop a reduced-order model for the parametrized PDE that is cheap to evaluate. This is motivated by applications like parameter estimations, optimal control and design, where repeated evaluations of the nonlinear equation are required. In the present paper, the spatial approximation is realized by applying the reduced-basis (RB) method [12, 24], where the FV solution has to be computed for certain different values of the parameters. The RB solution of the $\boldsymbol{\mu}$PDE for

[*] Corresponding author. Email: Stefan.Volkwein@uni-konstanz.de

any admissible parameter $\boldsymbol{\mu} \in \mathcal{D}$ is computed (during the *online* stage) as Galerkin projection into the RB space defined by a typically small set of solutions of the $\boldsymbol{\mu}$PDE computed during the *offline* stage and corresponding to a small subset of parameter values $\mathcal{D}_{\text{train}} \subsetneq \mathcal{D}$ accurately chosen. A combination of the usual greedy strategy and the method of proper orthogonal decomposition (the *POD-greedy algorithm*; c.f. [13]) is used as the classical sampling strategy to select parameter values that define the set of basis functions. To require as little FV solutions as possible, a rigorous and quickly evaluated a-posteriori error estimate is needed in the greedy method; see [7, 8] for parabolic problems. We derived an a-posteriori error estimate for the $\boldsymbol{\mu}$PDE under consideration, where we have to deal carefully with the nonlinear term. This theoretical result is utilized in our numerical experiments to construct a RB basis. After obtaining an efficient reduced order model we want to utilize it in a parameter estimation problem. For RB discretization of the coupled lithium-ion battery model we refer to [30, 31]. Further, POD reduced-order modeling for simulation and parameter estimation of battery models is utilized in [1] and [18, 19], for instance.

The paper is organized in the following manner: In Section 2 we introduce the $\boldsymbol{\mu}$PDE describing the mass transport in the (positive) electrode of a battery. The FV scheme is briefly explained in Section 3. The RB and POD method are explained in Section 4. Further, the POD-greedy algorithm is recalled. Section 5 is devoted to the a-posteriori error analysis. A parameter estimation problem is introduced in Section 6 and numerical tests are presented in Section 7.

2. Problem formulation

For $L > 0$ let $\Omega = (0, L) \subset \mathbb{R}$ be the spatial interval, $T > 0$ the final time and $Q = \Omega \times (0, T)$. By $H = L^2(\Omega)$ we denote the Lebesgue space of (equivalence classes of) functions which are (Lebesgue) measurable and square integrable. Furthermore, $V = H^1(\Omega) \subset H$ stands for the Sobolev space

$$V = \left\{ \varphi \in H \,\Big|\, \int_\Omega \left|\varphi'(x)\right|^2 \mathrm{d}x < \infty \right\}.$$

For more details on Lebesgue and Sobolev spaces we refer to [5], for instance. The space $L^2(0, T; V)$ stands for the space of (equivalence classes) of measurable abstract functions $\varphi : [0, T] \to V$, which are square integrable, i.e.,

$$\int_0^T \|\varphi(t)\|_V^2 \, \mathrm{d}t < \infty.$$

When t is fixed, the expression $\varphi(t)$ stands for the function $\varphi(t, \cdot)$ considered as a function in Ω only. Recall that

$$W(0, T) = \{\varphi \in L^2(0, T; V) \,|\, \varphi_t \in L^2(0, T; V)\}$$

is a Hilbert space supplied with its corresponding inner product; see, e.g., [5].

The set of admissible parameters is given as

$$\mathcal{D} = \left\{ \boldsymbol{\mu} = (\mu_1, \mu_2) \in \mathbb{R}^2 \,\big|\, \mu_i^a \le \mu_i \le \mu_i^b \text{ for } i = 1, 2 \right\}$$

with $0 < \mu_i^a \le \mu_i^b$ for $i = 1,2$. For an arbitrary $\boldsymbol{\mu} = (\mu_1, \mu_2) \in \mathcal{D}$ we consider the semilinear parabolic problem

$$\frac{\partial c}{\partial t}(t,x) - \frac{\partial}{\partial x}\Big(\mu_1 \frac{\partial c}{\partial x}(t,x)\Big) = 0 \quad \text{for all } (t,x) \in Q \tag{1a}$$

together with the Neumann and Robin boundary conditions

$$\frac{\partial c}{\partial x}(t,0) = 0, \quad \mu_1 \frac{\partial c}{\partial x}(t,L) = -\mu_2 \sqrt{c(t,L)} \quad \text{for all } t \in [0,T] \tag{1b}$$

and the initial condition

$$c(0,x) = c_\circ(x) \quad \text{for all } x \in \Omega. \tag{1c}$$

We assume that $c_\circ$ belongs to $C(\overline{\Omega})$ and satisfies

$$0 < c_\circ^a \le c_\circ(x) \le c_\circ^b \quad \text{for all } x \in \overline{\Omega} \tag{2}$$

with constants $0 < c_\circ^a \le c_\circ^b < \infty$. Moreover, the concentration should be at least non-negative so that we can evaluate the square root term in (1b).

Remark 2.1: 1) Problem (1) is motivated by a system of partial differential equations which is used as a model for lithium-ion batteries; see [20, 21]. More precisely, (1) describes the mass transport in the (positive) electrode of a lithium-ion battery and c stands for the concentration of lithium ions. The nonlinear boundary condition in (1b) contains the c-dependent prefactor of the Butler-Volmer equation describing the exchange of the lithium ions at the interfaces between the electrode and the electrolyte [2].

2) Since the results of a one-dimensional battery model are comparable to the ones obtained by the associated three-dimensional model [31], the spatial domain is chosen to be an interval in the present paper. ◇

For a given parameter $\boldsymbol{\mu} \in \mathcal{D}$ a function $c = c(\boldsymbol{\mu}) \in L^2(0,T;V)$ is called a *weak solution* to (1) provided c satisfies $c > 0$ in Q and

$$\int_0^T \int_\Omega -c(t)\varphi_t(t) + \mu_1 c_x(t)\varphi_x(t)\,\mathrm{d}x\mathrm{d}t + \mu_2 \int_0^T \sqrt{c(t,L)}\varphi(t,L)\,\mathrm{d}t = \int_\Omega c_\circ \varphi(0)\,\mathrm{d}x$$

holds for all $\varphi \in H^1(0,T;H) \cap L^2(0,T;V)$ with $\varphi(T) = 0$ in Ω.

Assumption 2.2: *For any $\boldsymbol{\mu} \in \mathcal{D}$ there exists a unique weak solution $c = c(\boldsymbol{\mu}) \in W(0,T) \cap C(\overline{Q})$ to* (1) *satisfying*

$$\mathsf{c}_{\mathsf{min}} \le c(t,x) \le \mathsf{c}_{\mathsf{max}} \quad \textit{for all } (t,x) \in \overline{Q} \tag{3}$$

with $\boldsymbol{\mu}$-independent constants $0 < \mathsf{c}_{\mathsf{min}} \le \mathsf{c}_{\mathsf{max}}$.

Remark 2.3: Let $\boldsymbol{\mu} \in \mathcal{D}$ and (2) hold. Then, it follows from [25, Proposition 3.3] that there exists a time $T_\circ > 0$ such that (1) admits a unique solution in $W(0,T_\circ) \cap C(\overline{Q})$. This solution satisfies (3) in $\overline{Q}_\circ = [0,T_\circ] \times \overline{\Omega}$. However, it is not a-priorly clear that we have $T_\circ \geq T$, so that we can only ensure the unique existence of a positive weak solution locally in time. ◇

3. Finite volume (FV) discretization

We discretize (1) by the classical FV method; see, e.g., [3]. The integration over time is realized by the backward Euler method. For given $\mathcal{N} \in \mathbb{N}$ let $h = L/\mathcal{N}$ be the equidistant spatial grid size. Define the center points $x_i = h/2 + (i-1)h$ for $i = 1, \dots, \mathcal{N}$. We devide Ω into $\mathcal{N}$ subintervals $\Omega_i = (x_i - h/2, x_i + h/2)$. Moreover, for given $K \in \mathbb{N}$ let $k = T/(K-1)$ be the equidistant time grid size and $t_j = (j-1)k$, $j = 1, \dots, K$, the temporal grid points. By $\mathrm{c}^h_{j,i}(\boldsymbol{\mu})$, $i = 1, \dots, \mathcal{N}$ and $j = 1, \dots, K$, we denote the FV approximation of the concentration $c(\boldsymbol{\mu})$ at $(t_j, x_i) \in \overline{Q}$ for a given parameter $\boldsymbol{\mu} \in \mathcal{D}$. We define the vector

$$\mathrm{c}^h_j(\boldsymbol{\mu}) = \begin{pmatrix} c^h_{j,1}(\boldsymbol{\mu}) \\ \vdots \\ c^h_{j,\mathcal{N}}(\boldsymbol{\mu}) \end{pmatrix} \in \mathbb{R}^{\mathcal{N}} \quad \text{for } j = 1, \dots, K \text{ and } \boldsymbol{\mu} \in \mathcal{D}.$$

Proceeding in the standard way we derive the following FV scheme for given parameter $\boldsymbol{\mu} \in \mathcal{D}$:

$$0 \overset{!}{=} \mathrm{F}^h_j\big(\mathrm{c}^h_j(\boldsymbol{\mu}); \boldsymbol{\mu}\big) := \mathrm{L}^h_{\mathsf{im}}(\mu_1)\mathrm{c}^h_j(\boldsymbol{\mu}) - \mathrm{L}^h_{\mathsf{ex}}\mathrm{c}^h_{j-1}(\boldsymbol{\mu}) + \mu_2 \mathrm{g}^h\big(\mathrm{c}^h_j(\boldsymbol{\mu})\big) \tag{4}$$

for $j = 2, \dots, K$, where we have used the matrices for the implicit and explicit parts

$$\mathrm{L}^h_{\mathsf{im}}(\mu_1) = h\mathrm{I}^h + \mu_1 \mathrm{S}^h \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}}, \quad \mathrm{L}^h_{\mathsf{ex}} = h\mathrm{I}^h \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}},$$

respectively, with the identity matrix I^h and the stiffness matrix

$$\mathrm{S}^h = \frac{k}{h}\begin{pmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{pmatrix} \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}}, \; \mathrm{g}^h\big(\mathrm{c}^h_j(\boldsymbol{\mu})\big) = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ k\sqrt{\mathrm{c}^h_{j,\mathcal{N}}(\boldsymbol{\mu})} \end{pmatrix} \in \mathbb{R}^{\mathcal{N}}.$$

Moreover, we put $\mathrm{c}^h_{1,i}(\boldsymbol{\mu}) = c_\circ(x_i)$ for $1 \leq i \leq \mathcal{N}$. In the following assumption we suppose that the FV scheme admits a unique solution with positive values.

Assumption 3.1: *For any $\boldsymbol{\mu} \in \mathcal{D}$, for any $\mathcal{N} \in \mathbb{N}$ and $K \in \mathbb{N}$ there exists a unique solution sequence $\{\mathrm{c}^{\mathcal{N}}_j(\boldsymbol{\mu})\}_{j=1}^K \subset \mathbb{R}^{\mathcal{N}}$ satisfying $\mathrm{c}^h_{1,i}(\boldsymbol{\mu}) = c_\circ(x_i)$ for $1 \leq i \leq \mathcal{N}$ and (4). Moreover, we have*

$$\mathsf{c}_{\mathsf{min}} \leq \mathrm{c}^h_{j,i}(\boldsymbol{\mu}) \leq \mathsf{c}_{\mathsf{max}} \quad \textit{for } 1 \leq j \leq K \textit{ and } 1 \leq i \leq \mathcal{N} \tag{5}$$

with $\boldsymbol{\mu}$-independent constants $\mathsf{c}_{\mathsf{min}}$ and $\mathsf{c}_{\mathsf{max}}$ introduced in Assumption 2.2.

Remark 3.2: To solve (4) numerically we apply a globalized Newton method [4] starting with the numerical solution of the previous time step. ◇

4. The Reduced Basis (RB) Method

In this section we introduce briefly the RB method for (4). Applying the RB method to (4) we want to diminish significantly the number $\mathcal{N}$ of unknowns in comparison to the FV model via Galerkin projection. We follow the approach presented in [13]. The main idea of the RB method is that a certain number of appropriate FV solutions can be used to construct the RB solutions by a linear combination. Clearly, applying the RB method is only worth it, if one is interested in many function evaluations. Here, we are interested in a parameter estimation for (1); see Sections 6 and 7.2.

4.1. *The Reduced Basis Method for FV schemes*

The first step of the RB method is to replace the parameter set $\mathcal{D}$ by a discrete training set

$$\mathcal{D}_{\text{train}} = \{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_{\mathsf{p}}\} \subset \mathcal{D} \quad \text{with } \mathsf{p} \in \mathbb{N}.$$

We start with the FV solution sequence $\{\mathrm{c}_j^h(\boldsymbol{\mu}^{(1)})\}_{j=1}^K \subset \mathbb{R}^{\mathcal{N}}$ for an initial parameter choice $\boldsymbol{\mu}^{(1)} \in \mathcal{D}_{\text{train}}$. Utilizing the POD method we construct with this information the first RB basis vector(s) in $\mathbb{R}^{\mathcal{N}}$. We estimate the error between the FV and the RB solution constructed with this basis for all parameters $\boldsymbol{\mu} \in \mathcal{D}_{\text{train}}$. If the biggest error is smaller than our predefined tolerance $0 < \varepsilon^{\mathsf{gr}} \ll 1$ we found our reduced basis. If it is above $\varepsilon^{\mathsf{gr}}$, we compute the FV solution to the parameter, where the estimated error is the biggest. This information we add to our already computed basis. Again we estimate the error between the FV and the RB solution to all parameters $\boldsymbol{\mu}$ of our training set $\mathcal{D}_{\text{train}}$, but now we utilize the updated basis for the computation of the RB solution. We proceed with the so-called *greedy algorithm* as long as the estimated error is above $\varepsilon^{\mathsf{gr}}$.

One of the big advantages of the RB method is that the computation can be decomposed into a computationally expensive *offline* and cheap *online* part. In the offline part the RB is determined. Furthermore, all parameter-independent parts of the RB model are computed. In the online part the RB solutions to the requested parameter are computed.

The accuracy of the RB solutions in comparison to the FV solutions in $\mathcal{D}_{\text{train}}$ is given by the tolerance $\varepsilon^{\mathsf{gr}}$. If the mapping $\mathcal{D} \ni \boldsymbol{\mu} \mapsto \mathrm{c}_j^h(\boldsymbol{\mu}) \in \mathbb{R}^{\mathcal{N}}$ is sufficiently regular for $1 \leq j \leq K$ and $\mathcal{D}_{\text{train}}$ is chosen appropriately, the error $\mathrm{c}^h(\boldsymbol{\mu}) - \mathrm{c}^N(\boldsymbol{\mu})$ is close to $\varepsilon^{\mathsf{gr}}$ for all $\boldsymbol{\mu} \in \mathcal{D}$. Let us mention that there are also techniques, where the parameter set is discretized adaptively [11], but we do not follow this approach here.

Considering nonlinear problems some mathematical effort has to be put in developing the error estimator. An error estimator for linear problems is examined in [13]. In [7, 8] a-posteriori error estimates for Galerkin approximations applied to nonlinear parabolic equations are considered.

To find an appropriate basis for our selected parameter set range we need FV solutions to selected parameters. For time dependent problems we apply the POD-greedy algorithm, see Section 4.3: the computed FV solution of the greedy algorithm is reduced in time by the POD method, see e.g. [29]. In this algorithm there are two loops. The *outer* loop is similar to the well-known greedy algorithm for elliptic problems. For the chosen parameter $\boldsymbol{\mu} \in \mathcal{D}_{\text{train}}$ the error between the FV and the RB solution is the biggest. In the *inner* loop we apply the POD method to the snapshot matrix

$$\mathrm{C}^h(\boldsymbol{\mu}) = \big[\mathrm{c}_1^h(\boldsymbol{\mu})\,|\ldots|\,\mathrm{c}_K^h(\boldsymbol{\mu})\big] \in \mathbb{R}^{\mathcal{N}\times K} \quad \text{for } \boldsymbol{\mu} \in \mathcal{D}_{\mathrm{train}}.$$

Let us introduce a weighted inner product in $\mathbb{R}^{\mathcal{N}}$ by

$$\langle \mathrm{x}, \mathrm{y}\rangle_{\mathrm{W}} = \mathrm{x}^\top \mathrm{W}\mathrm{y} \quad \text{for } \mathrm{x}, \mathrm{y} \in \mathbb{R}^{\mathcal{N}}$$

with a symmetric and positive definite matrix $\mathrm{W} \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}}$ (i.e., $\mathrm{W} = \mathrm{W}^\top$ and $\mathrm{W} \succ 0$). Suppose that we have computed an orthonormal RB basis $\{\xi_i^N\}_{i=1}^N \subset \mathbb{R}^{\mathcal{N}}$ by the POD-greedy algorithm with $\mathbb{N} \ni N \ll \mathcal{N}$. We set

$$\Xi = \big[\xi_1\,|\ldots|\,\xi_N\big] \in \mathbb{R}^{\mathcal{N}\times N}.$$

Using the Galerkin projection we approximate the solution $\mathrm{c}_j^h(\boldsymbol{\mu}) \in \mathbb{R}^{\mathcal{N}}$ by an $\mathcal{N}$-dimensional vector $\mathrm{c}_j^N(\boldsymbol{\mu})$ for any $j \in \{1, \ldots, K\}$. Now, the RB approximation is given by the vectors

$$\mathrm{c}_j^N(\boldsymbol{\mu}) = \begin{pmatrix} \mathrm{c}_{j,1}^N(\boldsymbol{\mu}) \\ \vdots \\ \mathrm{c}_{j,\mathcal{N}}^N(\boldsymbol{\mu}) \end{pmatrix} = \sum_{i=1}^N \mathfrak{c}_{j,i}^N(\boldsymbol{\mu})\xi_i = \Xi\mathfrak{c}_j^N(\boldsymbol{\mu}) \in \mathbb{R}^{\mathcal{N}} \text{ for } j = 1, \ldots, K \text{ and } \boldsymbol{\mu} \in \mathcal{D},$$

where we set $\mathfrak{c}_j^N(\boldsymbol{\mu}) = (\mathfrak{c}_{j,1}^N(\boldsymbol{\mu}), \ldots, \mathfrak{c}_{j,N}^N(\boldsymbol{\mu}))^\top \in \mathbb{R}^N$. We replace $\mathrm{c}_j^h(\boldsymbol{\mu})$ in (4) by $\mathrm{c}_j^N(\boldsymbol{\mu})$ and multiply by $\xi_i^\top\mathrm{W}$ from the left for $1 \le i \le N$. Then, for any $\boldsymbol{\mu} \in \mathcal{D}$ we derive the equations

$$0 \overset{!}{=} \big\langle \xi_i, \mathrm{F}_j^h\big(\mathrm{c}_j^N(\boldsymbol{\mu}); \boldsymbol{\mu}\big)\big\rangle_{\mathrm{W}} = \xi_i^\top \mathrm{W}\mathrm{F}_j^h\big(\mathrm{c}_j^N(\boldsymbol{\mu}); \boldsymbol{\mu}\big) \quad \text{for } i = 1, \ldots, N \text{ and } j = 2, \ldots, K$$

which can be expressed as

$$0 \overset{!}{=} \mathrm{F}_j^N\big(\mathrm{c}_j^N(\boldsymbol{\mu}); \boldsymbol{\mu}\big) \coloneqq \Xi^\top \mathrm{W}\mathrm{F}_j^h\big(\mathrm{c}_j^N(\boldsymbol{\mu}); \boldsymbol{\mu}\big) \in \mathbb{R}^N \quad \text{for } j = 2, \ldots, K \tag{6}$$

of N nonlinear equations for the N unknowns coefficients $\{\mathfrak{c}_{j,i}^N(\boldsymbol{\mu})\}_{i=1}^N$. More precisely, we find

$$\mathrm{F}_j^N\big(\mathrm{c}_j^N(\boldsymbol{\mu}); \boldsymbol{\mu}\big) = \mathrm{L}_{\mathsf{im}}^N(\mu_1)\mathfrak{c}_j^N(\boldsymbol{\mu}) - \mathrm{L}_{\mathsf{ex}}^N\mathfrak{c}_{j-1}^N(\boldsymbol{\mu}) + \mu_2\mathrm{g}^N\big(\Xi\mathfrak{c}_j^N(\boldsymbol{\mu})\big) \in \mathbb{R}^N, \tag{7}$$

where we have set the matrices

$$\mathrm{L}_{\mathsf{im}}^N(\mu_1) = \Xi^\top\mathrm{W}\mathrm{L}_{\mathsf{im}}^h(\mu_1)\Xi \in \mathbb{R}^{N\times N}, \quad \mathrm{L}_{\mathsf{ex}}^N = \Xi^\top\mathrm{W}\mathrm{L}_{\mathsf{ex}}^h\Xi \in \mathbb{R}^{N\times N}$$

and the vector

$$\mathrm{g}^N\big(\Xi\mathfrak{c}_j^N(\boldsymbol{\mu})\big) = \Xi^\top\mathrm{W}\mathrm{g}^h\big(\Xi\mathfrak{c}_j^N(\boldsymbol{\mu})\big) \in \mathbb{R}^N.$$

Analogous to Assumption 3.1 we suppose the following hypothesis.

Assumption 4.1: *For any $\boldsymbol{\mu} \in \mathcal{D}$, for any $N \in \mathbb{N}$ and $K \in \mathbb{N}$ there exists a unique solution sequence $\{\mathrm{c}_j^N(\boldsymbol{\mu})\}_{j=1}^K \subset \mathbb{R}^{\mathcal{N}}$ satisfying* (6) *and*

$$\mathrm{c}_{\mathsf{min}} \le \mathrm{c}_{j,i}^N(\boldsymbol{\mu}) \le \mathrm{c}_{\mathsf{max}} \quad \textit{for } 1 \le j \le K \textit{ and } 1 \le i \le \mathcal{N} \tag{8}$$

with $\boldsymbol{\mu}$-independent constants $\mathrm{c}_{\mathsf{min}}$ *and* $\mathrm{c}_{\mathsf{max}}$ *introduced in Assumption* 2.2.

4.2. *Proper Orthogonal Decomposition (POD)*

The POD method is used for the POD-greedy algorithm. It reduces the snapshot matrix $\mathrm{C}^h(\boldsymbol{\mu}) \in \mathbb{R}^{\mathcal{N}\times K}$ in time and extracts the essential characteristics. To get $\ell \le d = \operatorname{rank} \mathrm{C}^h(\boldsymbol{\mu})$ POD basis vectors and associated POD modes the following optimization problem is solved [15, 16, 29]:

$$\begin{cases} \min J(\psi_1,\dots,\psi_\ell) = \sum\limits_{j=1}^K \alpha_j \left\| \mathrm{c}_j^h(\boldsymbol{\mu}) - \sum\limits_{i=1}^\ell \left\langle \mathrm{c}_j^h(\boldsymbol{\mu}), \psi_i \right\rangle_{\mathrm{W}} \psi_i \right\|_{\mathrm{W}}^2 \\ \text{subject to } \{\psi_i\}_{i=1}^\ell \subset \mathbb{R}^{\mathcal{N}} \text{ and } \langle \psi_i, \psi_j \rangle_{\mathrm{W}} = \delta_{ij} \text{ for } 1 \le i,j \le \ell, \end{cases} \tag{9}$$

where $\|\cdot\|_{\mathrm{W}} = \langle\cdot,\cdot\rangle_{\mathrm{W}}^{1/2}$ denotes the weighted norm in $\mathbb{R}^{\mathcal{N}}$, δ_{ij} denotes the Kronecker symbol and $\alpha_1 = \alpha_M = \Delta t/2$, $\alpha_j = \Delta t$, $2 \le j \le K-1$, are the trapezoidal weights associated with or temporal grid; see [29, Section 4]. The solution of the optimization problem (9) is characterized by the first ℓ eigenvectors $\{\psi_i\}_{i=1}^\ell$ solving the symmetric eigenvalue problem

$$\sum_{j=1}^K \alpha_j \left\langle \mathrm{c}_j^h(\boldsymbol{\mu}), \psi_i \right\rangle_{\mathrm{W}} \mathrm{c}_j^h(\boldsymbol{\mu}) = \lambda_i \psi_i \quad \text{for } i = 1,\dots,\mathcal{N} \tag{10}$$

with $\lambda_1 \ge \ldots \ge \lambda_d > \lambda_{d+1} = \ldots = \lambda_{\mathcal{N}} = 0$. In particular, we have [16, Proposition 1]

$$\mathcal{E}(\ell) := \sum_{j=1}^K \alpha_j \left\| \mathrm{c}_j^h(\boldsymbol{\mu}) - \sum_{i=1}^\ell \left\langle \mathrm{c}_j^h(\boldsymbol{\mu}), \psi_i \right\rangle_{\mathrm{W}} \psi_i \right\|_{\mathrm{W}}^2 = \sum_{i=\ell+1}^d \lambda_i.$$

We introduce the POD space $V^\ell = \operatorname{span}\{\psi_1,\dots,\psi_\ell\} \subset \mathbb{R}^{\mathcal{N}}$ and the orthogonal projection $\mathcal{P}^\ell : \mathbb{R}^{\mathcal{N}} \to V^\ell$ by

$$\mathcal{P}^\ell \mathrm{x} = \sum_{i=1}^\ell \left\langle \mathrm{x}, \psi_i \right\rangle_{\mathrm{W}} \psi_i = \Psi\Psi^\top \mathrm{W}\mathrm{x} \quad \text{for } \mathrm{x} \in \mathbb{R}^{\mathcal{N}}.$$

with $\Psi = [\psi_1 | \ldots | \psi_\ell] \in \mathbb{R}^{\mathcal{N}\times\ell}$. In Algorithm 4.1 the POD method is summarized.

4.3. *Basis computation by the POD-greedy algorithm*

Suppose that $\|\cdot\|$ is a (vector) norm on $\mathbb{R}^{\mathcal{N}}$. The POD-greedy algorithm relies essentially on the availability of a sharp error estimator $\Delta^N(\boldsymbol{\mu})$ for the error between

Algorithm 4.1 (POD algorithm)

```
Require: $\mathrm{C}^h(\boldsymbol{\mu}) \in \mathbb{R}^{\mathcal{N}\times K}$, $0 \ll \varepsilon^{\mathsf{pod}} < 1$. $\mathrm{W} \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}}$ with $\mathrm{W} = \mathrm{W}^\top$ and $\mathrm{W} > 0$;
$d \leftarrow \operatorname{rank} \mathrm{C}^h(\boldsymbol{\mu})$;
$\{(\psi_i, \lambda_i)\}_{i=1}^d \leftarrow$ solve (10);
$\ell \leftarrow 1$
$\mathcal{E}(\ell) \leftarrow \lambda_1$;
while $\mathcal{E}(\ell) < \varepsilon^{\mathsf{pod}}(\boldsymbol{\mu})$ and $\ell < d$ do
    $\mathcal{E}(\ell + 1) \leftarrow \mathcal{E}(\ell) + \lambda_{\ell+1}$;
    $\ell \leftarrow \ell + 1$;
end while
return POD basis $\{\psi_i\}_{i=1}^\ell$, eigenvalues $\{\lambda_i\}_{i=1}^\ell$ and error $\mathcal{E}(\ell)$;
```

the FV and the RB solution, i.e.,

$$\mathrm{e}_j^N(\boldsymbol{\mu}) = \left\|\mathrm{c}_j^h(\boldsymbol{\mu}) - \mathrm{c}_j^N(\boldsymbol{\mu})\right\| \le \Delta_j^N(\boldsymbol{\mu}) \quad \text{and} \quad \eta_j^{\mathsf{eff}}(\boldsymbol{\mu}) = \frac{\Delta_j^N(\boldsymbol{\mu})}{\mathrm{e}_j^N(\boldsymbol{\mu})} \approx 1$$

for any $\boldsymbol{\mu} \in \mathcal{D}_{\mathrm{train}}$ and $j \in \{1, \ldots, K\}$, where $\eta_j^{\mathsf{eff}}(\boldsymbol{\mu})$ is called the efficiency (of the a-posteriori error estimator). If $\eta_j^{\mathsf{eff}}(\boldsymbol{\mu}) \gg 1$ holds the RB error is overestimated too much which leads to large number of basis functions. Then, the POD-greedy algorithm and the offline phase are computationally too expensive (and thus the online phase because more basis vectors than needed are used). In the worst case the tolerance for the RB solution is not even reached. The POD-greedy method is presented in Algorithm 4.2 [13].

Algorithm 4.2 (POD-greedy algorithm)

```
Require: $\mathcal{D}_{\mathrm{train}} = \{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_{\mathsf{p}}\} \subset \mathcal{D}$, $0 < \varepsilon^{\mathsf{gr}} \ll 1$, $0 \ll \varepsilon^{\mathsf{pod}} < 1$, an $k_{\mathsf{max}} \in \mathbb{N}$;
Choose $\boldsymbol{\mu}^{(1)} \in \mathcal{D}_{\mathrm{train}}$;
$\mathrm{C}^h(\boldsymbol{\mu}^{(1)}) \leftarrow$ solve (4);
$\psi_1, \ldots, \psi_{\ell_1} \leftarrow$ call Algorithm 4.1 with inputs $\mathrm{C}^h(\boldsymbol{\mu}^{(1)})$ and $\varepsilon^{\mathsf{pod}}$;
$N \leftarrow \ell_1$, $\xi_i \leftarrow \psi_i$ $(1 \le i \le N)$ and $k \leftarrow 1$;
while $k < k_{\mathsf{max}}$ and $\max_{\boldsymbol{\mu}\in\mathcal{D}_{\mathrm{train}}} \Delta^N(\boldsymbol{\mu}) > \varepsilon^{\mathsf{gr}}$ do
    $\boldsymbol{\mu}^{(k+1)} \leftarrow \operatorname{argmax}_{\boldsymbol{\mu}\in\mathcal{D}_{\mathrm{train}}} \Delta^N(\boldsymbol{\mu})$;
    $\mathrm{C}^h(\boldsymbol{\mu}^{(k+1)}) \leftarrow$ solve (4);
    $\psi_1, \ldots, \psi_{\ell_{k+1}} \leftarrow$ call Algorithm 4.1 with inputs $\mathrm{C}^h(\boldsymbol{\mu}^{(k+1)}) - \mathcal{P}^\ell \mathrm{C}^h(\boldsymbol{\mu}^{(k+1)})$ and
    $\varepsilon^{\mathsf{pod}}$;
    $\xi_1, \ldots, \xi_{N+\ell_{k+1}} \leftarrow$ Algorithm 4.3 with $\xi_1, \ldots, \xi_N, \psi_1, \ldots, \psi_{\ell_{k+1}}$;
    $N \leftarrow N + \ell_{k+1}$ and $k \leftarrow k + 1$;
end while
return reduced-basis $\Xi = [\xi_1 \,|\, \ldots \,|\, \xi_N]$;
```

Remark 4.2: 1) In Algorithm 4.2 we may also use the error itself instead of the (computationally expensive) estimator $\Delta^N(\boldsymbol{\mu})$. Then, the method is called *strong* POD-greedy algorithm, in the other case *weak*.

2) The basis vectors are chosen orthonormal for stability reasons, cf. [26, Section 5.7]. However, in numerical realizations instabilities may occur. For that reason

the ξ's are orthonormalized the after each Greedy step by Algorithm 4.3; see e.g. [17]. We stop Algorithm 4.3 if the information of the new basis becomes too small. This is the case if

$$\|w_j\|_1 = \sum_{i=1}^{\mathcal{N}} \big|(w_j)_i\big| < \varepsilon^{\mathsf{GS}} \quad \text{with } \varepsilon^{\mathsf{GS}} = 10^{-14},$$

where w_j is the one determined in line 3 of Algorithm 4.3 and $(w_j)_i$ denotes its i-th component. Other stabilization techniques can be found in [27]. ◇

Algorithm 4.3 (Gram-Schmidt orthogonalization with weighted inner product)

```
Require: $v_1, \dots, v_I \in \mathbb{R}^{\mathcal{N}}$;
1: $w_1 \leftarrow v/\|v_1\|_{\mathrm{W}}$;
2: for $i = 2$ to $I$ do
3:     $w_i \leftarrow v_i - \sum_{j=1}^{i-1} \langle w_j, v_i \rangle_{\mathrm{W}}\, w_j$;
4:     $w_i \leftarrow w_i/\|w_i\|_{\mathrm{W}}$;
5: end for
6: return W-orthonormal basis $[w_1 | \dots | w_n] \in \mathbb{R}^{\mathcal{N} \times I}$;
```

5. Error analysis

In the present paper we improve the error estimator which was presented in [31, Theorem 6.5].

5.1. *Non-negative matrices and inverses*

Recall the following definition; see, e.g., [23, p. 54].

Definition 5.1: *A matrix* $\mathrm{A} = ((\mathrm{a}_{ij})) \in \mathbb{R}^{\mathcal{N} \times \mathcal{N}}$ *is called an* M-matrix *if* A *is invertible, its inverse* A^{-1} *possesses only non-negative coefficients and* $\mathrm{a}_{ij} \leq 0$ *for* $1 \leq i, j \leq \mathcal{N}$ *with* $i \neq j$.

A sufficient condition for a matrix to be an M-matrix is given in the next lemma [23, pp. 55-56]. For the definition of a strictly diagonally dominant matrix we refer to [23, p. 48].

Lemma 5.2: *Let* $\mathrm{A} = ((\mathrm{a}_{ij})) \in \mathbb{R}^{\mathcal{N} \times \mathcal{N}}$ *be strictly diagonally dominant and assume that* $\mathrm{a}_{ij} \leq 0$ *for* $1 \leq i, j \leq \mathcal{N}$ *with* $i \neq j$ *and that* $\mathrm{a}_{ii} > 0$ *for* $1 \leq i, \leq \mathcal{N}$. *Then,* A *is an* M-*matrix.*

We have introduced the tridiagonal matrix

$$\mathrm{L}^h_{\mathsf{im}}(\mu_1) = \begin{pmatrix} h+\frac{\mu_1 k}{h} & -\frac{\mu_1 k}{h} & & & \\ -\frac{\mu_1 k}{h} & h+\frac{2\mu_1 k}{h} & -\frac{\mu_1 k}{h} & & \\ & \ddots & \ddots & \ddots & \\ & & -\frac{\mu_1 k}{h} & h+\frac{2\mu_1 k}{h} & -\frac{\mu_1 k}{h} \\ & & & -\frac{\mu_1 k}{h} & h+\frac{\mu_1 k}{h} \end{pmatrix} \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}}$$

in Section 3. Since $\mu_1 \geq \mu_1^a > 0$ holds, the diagonal elements are positive and the minor diagonal elements are non-positive. Furthermore, $\mathrm{L}^h_{\mathsf{im}}(\mu_1)$ is strictly diagonally dominant. By Lemma 5.2 the matrix $\mathrm{L}^h_{\mathsf{im}}(\mu_1)$ is invertible for any $\boldsymbol{\mu} \in \mathcal{D}$. Moreover, it follows analogously that

$$\mathrm{L}^h_{\mathsf{im}}(\mu_1) + \eta \mathrm{D}^h \quad \text{with } \eta \geq 0 \text{ and } \mathrm{D}^h = \begin{pmatrix} 0 & & & \\ & \ddots & & \\ & & 0 & \\ & & & 1 \end{pmatrix} \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}} \tag{11}$$

is an M-matrix and therefore regular as well for all $\boldsymbol{\mu} \in \mathcal{D}$.

We have introduced $\|\cdot\|$ as a (vector) norm on $\mathbb{R}^{\mathcal{N}}$. The (associated) matrix norm is defined as

$$\|\mathrm{A}\| = \sup_{\|\mathrm{x}\|=1} \|\mathrm{Ax}\| = \sup_{\mathrm{x}\neq 0} \frac{\|\mathrm{Ax}\|}{\|\mathrm{x}\|} \quad \text{for } \mathrm{A} \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}}.$$

It follows that

$$\|\mathrm{Ax}\| \leq \|\mathrm{A}\|\|\mathrm{x}\| \quad \text{for all } \mathrm{A} \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}} \text{ and } \mathrm{x} \in \mathbb{R}^{\mathcal{N}}. \tag{12}$$

Let $\mathrm{A} \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}}$ be regular. Using (12) we infer that

$$\|\mathrm{x}\| = \|\mathrm{A}^{-1}\mathrm{Ax}\| \leq \|\mathrm{A}^{-1}\|\|\mathrm{Ax}\| \quad \text{for all } \mathrm{x} \in \mathbb{R}^{\mathcal{N}}$$

which gives

$$\|\mathrm{Ax}\| \geq \frac{\|\mathrm{x}\|}{\|\mathrm{A}^{-1}\|} \quad \text{for all } \mathrm{x} \in \mathbb{R}^{\mathcal{N}}. \tag{13}$$

The next result is known as *perturbation lemma*; see, e.g., [23, p. 45].

Lemma 5.3: *Let* A, B $\in \mathbb{R}^{\mathcal{N}\times\mathcal{N}}$ *be given. We suppose that* A *is invertible satisfying* $\|\mathrm{A}^{-1}\| \leq \beta$ *for a positive constant* β*. If* $\|\mathrm{A} - \mathrm{B}\| \leq \gamma$ *and* $\beta\gamma < 1$*, then* B *is also invertible and satisfies*

$$\|\mathrm{B}^{-1}\| \leq \frac{\beta}{1-\beta\gamma}.$$

5.2. *A-posteriori error analysis*

For every $\boldsymbol{\mu} \in \mathcal{D}$ and $j \in \{1, \dots, K\}$ the error between the j-th FV solution $\mathrm{c}_j^h(\boldsymbol{\mu})$ and the RB solution $\mathrm{c}_j^N(\boldsymbol{\mu})$ is defined by

$$\mathrm{e}_j^N(\boldsymbol{\mu}) = \mathrm{c}_j^h(\boldsymbol{\mu}) - \mathrm{c}_j^N(\boldsymbol{\mu}) \in \mathbb{R}^{\mathcal{N}} \quad \text{for } j = 1, \dots, K$$

and the residuals are given by

$$\mathrm{r}_j^N(\boldsymbol{\mu}) = \mathrm{F}_j^h\big(\mathrm{c}_j^N(\boldsymbol{\mu}); \boldsymbol{\mu}\big) \in \mathbb{R}^{\mathcal{N}} \quad \text{for } j = 2, \dots, K. \tag{14}$$

Since we solve (4) with a globalized Newton method, we can not guarantee that our FV solution $\mathrm{c}_j^h(\boldsymbol{\mu})$ fulfils $\mathrm{F}_j^h(\mathrm{c}_j^h(\boldsymbol{\mu}); \boldsymbol{\mu}) = 0$. Instead, we have

$$\big\|\mathrm{F}_j^h(\mathrm{c}_j^h(\boldsymbol{\mu}); \boldsymbol{\mu})\big\| \leq \varepsilon^{\mathsf{new}} \quad \text{for } j \in \{2, \dots, K\} \text{ and } \boldsymbol{\mu} \in \mathcal{D} \tag{15}$$

with a user-specified Newton tolerance $0 < \varepsilon^{\mathsf{new}} \ll 1$. Analogously, for RB solution $\mathrm{c}_j^N(\boldsymbol{\mu})$ the equation (6) is in general not valid, but it fulfils the inequality

$$\big\|\mathrm{F}_j^N(\mathrm{c}_j^N(\boldsymbol{\mu}); \boldsymbol{\mu})\big\| \leq \varepsilon^{\mathsf{new}} \quad \text{for } j \in \{2, \dots, K\} \text{ and } \boldsymbol{\mu} \in \mathcal{D}, \tag{16}$$

where we utilize the same Newton tolerance $\varepsilon^{\mathsf{new}}$ as for the FV system. Let us define the matrix

$$\mathrm{G}^h(\boldsymbol{\mu}) = \mathrm{L}_{\mathsf{im}}^h(\mu_1) + \eta \mathrm{D}^h \quad \text{with } \mathrm{D}^h \text{ as in (11) and } \eta = \frac{\mu_2 k}{\sqrt{c_{\mathsf{min}}} + \sqrt{\mathrm{c}_{j,\mathcal{N}}^N(\boldsymbol{\mu})}}, \tag{17}$$

where we have introduced the matrix D^h in (11). If Assumption 4.1 and $\boldsymbol{\mu} = (\mu_1, \mu_2) \in \mathcal{D}$ are satisfied, $\eta > 0$ follows. Therefore, $\mathrm{G}^h(\boldsymbol{\mu})$ is an M-matrix and invertible.

Theorem 5.4 (A-posteriori error estimate)**:** *Let $\boldsymbol{\mu} \in \mathcal{D}$, Assumptions* 3.1 *and* 4.1 *hold. Suppose that $\{\mathrm{c}_j^h(\boldsymbol{\mu})\}_{j=1}^K$ and $\{\mathrm{c}_j^N(\boldsymbol{\mu})\}_{j=1}^K$ denote the (inexact) FV and the RB solutions satisfying* (15) *and* (16)*, respectively, with the Newton tolerance $\varepsilon^{\mathsf{new}} > 0$. Let the* M*-matrix $\mathrm{G}^h(\boldsymbol{\mu})$ be given by* (17)*. If the step size satisfies*

$$k \leq \frac{\sqrt{c_{\mathsf{min}}} + \sqrt{\mathrm{c}_{j,\mathcal{N}}^N(\boldsymbol{\mu})}}{2\mu_2^b \,\|\mathrm{G}^h(\boldsymbol{\mu})^{-1}\| \|\mathrm{D}^h\|} \quad \textit{for } j \in \{2, \dots, K\}, \tag{18}$$

then

$$\big\|\mathrm{e}_j^N(\boldsymbol{\mu})\big\| \leq 2\,\|\mathrm{G}^h(\boldsymbol{\mu})^{-1}\| \Big(\varepsilon^{\mathsf{new}} + \|\mathrm{r}_j^N(\boldsymbol{\mu})\| + \|\mathrm{L}_{\mathsf{ex}}^h\| \|\mathrm{e}_{j-1}^N(\boldsymbol{\mu})\|\Big) \tag{19}$$

for $j \in \{2, \dots, K\}$ and $\boldsymbol{\mu} \in \mathcal{D}$, where the residual $\mathrm{r}_j^N(\boldsymbol{\mu})$ has been introduced in (14)*.*

Proof. Let $\boldsymbol{\mu} \in \mathcal{D}$ and $j \in \{1, \dots, K\}$. Using (4), (7) and the equation

$$\sqrt{\mathrm{c}^h_{j,\mathcal{N}}(\boldsymbol{\mu})} - \sqrt{\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})} = \frac{\mathrm{c}^h_{j,\mathcal{N}}(\boldsymbol{\mu}) - \mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}{\sqrt{\mathrm{c}^h_{j,\mathcal{N}}(\boldsymbol{\mu})} + \sqrt{\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}} = \frac{\mathrm{e}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}{\sqrt{\mathrm{c}^h_{j,\mathcal{N}}(\boldsymbol{\mu})} + \sqrt{\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}}$$

we obtain

$$\begin{aligned} &\mathrm{F}^h_j\big(\mathrm{c}^h_j(\boldsymbol{\mu});\boldsymbol{\mu}\big) - \mathrm{F}^h_j\big(\mathrm{c}^N_j(\boldsymbol{\mu});\boldsymbol{\mu}\big) \\ &= \mathrm{L}^h_{\mathsf{im}}(\mu_1)\mathrm{e}^N_j(\boldsymbol{\mu}) - \mathrm{L}^h_{\mathsf{ex}}\mathrm{e}^N_{j-1}(\boldsymbol{\mu}) + \mu_2 k \left(\sqrt{\mathrm{c}^h_{j,\mathcal{N}}(\boldsymbol{\mu})} - \sqrt{\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}\right)\begin{pmatrix}0\\ \vdots\\ 0\\ 1\end{pmatrix} \\ &= \mathrm{L}^h_{\mathsf{im}}(\mu_1)\mathrm{e}^N_j(\boldsymbol{\mu}) - \mathrm{L}^h_{\mathsf{ex}}\mathrm{e}^N_{j-1}(\boldsymbol{\mu}) + \frac{\mu_2 k \mathrm{e}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}{\sqrt{\mathrm{c}^h_{j,\mathcal{N}}(\boldsymbol{\mu})} + \sqrt{\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}}\begin{pmatrix}0\\ \vdots\\ 0\\ 1\end{pmatrix}. \end{aligned} \tag{20}$$

Recall that $\mu_2 \geq \mu_2^a > 0$ and $\mathrm{c}^h_{j,\mathcal{N}}(\boldsymbol{\mu})$, $\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu}) \geq c_{\mathsf{min}} > 0$ hold for all $j \in \{2, \dots, K\}$ and all $\boldsymbol{\mu} \in \mathcal{D}$. Therefore,

$$\eta^h_j(\mu_2) := \frac{\mu_2}{\sqrt{\mathrm{c}^h_{j,\mathcal{N}}(\boldsymbol{\mu})} + \sqrt{\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}}$$

is positive and the tridiagonal matrix $\mathrm{L}^h_{\mathsf{im}}(\mu_1) + k\eta^h_j(\mu_2)\mathrm{D}^h$ is an M-matrix (compare (11)). From (20) and (13), it follows that

$$\begin{aligned} &\big\|\mathrm{F}^h_j\big(\mathrm{c}^h_j(\boldsymbol{\mu});\boldsymbol{\mu}\big) - \mathrm{F}^h_j\big(\mathrm{c}^N_j(\boldsymbol{\mu});\boldsymbol{\mu}\big) + \mathrm{L}^h_{\mathsf{ex}}\mathrm{e}^N_{j-1}(\boldsymbol{\mu})\big\| \\ &= \left\|\big(\mathrm{L}^h_{\mathsf{im}}(\mu_1) + k\eta^h_j(\mu_2)\mathrm{D}^h\big)\mathrm{e}^N_j(\boldsymbol{\mu})\right\| \geq \big\|\mathrm{e}^N_j(\boldsymbol{\mu})\big\| \Big/ \left\|\big(\mathrm{L}^h_{\mathsf{im}}(\mu_1) + k\eta^h_j(\mu_2)\mathrm{D}^h\big)^{-1}\right\| \end{aligned} \tag{21}$$

for all $j \in \{2, \dots, K\}$ and all $\boldsymbol{\mu} \in \mathcal{D}$. Thus, we infer from (21), (12), (14) and the triangle inequality that

$$\begin{aligned} \big\|\mathrm{e}^N_j(\boldsymbol{\mu})\big\| &\leq \left\|\big(\mathrm{L}^h_{\mathsf{im}}(\mu_1) + k\eta^h_j(\mu_2)\mathrm{D}^h\big)^{-1}\right\| \\ &\quad \cdot \big\|\mathrm{F}^h_j\big(\mathrm{c}^h_j(\boldsymbol{\mu});\boldsymbol{\mu}\big) - \mathrm{F}^h_j\big(\mathrm{c}^N_j(\boldsymbol{\mu});\boldsymbol{\mu}\big) + \mathrm{L}^h_{\mathsf{ex}}\mathrm{e}^N_{j-1}(\boldsymbol{\mu})\big\| \\ &\leq \left\|\big(\mathrm{L}^h_{\mathsf{im}}(\mu_1) + k\eta^h_j(\mu_2)\mathrm{D}^h\big)^{-1}\right\| \left(\varepsilon^{\mathsf{new}} + \big\|\mathrm{r}^N_j(\boldsymbol{\mu})\big\| + \big\|\mathrm{L}^h_{\mathsf{ex}}\big\|\big\|\mathrm{e}^N_{j-1}(\boldsymbol{\mu})\big\|\right). \end{aligned} \tag{22}$$

To derive an a-posteriori error estimate, we have to avoid the FV term $\mathrm{c}^h_{j,\mathcal{N}}(\boldsymbol{\mu})$ in (22). Due to Assumption 3.1 we have

$$\begin{aligned} 0 &\leq \frac{\sqrt{\mathrm{c}^h_{j,\mathcal{N}}(\boldsymbol{\mu})} - \sqrt{c_{\mathsf{min}}}}{\big(\sqrt{c_{\mathsf{min}}} + \sqrt{\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}\big)\big(\sqrt{\mathrm{c}^h_{j,\mathcal{N}}(\boldsymbol{\mu})} + \sqrt{\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}\big)} \\ &= \frac{1}{\sqrt{c_{\mathsf{min}}} + \sqrt{\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}} - \frac{1}{\sqrt{\mathrm{c}^h_{j,\mathcal{N}}(\boldsymbol{\mu})} + \sqrt{\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}} \leq \frac{1}{\sqrt{c_{\mathsf{min}}} + \sqrt{\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}} \end{aligned}$$

for all $j \in \{2, \dots, K\}$ and all $\boldsymbol{\mu} \in \mathcal{D}$. Therefore, if k satisfies (18), it follows that

$$\begin{aligned}
&\left\| \mathrm{G}^h(\boldsymbol{\mu}) - \left(\mathrm{L}^h_{\mathsf{im}}(\mu_1) + k\eta_j(\mu_2)\mathrm{D}^h\right)\right\| \\
&\quad = \mu_2 k \left(\frac{1}{\sqrt{c_{\mathsf{min}}} + \sqrt{\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}} - \frac{1}{\sqrt{\mathrm{c}^h_{j,\mathcal{N}}(\boldsymbol{\mu})} + \sqrt{\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}} \right) \left\| \mathrm{D}^h \right\| \\
&\quad \le \frac{\mu_2 k \left\| \mathrm{D}^h \right\|}{\sqrt{c_{\mathsf{min}}} + \sqrt{\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}} \le \frac{\mu_2}{\mu_2^b} \frac{1}{2\left\| \mathrm{G}^h(\boldsymbol{\mu})^{-1} \right\|} \le \frac{1}{2\left\| \mathrm{G}^h(\boldsymbol{\mu})^{-1} \right\|}
\end{aligned}$$

for $j \in \{2, \dots, K\}$ and $\boldsymbol{\mu} \in \mathcal{D}$. Hence, we conclude from

$$\left\| \mathrm{G}^h(\boldsymbol{\mu})^{-1} \right\| \left\| \mathrm{G}^h(\boldsymbol{\mu}) - \left(\mathrm{L}^h_{\mathsf{im}}(\mu_1) + k\eta_j(\mu_2)\mathrm{D}^h\right)\right\| \le \frac{1}{2}$$

and Lemma 5.3 that

$$\left\| \left(\mathrm{L}^h_{\mathsf{im}}(\mu_1) + k\eta_j(\mu_2)\mathrm{D}^h\right)^{-1} \right\| \le 2 \left\| \mathrm{G}^h(\boldsymbol{\mu})^{-1} \right\|$$

for all $j \in \{2, \dots, K\}$ and all $\boldsymbol{\mu} \in \mathcal{D}$. □

Remark 5.5: Estimate (19) depends on the constant c_{min}, which is usually unknown. Due to Theorem 5.4 we have $\mathrm{c}^N_j(\boldsymbol{\mu}) \to \mathrm{c}^h_j(\boldsymbol{\mu})$ as $N \to \mathcal{N}$ for any $j \in \{1, \dots, K\}$. Hence, we assume that

$$\mathrm{c}^h_{j,\mathcal{N}}(\boldsymbol{\mu}) \ge \frac{\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}{4} > 0 \quad \text{for } N \text{ sufficiently large.}$$

Then, we replace c_{min} by $\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})/4$ and proceed as in the proof of Theorem 5.4. From

$$\sqrt{\mathrm{c}^h_{j,\mathcal{N}}(\boldsymbol{\mu})} + \sqrt{\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})} \ge \frac{3}{2}\sqrt{\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}$$

we derive the following estimate for sufficiently large N: The matrix

$$\widehat{\mathrm{G}}^h(\boldsymbol{\mu}) = \mathrm{L}^h_{\mathsf{im}}(\mu_1) + \hat{\eta}\mathrm{D}^h \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}} \quad \text{with } \mathrm{D}^h \text{ as in (11) and } \hat{\eta} = \frac{2\mu_2 k}{3\sqrt{\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}}$$

is an M-matrix. If the step size satisfies

$$k \le \frac{3\sqrt{\mathrm{c}^N_{j,\mathcal{N}}(\boldsymbol{\mu})}}{4\mu_2^b \left\| \widehat{\mathrm{G}}^h(\boldsymbol{\mu})^{-1} \right\| \left\| \mathrm{D}^h \right\|} \quad \text{for } j \in \{2, \dots, K\},$$

then

$$\left\| \mathrm{e}^N_j(\boldsymbol{\mu}) \right\| \le 2 \left\| \widehat{\mathrm{G}}^h(\boldsymbol{\mu})^{-1} \right\| \left(\varepsilon^{\mathsf{new}} + \left\| \mathrm{r}^N_j(\boldsymbol{\mu}) \right\| + \left\| \mathrm{L}^h_{\mathsf{ex}} \right\| \left\| \mathrm{e}^N_{j-1}(\boldsymbol{\mu}) \right\| \right) \tag{23}$$

for all $j \in \{2, \dots, K\}$ and all $\boldsymbol{\mu} \in \mathcal{D}$. ◇

Remark 5.6: In Section 7.1 we estimate the error in the maximum norm. Hence, we have to compute $\|(\mathrm{G}(\boldsymbol{\mu}))^{-1}\|_\infty$. For tridiagonal matrices like in the present case, there exist efficient algorithms. We use one of Hargreaves, cf. [9, Algorithm]. In our numerical example the computation of the norm $\|(\mathrm{G}(\boldsymbol{\mu}))^{-1}\|_\infty$ for a $10^2 \times 10^2$ matrix lasts less than 0.01 seconds, for a $10^3 \times 10^3$ matrix less than 0.1 seconds. ◇

6. Parameter estimation problem

We suppose that Assumption 2.2 holds. The nonlinear model (1) contains a parameter $\boldsymbol{\mu} \in \mathcal{D}$, which has to be identified in order to calibrate the model. However, measurements for the concentration $c(\boldsymbol{\mu})$ are not directly available. Instead, the *state of charge*

$$\mathsf{SoC}(t;\boldsymbol{\mu}) = \frac{1}{c_{\mathsf{max}}} \int_\Omega c(t,x;\boldsymbol{\mu})\,\mathrm{d}x \quad \text{for } t \in [0,T] \tag{24}$$

can be measured, where $c(\cdot,\cdot;\boldsymbol{\mu}) \in \mathcal{C} = W(0,T) \cap L^\infty(Q)$ is the unique weak solution to (1) for given $\boldsymbol{\mu} \in \mathcal{D}$. Suppose that $\mathsf{SoC}^d \in L^\infty(0,T)$ is a given desired state of charge profile. Then, we consider the least squares objective

$$\hat{J} : \mathcal{D} \to \mathbb{R}, \quad \hat{J}(\boldsymbol{\mu}) = \frac{1}{2} \int_0^T \big|\mathsf{SoC}(t;\boldsymbol{\mu}) - \mathsf{SoC}^d(t)\big|^2 \,\mathrm{d}t \text{ for } \boldsymbol{\mu} \in \mathcal{D},$$

where $\mathsf{SoC}(t,\boldsymbol{\mu})$ is given by (24). Now, the parameter estimation can be expressed as

$$\min \hat{J}(\boldsymbol{\mu}) \text{ subject to } \boldsymbol{\mu} \in \mathcal{D}. \tag{25}$$

Problem (25) is a PDE-constrained, non-convex optimization problem so that many local optimal solutions my occur. Moreover, it is not a-priorily clear that (25) admits an optimal solution at all. We refer the reader to [14, 28] for more details on this subject.

Here, we follow the approach “first discretize then optimize”, cf. e.g. [14]. Suppose that Assumption 3.1 holds. We utilize the already defined temporal grid $t_j = (j-1)k$, $1 \le j \le K$ and replace the temporal integral by a trapezoidal rule:

$$\begin{aligned}\hat{J}(\boldsymbol{\mu}) \approx \frac{k}{4}\big|\mathsf{SoC}(t_1;\boldsymbol{\mu}) - \mathsf{SoC}^d(t_1)\big|^2 + \frac{k}{2}\sum_{j=2}^{K-1}\big|\mathsf{SoC}(t_j;\boldsymbol{\mu}) - \mathsf{SoC}^d(t_j)\big|^2 \\ + \frac{k}{4}\big|\mathsf{SoC}(t_K;\boldsymbol{\mu}) - \mathsf{SoC}^d(t_K)\big|^2 \quad \text{for } \boldsymbol{\mu} \in \mathcal{D}.\end{aligned}$$

Next, we replace the state of charge by a FV approximation and evaluate the spatial integral by a midpoint rule:

$$\mathsf{SoC}(t_j;\boldsymbol{\mu}) \approx \mathsf{SoC}^h_j(\boldsymbol{\mu}) = \frac{h}{\mathsf{c}_{\mathsf{max}}} \sum_{i=1}^{\mathcal{N}} \mathrm{c}^h_{j,i}(\boldsymbol{\mu}) \quad \text{for } j \in \{1,\ldots,K\},$$

where $\mathrm{c}^h_{j,i}(\boldsymbol{\mu})$ denotes the unique FV approximation for the concentration at

$(t_j, x_i) \in \overline{Q}$ and for $\boldsymbol{\mu} \in \mathcal{D}$. Finally, we define the FV reduced objective as

$$\begin{aligned}\hat{J}_k^h(\boldsymbol{\mu}) = \frac{k}{4}\left|\mathsf{SoC}_1^h(\boldsymbol{\mu}) - \mathsf{SoC}^d(t_1)\right|^2 + \frac{k}{2}\sum_{j=2}^{K-1}\left|\mathsf{SoC}_j^h(\boldsymbol{\mu}) - \mathsf{SoC}^d(t_j)\right|^2 \\ + \frac{k}{4}\left|\mathsf{SoC}_K^h(\boldsymbol{\mu}) - \mathsf{SoC}^d(t_K)\right|^2.\end{aligned}$$

Instead of (25) we consider the FV-based parameter estimation problem

$$\min \hat{J}_k^h(\boldsymbol{\mu}) \text{ subject to } \boldsymbol{\mu} \in \mathcal{D}. \tag{26}$$

Problem (26) depends on the size of $\mathcal{N}$ which may be large. Therefore, we are interested to utilize the RB method introduced in Section 3 in order to solve (26) approximately, but fast. Let us assume that Assumption 4.1 is valid. We replace the FV approximation $\mathsf{SoC}_j^h(\boldsymbol{\mu})$ of the state of charge by the following RB discretization:

$$\mathsf{SoC}_j^N(\boldsymbol{\mu}) = \frac{h}{\mathsf{c}_{\mathsf{max}}}\sum_{i=1}^{\mathcal{N}} \mathrm{c}_{j,i}^N(\boldsymbol{\mu}) \quad \text{for } j \in \{1, \ldots, K\},$$

where $\mathrm{c}_{j,i}^N(\boldsymbol{\mu})$ denotes the unique RB approximation for the concentration at (t_j, x_i) and for $\boldsymbol{\mu} \in \mathcal{D}$. Then, the RB-based parameter identification problem is expressed as

$$\min \hat{J}_k^N(\boldsymbol{\mu}) \text{ subject to } \boldsymbol{\mu} \in \mathcal{D} \tag{27}$$

with

$$\begin{aligned}\hat{J}_k^N(\boldsymbol{\mu}) = \frac{k}{4}\left|\mathsf{SoC}_1^N(\boldsymbol{\mu}) - \mathsf{SoC}^d(t_1)\right|^2 + \frac{k}{2}\sum_{j=2}^{K-1}\left|\mathsf{SoC}_j^N(\boldsymbol{\mu}) - \mathsf{SoC}^d(t_j)\right|^2 \\ + \frac{k}{4}\left|\mathsf{SoC}_K^N(\boldsymbol{\mu}) - \mathsf{SoC}^d(t_K)\right|^2.\end{aligned}$$

In Section 7.2 we will present numerical experiment for the parameter estimation problem.

7. Numerical experiments

In this section we first set an RB model which we use for the parameter estimation with respect to a desired SoC. All computations are done on a laptop: Linux Mint 17 Qiana, 64-bit; Intel(R) Core(TM) i5-4200U CPU 1.60GHz; 8 GB RAM; Matlab 8.1.0.604 (R2013a).

7.1. *Setting the RB model*

To check the suitability of our developed error estimators in Section 4 we do four different greedy runs. In all runs we consider the (estimated) error in the maximum norm and add one POD-mode in every greedy step. Our considered parameter set is

$\mathcal{N}$	300
h	0.03
k	0.1
$c_\circ$	55
$\mathrm{c}_{\max}$	60
K	20

Table 1. Parameters for the RB model.

	Run 1	Run 2	Run 3	Run 4
$t_{\text{off}}\,[s]$	$4.53\cdot 10^{1}$	$1.98\cdot 10^{2}$	$1.86\cdot 10^{2}$	$3.25\cdot 10^{1}$
$t_{\text{FVM}}\,[s]$	1.16	1.16	1.16	1.16
$t_{\text{RBM}}\,[s]$	$5.52\cdot 10^{-3}$	$5.36\cdot 10^{-3}$	$5.49\cdot 10^{-3}$	$5.36\cdot 10^{-3}$
speed-up	210	216	211	216
N	14	13	13	13
est.	$1.91\cdot 10^{-7}$	$8.66\cdot 10^{-5}$	$8.66\cdot 10^{-5}$	$4.50\cdot 10^{-7}$
err.	$1.91\cdot 10^{-7}$	$8.85\cdot 10-7$	$8.85\cdot 10^{-7}$	$8.85\cdot 10^{-7}$

Table 2. Results for the four different runs: computational offline time, computational time for the computation of a FV solution, computational time for the computation of a RB solution, speed-up, number of basis vectors, value of the estimator after the last POD-greedy step and value of the error after the last POD-greedy step.

$\mathcal{D} = [0.05, 5] \times [10^{-3}, 10^{-1}]$. The interval $[0.05, 5]$ as well as the interval $[10^{-3}, 10^{-1}]$ we discretize equidistantly with five nodes including the boundaries. Hence our training set $\mathcal{D}_{\text{train}}$ has the cardinal number 25. The remaining parameters are listed in Table 1. The accuracy of the damped Newton method is $\varepsilon^{\mathsf{new}} = 10^{-10}$ for the FV as well as for the RB solution. The greedy algorithm stops if the error is smaller than $\varepsilon^{\mathsf{gr}} = 10^{-6}$ or if the estimated error is smaller than $\varepsilon^{\mathsf{gr}} = 10^{-4}$ because we expect for this problem that the error estimators overestimate the error about two scales, cf. [31].

Run 1 Strong greedy: for the estimation of the error between the RB and FV solution the error itself is used.

Run 2 Weak greedy: for the estimation of the error the estimator of Theorem 5.4 (error est. 1) is used.

Run 3 Weak greedy: the error is estimated by the error of Remark (23) (error est 2).

Run 4 Weak greedy: the error is estimated by the residual. We stop the greedy algorithm as soon as the residual gets smaller than 10^{-6}.

In Figure 1 we plot the decay of the error and its estimators for the four different runs. We can hardly see a difference between the plots. As expected the two estimators overestimate the error about two scales. The values for the estimator 1 (19) and 2 (23) seem to agree. The computational time for the FV solutions to the complete training set is $1.74 \cdot 10^{1}\,s$. The different computational times, number of basis vectors, the value for the estimated error and error itself is listed in Table 2.

7.2. *Parameter estimation*

We use the RB model which we generated in the second run of the previous subsection for our optimization process. The results we compare to the results which we gain by using the FV model for the optimization process. We get the desired

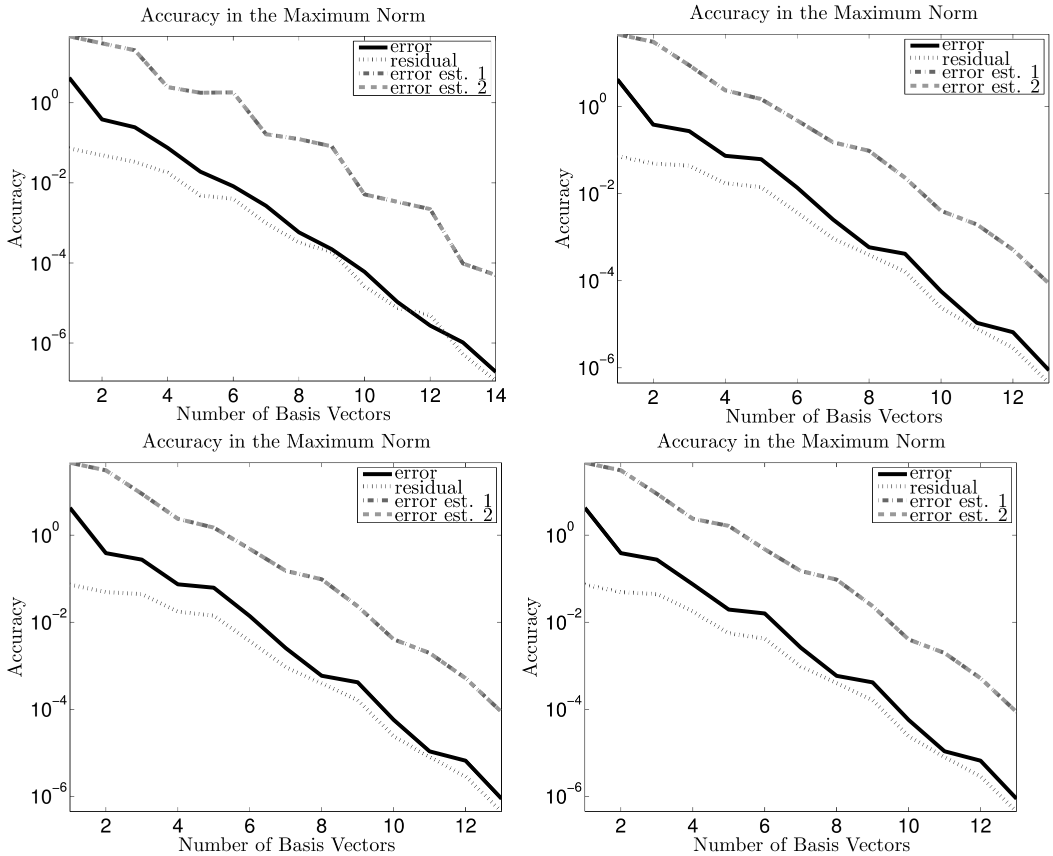

Figure 1. Decay of the error and its estimates for the four runs: Run 1 (up left), Run 2 (up right), Run 3 (bottom left) and Run 4 (bottom right).

SoC by inserting $\boldsymbol{\mu}^d$ in our FV model. We use the Matlab routine `fmincon` and the sequential quadratic programming (SQP), cf. e.g. [22]. We do not set a user defined gradient but use the Matlab internal one. The remaining settings for `fmincon` we set the maximum number of iterations `MaxIter` = $1,000$, the tolerance for the function value for the termination is set `TolFun` = 10^{-10}, termination tolerance for the step size `TolX` = 10^{-15} and the maximum of function evaluations `maxFunEvals` = $1,000$. The lower bound is given by $\boldsymbol{\mu}^a = (0.05, 0.001)$ and the upper bound is given by $\boldsymbol{\mu}^b = (5, 0.1)$. We do two numerical tests. In the first one we require comparative small values for μ_1 and μ_2 and the initial values are comparative big. In the second test we change the roles. The results are listed in Tables 3 and 4. The optimization results using the FV- or the RB model are the same. Hence our RB-model is suitable for the parameter estimation procedure. The speed-up factor is in this example is around 8.

The numerical results lead us to the assumption that the diffusion coefficient μ_1 has a negligible impact on the SoC for the considered parameter set range. The parameter μ_2 has an bigger impact. To confirm our assumption we plot the SoC in dependency on μ_1 and μ_2, cf. Figure 2.

	FVM	ROM
$\boldsymbol{\mu}^d$	$(0.1, 0.005)$	
$\boldsymbol{\mu}^{\text{init}}$	$(2.0, 0.09)$	
$\boldsymbol{\mu}^{\text{comp}}$	$(2.0000, 0.0050)$	$(2.0000, 0.0050)$
stopping criteria	local minimum found: first-order optimality measure `TolFun`	local minimum found: first-order optimality measure `TolFun`
$t_{\text{opt}}\,[s]$	21.01	2.51
iterations	10	10
funct. eval.	33	33
residual	$3.57 \cdot 10^{-17}$	$3.57 \cdot 10^{-17}$

Table 3. Parameter estimation - test 1: required parameter $\boldsymbol{\mu}^d$, initial parameter $\boldsymbol{\mu}^{\text{init}}$, estimated parameter $\boldsymbol{\mu}^{\text{comp}}$, stopping criteria, computational time for the optimization process, number of iterations, number of function evaluations, residual.

	FVM	ROM
$\boldsymbol{\mu}^d$	$(2.0, 0.09)$	
$\boldsymbol{\mu}^{\text{init}}$	$(0.1, 0.005)$	
$\boldsymbol{\mu}^{\text{comp}}$	$(0.1000, 0.0912)$	$(0.1000, 0.0912)$
stopping criteria	local minimum found: first-order optimality measure `TolFun`	local minimum found: first-order optimality measure `TolFun`
$t_{\text{opt}}\,[s]$	19.42	2.46
iterations	11	11
funct. eval.	36	36
residual	$5.46 \cdot 10^{-14}$	$5.46 \cdot 10^{-14}$

Table 4. Parameter estimation - test 2: required parameter $\boldsymbol{\mu}^d$, initial parameter $\boldsymbol{\mu}^{\text{init}}$, estimated parameter $\boldsymbol{\mu}^{\text{comp}}$, stopping criteria, computational time for the optimization process, number of iterations, number of function evaluations, residual.

Acknowledgement

The authors gratefully acknowledge partial financial support by the project *Reduced Basis Methods for Model Reduction and Sensitivity Analysis of Complex Partial Differential Equations with Applications to Lithium-Ion Batteries* funded by the company Adam Opel AG.

References

[1] L. Cai and R.E. White: *Reduction of model order based on proper orthogonal decomposition for lithium-ion battery simulations*, Journal of The Electrochemical Society 156 (2009), pp. A154–A161.

[2] P. AP14 and J. de Paula: *AP14' Physical Chemistry*, Oxford University Press, 10th edition (2014).

[3] T. Barth and M. Ohlberger: *Finite volume methods: Foundation and analysis.* In E. Stein, R. de Borst, and T.J.R. Hughes, editors, Encyclopedia of Computational Mechanics. John Wiley & Sons (2004).

[4] P. Deuflhard: *Newton Methods for Nonlinear Problems: Affine Invariance and Adaptive*

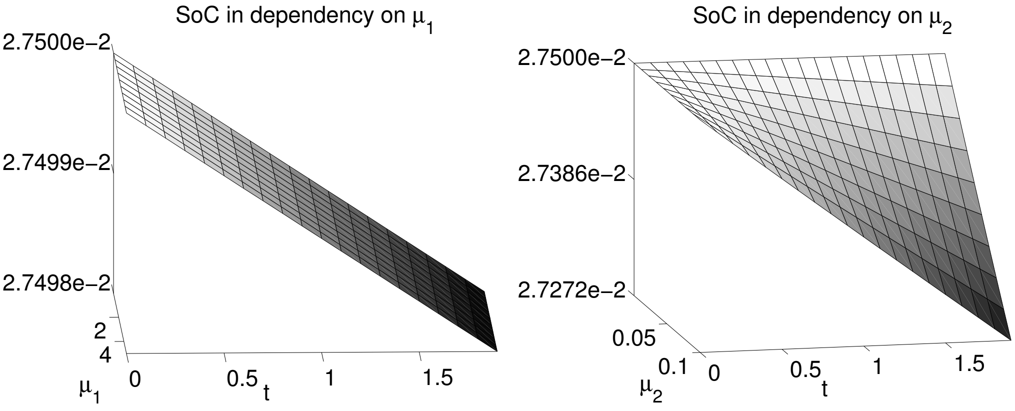

Figure 2. SoC in dependency on μ_1 (left) and μ_2 (right).

Algorithms, Springer, 2nd edition (2004).

[5] L.C. Evans: *Partial Differential Equations*, American Math. Society, Providence, Rhode Island, 2008.

[6] T. Fuller, M. Doyle and J. Newman: *Modeling of galvanostatic charge and discharge of the lithium ion battery/polymer/insertion cell*, Journal of the Chemical Society 140 (1993), pp. 1526–1533.

[7] M. A. Grepl, Y. Maday, N. C. Nguyen and A. T. Patera: *Efficient Reduced-Basis Treatment of Nonaffine Nonlinear Partial Differential Equations*, M2AN Mathematical Modelling and Numerical Analysis, 41(3) (2007), pp. 575–605.

[8] M. A. Grepl: *Reduced-Basis Approximation and A Posteriori Error Estimation for Parabolic Partial Differential Equations*, Ph.D. thesis, Massachusetts Institute of Technology (2005).

[9] G. I. Hargreaves: *Computing the Condition Number of Tridiagonal and Diagonal-Plus-Semiseparable Matrices in Linear Time*, Department of Mathematics, University of Manchester, technical report (2004).

[10] B. Haasdonk: *Convergence Rates of the POD-Greedy method*, M2AN Math. Model. Numer. Anal., 47 (2013), pp. 859–873.

[11] B. Haasdonk, M. Dihlmann and M. Ohlberger: *A Training Set and Multiple Basis Generation Approach for Parametrized Model Reduction Based on Adaptive Grids in Parameter Space*, Mathematical and Computer Modelling of Dynamical Systems, 17(4) (2011), pp. 423–442.

[12] J.S. Hesthaven, G. Rozza, and B. Stamm: *Certified Reduced Basis Methods for Parametrized Partial Differential Equations*, Springer, 2016.

[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) (2008), pp. 277–302.

[14] M. Hinze, R. Pinnau, M. Ulbrich and S. Ulbrich: *Optimization with PDE Constraints*, Mathematical Modelling: Theory and Applications, Springer Netherlands (2009).

[15] P. Holmes, J. L. Lumley, G. Berkooz and C. W. Rowley: *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*, 2nd edition, Cambridge Monographs on Mechanics, Cambridge University Press (2012).

[16] K. Kunisch and S. Volkwein: *Galerkin Proper Orthogonal Decomposition Methods for Parabolic Problems*, Numerische Mathematik, 90 (2001), pp. 117–148.

[17] S. Lang: *Linear Algebra*, 3rd edition, Springer (2004).

[18] O. Lass and S. Volkwein: *POD Galerkin schemes for nonlinear elliptic-parabolic systems*, SIAM Journal on Scientific Computing, 35 (2013), pp. A1271–A1298.

[19] O. Lass and S. Volkwein, *Parameter identification for nonlinear elliptic-parabolic systems with application in lithium-ion battery modeling*, Computational Optimization

and Applications 62 (2015), pp. 217-239.

[20] A. Latz, J. Zausch and O. Iliev: *Modeling of Species and Charge Transport in Li–Ion Batteries Based on Non-Equilibrium Thermodynamics*, technical report, Fraunhofer ITWM, Kaiserslautern, 190(2010).

[21] A. Latz and J. Zausch: *Thermodynamic Consistent Transport Theory of Li-Ion Batteries*, technical report, Fraunhofer ITWM, Kaiserslautern, 195 (2010).

[22] J. Nocedal and S. J. Wright: *Numerical Optimization*, 2nd edition, Springer (2006).

[23] J.M. Ortega and W.C. Rheinboldt: *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York (1970).

[24] A. T. Patera and G. Rozza: *Reduced Basis Approximation and A Posteriori Error Estimation for Parametrized Partial Differential Equations*, MIT (2006).

[25] J.-P. Raymond and H. Zidani, *Hamiltonian Pontryagin's principles for control problems governed by semilinear parabolic equations*, Applied Mathematics and Optimization 39 (1999), pp. 143-177.

[26] G. Rozza: *Shape Design by Optimal Flow Control and Reduced Basis Techniques*, Ph.D. thesis, EPFL, Lausanne (2005).

[27] L.N. Trefethen and D. Bau III: *Numerical Linear Algebra*, SIAM, Philadelphia, PA (1997).

[28] F. Tröltzsch, *Optimal Control of Partial Differential Equations: Theory, Methods and applications*, Graduate Studies in Mathematics, Vol. 112, American Mathematical Society (2010).

[29] S. Volkwein: *Model Reduction Using Proper Orthogonal Decomposition*, lecture notes, Universität Konstanz (2012).

[30] S. Volkwein and A. Wesche: *The reduced basis method applied to transport equations of a lithium-ion battery*, The International Journal for Computation and Mathematics in Electrical and Electronic Engineering 32 (2013), pp. 1760–1772.

[31] A. Wesche, *Reduced Basis Methods for Model Reduction and Sensitivity Analysis of Complex Partial Differential Equations with Applications to Lithium-Ion Batteries*, Ph.D. thesis., University of Konstanz, 2015.

[32] J. Xu, J. Wu and H. Zou: *On the well posedness of mathematical model for lithium-ion battery systems*, Methods and Applications of Analysis 13 (2006), pp. 275–298.