



# POD GALERKIN SCHEMES FOR NONLINEAR ELLIPTIC-PARABOLIC SYSTEMS

O. LASS AND S. VOLKWEIN

**ABSTRACT.** In this paper the authors study a nonlinear elliptic-parabolic system, which is motivated by mathematical models for lithium ion batteries. For the reliable and fast numerical solution of the system a reduced-order approach based on proper orthogonal decomposition (POD) is applied. The strategy is justified by an a-priori error estimate for the error between the solution to the coupled system and its POD approximation. The nonlinear coupling is realized by variants of the empirical interpolation introduced by Barrault et al. [3] and Chaturantabut et al. [4]. Numerical examples illustrate the efficiency of the proposed reduced-order modeling.

## 1. INTRODUCTION

Numerical simulation has emerged as an essential tool in industrial, engineering, and scientific research and development. The tremendous advances in the past decades have resulted in the availability of numerical simulation methods for increasingly complex systems of partial differential equations (PDEs). However, in many applications and fields, the complex system in question is essentially a combination of several types of PDEs each (possibly) involving different physics. This leads to multi component systems involving a variety of parameters, e.g. partly unknown or control parameters. The determination of unknown parameters (calibration) or the control, design, and optimization of coupled PDE systems thus requires repeated simulations of the multi component system with respect to different parameter values or control inputs. Therefore, fast and reliable *reduced* computational models are highly needed for such complex systems.

In the present paper we consider an elliptic-parabolic PDE system consisting of two elliptic and one parabolic equation. These coupled system can be viewed as a generalization of a mathematical model for lithium ion batteries; see, e.g., [10, 22, 25]. The parabolic equation describes the concentration of lithium ions and the two elliptic PDEs model the potential in the solid and liquid phase, respectively. These equations are coupled by a strong nonlinearity, which is a catenation of the square root, the hyperbolic sine and the logarithmic functions.

For the spatial approximation we apply a Galerkin scheme using proper orthogonal decomposition (POD); see e.g. [14, 24]. POD is based on projecting the

---

*Date:* September 9, 2011.

*2000 Mathematics Subject Classification.* 49K20, 65K10, 65K30.

*Key words and phrases.* Elliptic-parabolic systems, semilinear equations, Newton method, proper orthogonal decomposition, empirical interpolation, error estimates.

The authors gratefully acknowledges support by the German Science Fund *Numerical and Analytical Methods for Elliptic-Parabolic Systems Appearing in the Modeling of Lithium-Ion Batteries* (Excellence Initiative) and the support by the Zukunftsfonds Steiermark project *Prognosesicheres Batteriemodell für Elektro- und Hybridfahrzeuge*.

dynamical system onto subspaces of basis elements that express characteristics of the expected solution. This is in contrast to, e.g., finite element techniques, where the elements are not correlated to the physical properties of the system they approximate. The time integration is done by an implicit Euler method. For the numerical solution of the nonlinear discrete equations we apply a globalized Newton method with an appropriate computation of the initial guess.

Since the reduced-order model is a nonlinear system, the problem arises that the evaluation of the nonlinearity is of high complexity. To avoid this computational expensive evaluation the empirical interpolation method (EIM) was introduced [3]. This method is often used in the combination with the reduced basis approach; see, e.g., [12, 21]. The second approach we will investigate here is the discrete empirical interpolation method (DEIM) as introduced in [4, 6]. The basic concept of both methods is very similar. While the EIM implementation is based on a greedy algorithm the DEIM implementation is based on a POD approach combined with a greedy algorithm. It turns out that the offline computation for the EIM method is more expensive than for the DEIM method, whereas the online computation for the EIM method is cheaper than for the DEIM method (since a triangular system has to be solved instead of a full system).

In the present work an a-priori error estimate is presented for the POD Galerkin scheme (without the interpolation method for the nonlinearity). Whereas the numerical computations are done for a one-dimensional spatial grid the error analysis is carried out for the three-dimensional case. The resulting error bounds depend essentially on the number of POD basis functions and on an approximation property for the variable solving the parabolic equation. This result is of the same type as a-priori error estimates for POD Galerkin scheme derived for elliptic and parabolic PDEs; see [16, 17, 18, 23]. We also refer to [15], where the authors present an a-priori error estimate for a nonlinear heat equation coupled with an ordinary differential equation. Up to the authors knowledge there are no POD a-priori error analysis results available for systems with different PDE types.

The paper is organized in the following manner: In Section 2 the nonlinear elliptic-parabolic system is formulated. Section 3 is devoted to the POD Galerkin approximation. We discuss briefly the computation of the POD basis and the reduced-order model. Moreover, the different versions of the empirical interpolation are studied numerically. The a-priori error estimate is discussed in Section 4. The proofs are given in the appendix. In Section 5 numerical examples are presented. Finally, a conclusion is drawn in the last section.

## 2. THE NONLINEAR ELLIPTIC-PARABOLIC SYSTEM

In this section we formulate the nonlinear elliptic-parabolic system. Suppose that  $\Omega = (a, b) \subset \mathbb{R}$ ,  $a < b$ , is the spatial domain with boundary  $\Gamma = \{a, b\}$ . We set  $H = L^2(\Omega)$ ,  $V = H^1(\Omega)$  and

$$V_a = \{\varphi \in H^1(\Omega) \mid \varphi(a) = 0\}, \quad V_b = \{\varphi \in H^1(\Omega) \mid \varphi(b) = 0\}.$$

For the definition of Sobolev spaces we refer, e.g., to [1, 11]. For the terminal time  $T > 0$  let  $Q = (0, T) \times \Omega$  and  $\Sigma = (0, T) \times \Gamma$ . The space  $L^2(0, T; V)$  stands for the space of (equivalence classes) of measurable abstract functions  $\varphi : [0, T] \rightarrow V$ ,

which are square integrable, i.e.,

$$\int_0^T \|\varphi(t)\|_V^2 dt < \infty.$$

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

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

is a Hilbert space supplied with its common inner product; see [7].

For a given parameter  $\mu = (\mu_1, \mu_2) \in \mathcal{P}_{ad} \subset \mathbb{R}^2$  the triple  $(y, p, q) : Q \rightarrow \mathbb{R}$  satisfies the elliptic-parabolic systems

$$(2.1a) \quad y_t(t, x) - (c_1(x)y_x(t, x))_x + \mathcal{N}(y(t, x), p(t, x), q(t, x); \mu) = 0,$$

$$(2.1b) \quad -(c_2(x)p_x(t, x))_x + \mathcal{N}(y(t, x), p(t, x), q(t, x); \mu) = 0,$$

$$(2.1c) \quad -(c_3(x)q_x(t, x))_x - \mathcal{N}(y(t, x), p(t, x), q(t, x); \mu) = 0$$

for almost all (f.a.a.)  $(t, x)$  in  $Q$  together with the homogeneous mixed boundary conditions

$$(2.1d) \quad y_x(t, a) = y_x(t, b) = p(t, a) = p_x(t, b) = q_x(t, a) = q(t, b) = 0$$

f.a.a.  $t \in (0, T)$  and the initial condition

$$(2.1e) \quad y(0, x) = y_o(x)$$

f.a.a.  $x \in \Omega$ . The diffusion coefficients  $c_1, c_2, c_3$  are supposed to be piecewise constant and positive. Moreover,  $y_o : \Omega \rightarrow \mathbb{R}$  is a given bounded initial condition. The nonlinearity  $\mathcal{N} : \mathcal{Z}_{ad} \times \mathcal{P}_{ad} \rightarrow \mathbb{R}$  is given by

$$(2.2) \quad \mathcal{N}(z; \mu) = \mu_2 \sqrt{y} \sinh(\mu_1(q - p - \ln y))$$

for  $z = (y, p, q) \in \mathcal{Z}_{ad}$  and  $\mu = (\mu_1, \mu_2) \in \mathcal{P}_{ad}$ , where

$$\mathcal{Z}_{ad} = \{(y, p, q) \in \mathbb{R}^3 \mid y \geq y_{\min}\}, \quad \mathcal{P}_{ad} = \{(\mu_1, \mu_2) \in \mathbb{R}^2 \mid \mu_1 > 0 \text{ and } \mu_2 < 0\}$$

and  $y_{\min} > 0$  hold. With these choices (2.1) can be seen as a generalization of a mathematical model for lithium ion batteries; see, e.g., [10, 22, 25].  $\diamond$

**Remark 2.1.** 1) The positivity of the  $y$  component is needed to evaluate the terms  $\sqrt{y}$  and  $\ln y$  in the nonlinearity.

2) Notice that  $\mathcal{N}(\cdot; \mu) : \mathcal{Z}_{ad} \rightarrow \mathbb{R}$  is continuously differentiable for any  $\mu \in \mathcal{P}_{ad}$ . Furthermore,

$$\frac{\partial \mathcal{N}}{\partial p}(z; \mu) = -\mu_1 \mu_2 \sqrt{y} \cosh(\mu_1(q - p - \ln y)) = -\frac{\partial \mathcal{N}}{\partial q}(z; \mu) > 0$$

for  $z = (y, p, q) \in \mathcal{Z}_{ad}$  and  $\mu = (\mu_1, \mu_2) \in \mathcal{P}_{ad}$ . In the proof of the a-priori error estimate (see Section 4) we essentially utilize

$$(2.3) \quad \frac{\partial \mathcal{N}}{\partial p}(z; \mu) = -\frac{\partial \mathcal{N}}{\partial q}(z; \mu) > 0$$

for  $z = (y, p, q) \in \mathcal{Z}_{ad}$  and  $\mu = (\mu_1, \mu_2) \in \mathcal{P}_{ad}$ . Thus,  $\mathcal{P}_{ad}$  could be replaced by

$$\tilde{\mathcal{P}}_{ad} = \{(\mu_1, \mu_2) \in \mathbb{R}^2 \mid \mu_1 < 0 \text{ and } \mu_2 > 0\}$$

without any changes in the proofs.  $\diamond$

We introduce the function space

$$Z = (W(0, T; V) \times L^2(0, T; V_a) \times L^2(0, T; V_b)) \cap L^\infty(Q)^3$$

and call the triple  $(y, p, q)$  a *weak solution* to (2.1) if  $z = (y, p, q) \in Z$ ,  $y(0) = y_o$  in  $H$  and

$$(2.4a) \quad \int_{\Omega} y_t(t) \varphi + c_1 y_x(t) \varphi' + \mathcal{N}(y(t), p(t), q(t); \mu) \varphi \, dx = 0 \quad \text{for all } \varphi \in V,$$

$$(2.4b) \quad \int_{\Omega} c_2 p_x(t) \varphi' + \mathcal{N}(y(t), p(t), q(t); \mu) \varphi \, dx = 0 \quad \text{for all } \varphi \in V_a,$$

$$(2.4c) \quad \int_{\Omega} c_3 q_x(t) \varphi' - \mathcal{N}(y(t), p(t), q(t); \mu) \varphi \, dx = 0 \quad \text{for all } \varphi \in V_b.$$

It follows from [25] that (2.4) admits a unique weak solution. Therefore, the nonlinear solution operator  $\mathcal{S} : \mathcal{P}_{ad} \rightarrow Z$  is well-defined, where  $z = \mathcal{S}(\mu)$  is the weak solution to (2.1) for the parameter value  $\mu$ .

### 3. THE POD GALERKIN APPROXIMATION

In this section we introduce the POD method and develop the POD Galerkin scheme for (2.1). To evaluate the nonlinearity efficiently in our reduced-order approach we here use two techniques, the *empirical interpolation method* (EIM) and the *discrete empirical interpolation method* (DEIM).

**3.1. The POD method.** Suppose that  $z = (y, p, q) = \mathcal{S}(\mu)$  is the weak solution to (2.1) for a chosen parameter  $\mu \in \mathcal{P}_{ad}$ . We explain the computation of the POD basis for the first solution component  $y$ . Suppose that  $\mathcal{H}$  denotes either the space  $H$  or the space  $V$ . Notice that for  $p$  and  $q$  we choose  $H$  or  $V_a$  and  $V_b$ , respectively. The goal is to construct a low dimensional basis by solving the optimization problem

$$(3.1) \quad \begin{cases} \min_{\{\psi_i^y\}_{i=1}^{\ell^y}} \int_0^T \left\| y(t) - \sum_{i=1}^{\ell^y} \langle y(t), \psi_i^y \rangle_{\mathcal{H}} \psi_i^y \right\|_{\mathcal{H}}^2 \, dt \\ \text{subject to } \langle \psi_i^y, \psi_j^y \rangle_{\mathcal{H}} = \delta_{ij}. \end{cases}$$

To solve (3.1) let us define the integral operator  $\mathcal{R}$ . For  $y \in L^2(0, T; \mathcal{H})$  let  $\mathcal{R} : \mathcal{H} \rightarrow \mathcal{H}$  be given by

$$\mathcal{R}\psi^y = \int_0^T \langle y(t), \psi^y \rangle_{\mathcal{H}} y(t) \, dt \quad \text{for } \psi^y \in \mathcal{H}.$$

Clearly,  $\mathcal{R}$  is a linear bounded, nonnegative, self-adjoint operator which can be expressed as  $\mathcal{R} = \mathcal{Y}\mathcal{Y}^*$ , where  $\mathcal{Y} : L^2(0, T) \rightarrow \mathcal{H}$  is defined by

$$\mathcal{Y}v = \int_0^T v(t) y(t) \, dt \quad \text{for } v \in L^2(0, T),$$

and the adjoint  $\mathcal{Y}^* : \mathcal{H} \rightarrow L^2(0, T)$  is given by

$$(\mathcal{Y}^*\psi^y)(t) = \langle y(t), \psi^y \rangle_{\mathcal{H}} \quad \text{for } \psi^y \in \mathcal{H}.$$

We shall also utilize the operator  $\mathcal{K} : L^2(0, T) \rightarrow L^2(0, T)$  defined by

$$\mathcal{K} = \mathcal{Y}^*\mathcal{Y}$$

or explicitly

$$(\mathcal{K}v)(t) = \int_0^T \langle y(t), y(s) \rangle_{\mathcal{H}} v(s) \, ds \quad \text{for } v \in L^2(0, T).$$

For a proof of the following proposition we refer to [14] or [19, Proposition 2.1].

**Proposition 3.1.** *Except for possibly 0,  $\mathcal{K}$  and  $\mathcal{R}$  possess the same eigenvalues which are positive with identical multiplicities. Moreover,  $\psi^y$  is an eigenfunction of  $\mathcal{R}$  if and only if  $\mathcal{Y}^* \psi^y = \langle y(\cdot), \psi^y \rangle_{\mathcal{H}} \in L^2(0, T)$  is an eigenfunction of  $\mathcal{K}$ .*

The solution to (3.1) is given by the eigenfunctions corresponding to the  $\ell^y$  largest eigenvalues  $\lambda_i^y$  of the eigenvalue problem

$$(3.2a) \quad \mathcal{R}\psi_i^y = \int_0^T \langle y(t), \psi_i^y \rangle_{\mathcal{H}} y(t) \, dt = \lambda_i^y \psi_i^y \quad \text{for } i = 1, \dots, \ell^y,$$

$$(3.2b) \quad \langle \psi_i^y, \psi_j^y \rangle_{\mathcal{H}} = \delta_{ij} \quad \text{for } i, j = 1, \dots, \ell^y.$$

We shall utilize the POD basis  $\{\psi_i^y\}_{i=1}^{\ell^y}$  with respect to  $\mathcal{H} = H$  or  $\mathcal{H} = V$  satisfying  $\lambda_1^y \geq \dots \geq \lambda_{\ell^y}^y > 0$ ,

$$(3.3) \quad (\mathcal{K}v_i)(t) = \int_0^T \langle y(t), y(s) \rangle_{\mathcal{H}} v_i(s) \, ds = \lambda_i^y v_i(t) \quad \text{for } i = 1, \dots, \ell^y$$

and

$$\psi_i^y = \frac{1}{\sqrt{\lambda_i^y}} \int_0^T v_i(t) y(t) \, dt \in \mathcal{H} \quad \text{for } i = 1, \dots, \ell^y.$$

The POD-subspace for the variable  $y$  is then denoted by

$$V^{\ell^y} = \text{span} \{\psi_1^y, \dots, \psi_{\ell^y}^y\}.$$

Note that  $\psi_i^y \in V$  holds also for  $\mathcal{H} = H$ . This follows from (3.2a) using that  $y \in L^2(0, T; V)$ . In addition, we have the approximation error

$$\int_0^T \left\| y(t) - \sum_{i=1}^{\ell^y} \langle y(t), \psi_i^y \rangle_{\mathcal{H}} \psi_i^y \right\|_{\mathcal{H}}^2 dt = \sum_{i=\ell^y+1}^{\infty} \lambda_i^y.$$

An analogous result holds for the POD bases  $\{\psi_i^p\}_{i=1}^{\infty}$  and  $\{\psi_i^q\}_{i=1}^{\infty}$ .

For the solution components  $p$  and  $q$  we follow the same approach as for  $y$ . Hence we obtain POD bases  $\{\psi_i^p\}_{i=1}^{\ell^p}$  and  $\{\psi_i^q\}_{i=1}^{\ell^q}$ , respectively. We introduce super-indices  $y$ ,  $p$  and  $q$  for  $\ell$  and  $\psi$  to emphasize that the bases for the three solution components are computed independently and  $\ell$  may be different for each of the components.

**3.2. The reduced order model (ROM).** To obtain the POD Galerkin scheme for (2.1) we make the ansatz

$$y^\ell(t) = \sum_{i=1}^{\ell^y} \hat{y}_i(t) \psi_i^y, \quad p^\ell(t) = \sum_{i=1}^{\ell^p} \hat{p}_i(t) \psi_i^p, \quad q^\ell(t) = \sum_{i=1}^{\ell^q} \hat{q}_i(t) \psi_i^q,$$

and replace  $z = (y, p, q)$  by  $z^\ell = (y^\ell, p^\ell, q^\ell)$  in (2.4) and choose the POD basis functions as test functions. Hence, the POD Galerkin scheme is given by

$$(3.4a) \quad \int_{\Omega} y_i^\ell(t) \psi_i^y + c_1 y_x^\ell(t) (\psi_i^y)' + \mathcal{N}(z^\ell(t); \mu) \psi_i^y \, dx = 0, \quad 1 \leq i \leq \ell^y,$$

$$(3.4b) \quad \int_{\Omega} c_2 p_x^\ell(t) (\psi_i^p)' + \mathcal{N}(z^\ell(t); \mu) \psi_i^p \, dx = 0, \quad 1 \leq i \leq \ell^p,$$

$$(3.4c) \quad \int_{\Omega} c_3 q_x^\ell(t) (\psi_i^q)' - \mathcal{N}(z^\ell(t); \mu) \psi_i^q \, dx = 0, \quad 1 \leq i \leq \ell^q,$$

and

$$(3.4d) \quad \langle y^\ell(0), \psi_i^y \rangle_H = \langle y_0, \psi_i^y \rangle_H \quad \text{for } 1 \leq i \leq \ell^y.$$

**Assumption 1.** For any  $\mu \in \mathcal{P}_{ad}$  the reduced-order model (3.4) admits a unique weak solution denoted by  $z^\ell(\mu) = (y^\ell(\mu), p^\ell(\mu), q^\ell(\mu))$ .

**3.3. The discrete POD method.** Let us next introduce the discretized system. Here  $N_x$  and  $N_t$  will denote the number of discretization points in space and time, respectively. We utilize a finite element discretization and for this we define the discrete space

$$V^h = \text{span}\{\varphi_1, \dots, \varphi_{N_x}\} \subset V,$$

where the  $\varphi_i$ 's denote the  $N_x$  nodal basis functions. In the case of piecewise linear finite element discretization we get the typical hat functions. For the readers convenience we do not distinguish between the finite element spaces for  $V$ ,  $V_a$  and  $V_b$ . Any discretization scheme can be used, in particular there is no restriction to the finite element discretization. Using the standard Galerkin ansatz of the form

$$y^h(t, x) = \sum_{i=1}^{N_x} \bar{y}_i(t) \varphi_i(x), \quad p^h(t, x) = \sum_{i=1}^{N_x} \bar{p}_i(t) \varphi_i(x), \quad q^h(t, x) = \sum_{i=1}^{N_x} \bar{q}_i(t) \varphi_i(x),$$

and applying this to (2.4) we get the discrete nonlinear system

$$\begin{aligned} M \bar{y}_t(t) + S_{c_1} \bar{y}(t) + \bar{\mathcal{N}}(\bar{y}(t), \bar{p}(t), \bar{q}(t); \mu) &= 0, \\ S_{c_2} \bar{p}(t) + \bar{\mathcal{N}}(\bar{y}(t), \bar{p}(t), \bar{q}(t); \mu) &= 0, \\ S_{c_3} \bar{q}(t) - \bar{\mathcal{N}}(\bar{y}(t), \bar{p}(t), \bar{q}(t); \mu) &= 0, \end{aligned}$$

with

$$\begin{aligned} M_{ij} &= \int_{\Omega} \varphi_i(x) \varphi_j(x) \, dx, \quad (S_f)_{ij} = \int_{\Omega} f(x) \varphi_i'(x) \varphi_j'(x) \, dx, \\ (\bar{\mathcal{N}}(\bar{y}(t), \bar{p}(t), \bar{q}(t); \mu))_i &= \int_{\Omega} \mathcal{N}(y(t, x), p(t, x), q(t, x); \mu) \varphi_i(x) \, dx \end{aligned}$$

and

$$\bar{y}(t) = (\bar{y}_i(t))_{1 \leq i \leq N_x}, \quad \bar{p}(t) = (\bar{p}_i(t))_{1 \leq i \leq N_x}, \quad \bar{q}(t) = (\bar{q}_i(t))_{1 \leq i \leq N_x}.$$

This system can then be solved by using an appropriate method for the time discretization. In the numerical results presented we will use an implicit Euler method with equidistant time steps. To solve the arising nonlinear system a Newton method can be used. The convergence is ensured by applying a damping strategy; see [8, 9]. As snapshots for the POD computation we utilize the solutions to the nonlinear

system for the  $N_t$  discrete time instance. For that purpose we reformulate (3.1) in the discrete form:

$$(3.5) \quad \begin{cases} \min_{\{\psi_i^y\}_{i=1}^{\ell^y}} \sum_{j=0}^{N_t} \left\| y(t_j) - \sum_{i=1}^{\ell^y} \langle y(t_j), \psi_i^y \rangle_{\mathcal{H}} \psi_i^y \right\|_{\mathcal{H}}^2 \\ \text{subject to } \langle \psi_i^y, \psi_j^y \rangle_{\mathcal{H}} = \delta_{ij} \text{ for } 1 \leq i, j \leq \ell. \end{cases}$$

**Remark 3.2.** In contrast to [18] we do not include temporal weighting coefficients  $\alpha_j > 0$  in the sum. Therefore, our eigenvalues are scaled by approximately  $1/\Delta t$ . However, this is not a problem in the present work, since we do not study the asymptotic convergence for  $\Delta t \rightarrow 0$ . Moreover, our choice of the number of POD bases functions is based on the decay of the normalized eigenvalues  $\lambda_i / \sum_{j=1}^{N_t} \lambda_j$ .  $\diamond$

In our application we have  $N_t \ll N_x$ . Therefore, we determine the POD basis by using the  $N_t \times N_t$  correlation matrix

$$\mathcal{K}_{ij} = \langle y(t_j), y(t_i) \rangle_{\mathcal{H}}.$$

We assume the  $\ell^y$  largest eigenvalues of  $\mathcal{K}$  are given in the form  $\lambda_1^y \geq \dots \geq \lambda_{\ell^y}^y > 0$  and hence the POD basis is given by

$$\psi_i^y(x) = \frac{1}{\sqrt{\lambda_i^y}} \sum_{j=0}^{N_t} (v_i^y)_j y(t_j, x),$$

where  $v_k^y \in \mathbb{R}^{N_t}$  are the eigenvectors of  $\mathcal{K}$  to the corresponding eigenvalues  $\lambda_k^y$ . Note that  $\sigma_i^y = \sqrt{\lambda_i^y}$  corresponds to the singular values of the snapshot matrix  $Y = [y(t_0, \cdot), \dots, y(t_{N_t}, \cdot)]$  with respect to the  $\mathcal{H}$  inner product and the vectors  $\psi_i^y$  are computed from the left singular vectors. Note that the decay of the eigenvalues is essential since the approximation error for this approach is given by

$$(3.6) \quad \sum_{j=0}^{N_t} \left\| y(t_j) - \sum_{i=1}^{\ell^y} \langle y(t_j), \psi_i^y \rangle_{\mathcal{H}} \psi_i^y \right\|_{\mathcal{H}}^2 = \sum_{i=\ell^y+1}^{N_t} \lambda_i^y;$$

compare Section 4. Hence if the decay is too slow the resulting ROM will be of large dimension or one will obtain large approximation errors. In Figure 3.1 the decay of the singular values for all three variables is shown (left plot) together with the decay of the eigenvalues of the correlation matrix  $\mathcal{K}$  (right plot). The approach using the eigenvalues is preferred since the computational costs are cheaper. When computing the singular value decomposition one has to first discretize the inner product  $\langle \cdot, \cdot \rangle_{\mathcal{H}}$  which is usually done in the form  $\langle u, v \rangle_W = u^\top W v$  where  $W$  is a symmetric positive definite matrix. Note that since  $W$  is symmetric and positive definite,  $W$  possesses an eigenvalue decomposition of the form  $W = V D V^\top$ , where  $D$  is a diagonal matrix containing the eigenvalues  $\eta_1 \geq \dots \geq \eta_{N_x} > 0$  and  $V \in \mathbb{R}^{N_x \times N_x}$  is an orthogonal matrix. Hence we can define  $W^\alpha = V D^\alpha V^\top$  and  $D^\alpha = \text{diag}(\eta_1^\alpha, \dots, \eta_{N_x}^\alpha)$ . Thus the singular value decomposition with  $\mathcal{H}$  inner product is computed by the singular value decomposition of the matrix  $\tilde{Y} = W^{1/2} Y$ . Further to get the basis vectors  $\psi_i^y$  the matrix  $W^{-1/2}$  is needed. The computation of the matrices  $W^{1/2}$  and  $W^{-1/2}$  is very expensive. In Figure 3.1 (right plot) the eigenvalues of  $\mathcal{K}$  and the squared singular values of  $\tilde{Y}$  are shown. It can be seen that the first values are the same. The arising difference after approximately 20 values is a numerical issue since both methods work to the same accuracy. The



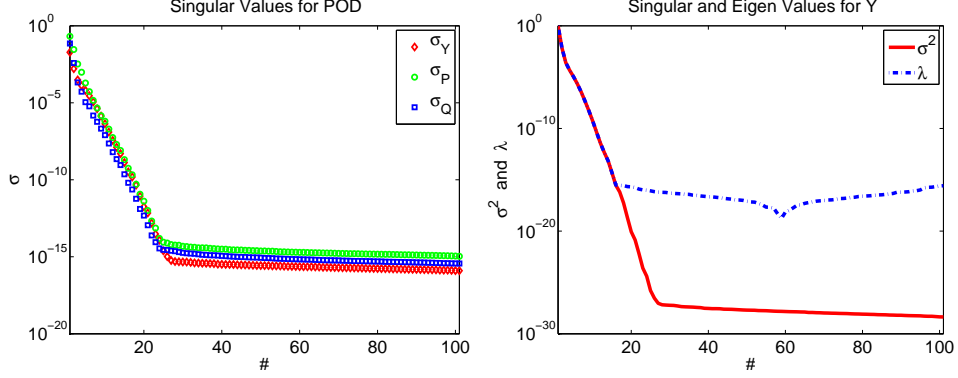


FIGURE 3.1. Decay of the singular values for the POD basis for  $y$ ,  $p$  and  $q$  (left) and a comparison of the decay of the singular values and eigenvalues for  $y$  (right).

difference only occurs since the singular values are squared for comparison. The increase of the eigenvalues after 60 computed values is also a numerical issue when the accuracy of the algorithm is exceeded. This issue is not influencing the POD method since we are only interested in the largest eigenvalues with an  $\ell^y$  smaller than 60. It can be seen that the kink in the eigenvalues can be used as a guess for the number of needed basis functions. This also coincides with the approximation error (3.6). In Figure 3.2 we compare the first six POD bases for the variable  $y$  obtained by the singular value decomposition and the eigenvalue decomposition. It can be seen that they are the same, the only difference is the sign of four of the six basis function, i.e., the second, third, fourth and fifth basis function. The settings for this numerical result are described in detail in Section 5, Run 1. Let us next introduce the reduced order model. For this purpose we look at (3.4). By using the POD Galerkin ansatz earlier introduced we can now write the ROM in the discrete form as

$$(3.7a) \quad \Psi_y^\top M \Psi_y \hat{y}_t(t) + \Psi_y^\top S_{c_1} \Psi_y \hat{y}(t) + \Psi_y^\top \bar{\mathcal{N}}(y^\ell(t), p^\ell(t), q^\ell(t); \mu) = 0,$$

$$(3.7b) \quad \Psi_p^\top S_{c_2} \Psi_p \hat{p}(t) + \Psi_p^\top \bar{\mathcal{N}}(y^\ell(t), p^\ell(t), q^\ell(t); \mu) = 0,$$

$$(3.7c) \quad \Psi_q^\top S_{c_3} \Psi_q \hat{q}(t) - \Psi_q^\top \bar{\mathcal{N}}(y^\ell(t), p^\ell(t), q^\ell(t); \mu) = 0,$$

where  $\Psi_f = [\psi_1^f, \dots, \psi_{\ell_f}^f]$  for  $f = \{y, p, q\}$ . To solve the ROM we can again apply the same strategy for the time discretization and the same solver as for the original problem. Let us remark that the dimension of the system to solve has decreased significantly since we assume that  $\ell^y + \ell^p + \ell^q \ll 3N_x$ .

**3.4. Empirical interpolation methods.** The ROM introduced in (3.7) is a non-linear system. Hence the problem with the POD Galerkin approach is the complexity of the evaluation of the nonlinearity. To illustrate this we have a look at the nonlinearity in (3.7a). We can write this as

$$\mathcal{N}^\ell(y^\ell(t), p^\ell(t), q^\ell(t); \mu) = \Psi_y^\top \bar{\mathcal{N}}(\Psi_y \hat{y}(t), \Psi_p \hat{p}(t), \Psi_q \hat{q}(t); \mu).$$

This can be interpreted in the way that the variables  $\hat{y}$ ,  $\hat{p}$  and  $\hat{q}$  are first expanded to a vector of dimension  $N_x$ , then the nonlinearity  $\bar{\mathcal{N}}$  is evaluated and at last the

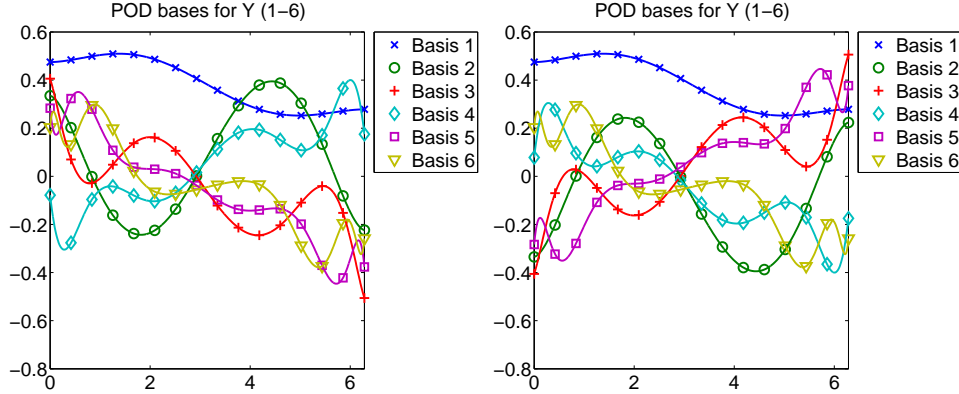


FIGURE 3.2. The first six POD basis functions for  $y$  computed using the correlation matrix  $\mathcal{K}$  (left) and the singular value decomposition (right).

result is reduced back to the low dimension  $\ell^y$  of the ROM. This is computationally expensive. Further this means that our ROM is not independent of the full dimension  $N_x$ . Note that when applying a Newton method to the system (3.7) the Jacobian of the nonlinearity is also needed. This can be expressed in the form

$$J_y^\ell(\mathcal{N}^\ell(y^\ell(t), p^\ell(t), q^\ell(t); \mu)) = \Psi_y^\top J_y(\mathcal{N}(\Psi_y \hat{y}(t), \Psi_p \hat{p}(t), \Psi_q \hat{q}(t); \mu)) \Psi_y,$$

where  $J_y(\mathcal{N})$  denotes the Jacobian with respect to  $y$  of  $\mathcal{N}$ . Again the same problem can be observed. Note that here the computation expenses are larger since the matrix  $J_y$  is of dimension  $N_x \times N_x$ . Hence not only a vector is transformed but a matrix of full dimension. To avoid this computational expensive evaluation the empirical interpolation method (EIM) was introduced [3]. This method is often used in the combination with the reduced basis approach [12]. The second approach we will investigate here is the discrete empirical interpolation method (DEIM) as introduced in [4, 6]. The basic concept of both methods is very similar. While the EIM implementation is based on a greedy algorithm the DEIM implementation is based on a POD approach combined with a greedy algorithm. We will now discuss both methods. We define

$$n(t, \mu) = \mathcal{N}(y^\ell(t), p^\ell(t), q^\ell(t); \mu).$$

Now  $n(t, \mu)$  is approximated by projecting on  $\phi_i \in \mathbb{R}^{N_x}$  for  $i = 1, \dots, \ell^{EIM}$ , i.e.  $n(t, \mu) \approx \Phi c(t, \mu)$ , with  $\Phi = [\phi_1, \dots, \phi_{\ell^{EIM}}]$ . Hence we can write the approximation of  $\mathcal{N}^\ell$  as

$$\mathcal{N}^\ell(y^\ell(t), p^\ell(t), q^\ell(t); \mu) \approx \Psi_y^\top \Phi c(t, \mu).$$

Here  $\ell^{EIM}$  denotes the number of basis functions chosen for the interpolation as well as the number of interpolation points. The question arising is how to compute  $\Phi$  and  $c(t, \mu)$ . Let  $\mathbf{p}^{EIM}$  be an index vector and  $\Phi \in \mathbb{R}^{N_x \times \ell^{EIM}}$  a matrix. Then by  $\Phi_{\{\mathbf{p}^{EIM}\}}$  we denote the submatrix consisting of the rows of  $\Phi$  corresponding to the indices in  $\mathbf{p}^{EIM}$ . Obviously, if we choose  $\ell^{EIM}$  indices then the overdetermined system  $n(t, \mu) = \Phi c(t, \mu)$  can be solved by choosing  $\ell^{EIM}$  rows of  $n(t, \mu)$  and  $\Phi$ . Here it is assumed that  $\Phi_{\{\mathbf{p}^{EIM}\}}$  is invertible.

Assuming we have computed  $\Phi$  and  $\mathbf{p}^{EIM}$  by an algorithm. Then we proceed as follows. For simplicity we introduce here the matrix  $P = [e_{\mathbf{p}_1^{EIM}}, \dots, e_{\mathbf{p}_{\ell^{EIM}}^{EIM}}]$ , where  $e_{\mathbf{p}_i^{EIM}} = [0, \dots, 0, 1, 0, \dots, 0]^\top \in \mathbb{R}^{N_x}$  is a vector with all zeros and at the  $\mathbf{p}_i^{EIM}$ -th row a one. Note that  $\Phi_{\{\mathbf{p}^{EIM}\}} = P^\top \Phi$  holds. To evaluate the approximate nonlinearity we need  $c(t, \mu)$ . Since we know  $\Phi$  and the index vector  $\mathbf{p}^{EIM}$  we can compute

$$c(t, \mu) = (P^\top \Phi)^{-1} P^\top n(t, \mu).$$

The nonlinearity on which we focus in this work can be evaluated pointwise and they only depend on the variables  $y, p, q$  and the parameter  $\mu$ . Hence the matrix  $P$  can be moved into the nonlinearity and we obtain

$$\begin{aligned} P^\top n(t, \mu) &= P^\top \mathcal{N}(y^\ell(t), p^\ell(t), q^\ell(t); \mu) \\ &= \mathcal{N}(P^\top \Psi_y \hat{y}(t), P^\top \Psi_p \hat{p}(t), P^\top \Psi_q \hat{q}(t); \mu). \end{aligned}$$

An extension for general nonlinearities is shown in [6]. Let us now have a look at the computational expenses. The matrices  $\Psi_f^\top \Phi \in \mathbb{R}^{\ell^f \times \ell^{EIM}}$ ,  $P^\top \Psi_f \in \mathbb{R}^{\ell^{EIM} \times \ell^f}$  for  $f = \{y, p, q\}$  and  $(P^\top \Phi)^{-1} \in \mathbb{R}^{\ell^{EIM} \times \ell^{EIM}}$  can be precomputed and are all independent of the full dimension  $N_x$ . Further, during the iterations the nonlinearity only has to be evaluated at the interpolation points, i.e. only at  $\ell^{EIM}$  points. This allows the ROM to be completely independent of the full dimension. Note that the used method is an interpolation and therefore is exact at the interpolation points. For the Jacobian the approach is similar. To summarize the results we now state the computation procedure for the evaluation of the nonlinearity

$$\mathcal{N}^\ell(y^\ell(t), p^\ell(t), q^\ell(t); \mu) = \Psi_y^\top \Phi (P^\top \Phi)^{-1} \mathcal{N}(P^\top \Psi_y \hat{y}(t), P^\top \Psi_p \hat{p}(t), P^\top \Psi_q \hat{q}(t); \mu)$$

and the Jacobian

$$\begin{aligned} J_y^\ell(\mathcal{N}^\ell(y^\ell(t), p^\ell(t), q^\ell(t); \mu)) &= \\ &= \Psi_y^\top \Phi (P^\top \Phi)^{-1} J_y(\mathcal{N}(P^\top \Psi_y \hat{y}(t), P^\top \Psi_p \hat{p}(t), P^\top \Psi_q \hat{q}(t); \mu)) P^\top \Psi_y \end{aligned}$$

for the ROM. For the variables  $p^\ell$  and  $q^\ell$  the results are very similar. Note that  $\Phi$  is independent of the three variables and hence only has to be computed once.

Let us now turn to the EIM and DEIM algorithms. When (2.4) is solved the nonlinearity  $\mathcal{N}$  is being evaluated for each time step. If these evaluations are stored the procedure to determine  $\Phi$  and the index vector  $\mathbf{p}_{EIM}$  does not involve any further evaluations of the nonlinearity. We denote by  $E$  the matrix with columns  $\mathcal{N}(y(t_i), p(t_i), q(t_i), \mu) \in \mathbb{R}^{N_x}$  for  $i = 1, \dots, N_t$ . Next let us have a look at the two algorithms of interest and let us present some numerical results. The detailed settings can be found in Section 5, Run 1.

In the algorithms  $\|\cdot\|_\infty$  stands for the maximum norm in  $\mathbb{R}^{N_x}$  and the operation ‘arg max’ returns the index, where the maximum entry occurs. In Algorithm 1 we state the EIM using a greedy algorithm. Here the basis  $\phi_i, i = 1, \dots, \ell^{EIM}$  is chosen from the provided snapshots of  $\mathcal{N}$  by scaling and shifting. The obtained basis is not orthonormal. The advantage of this method is that the submatrix  $\Phi_{\{\mathbf{p}_{EIM}\}}$  is an upper triangular matrix. Hence solving for  $c(t, \mu)$  is computationally cheap. The drawback of this method is that the computation of the basis is more expensive than the DEIM algorithm presented in Algorithm 2. The DEIM algorithm on the other hand generates the basis using the POD approach. Here the previously introduced POD approach is applied to the snapshots of the nonlinearity  $\mathcal{N}$  to compute  $\Phi$ .

---

**Algorithm 1** (The empirical interpolation method (EIM))
 

---

**Require:**  $\ell^{EIM}$  and snapshots  $E$

- 1:  $k \leftarrow \arg \max_{j=0, \dots, N_t} \|\bar{\mathcal{N}}(y(t_j), p(t_j), q(t_j), \mu)\|_\infty$
  - 2:  $\xi \leftarrow \bar{\mathcal{N}}(y(t_k), p(t_k), q(t_k), \mu)$
  - 3:  $\text{idx} \leftarrow \arg \max_{j=1, \dots, N_x} |\xi_j|$
  - 4:  $\phi_1 \leftarrow \xi / \xi_{\{\text{idx}\}}$
  - 5:  $\Phi = [\phi_1]$  and  $\mathbf{p}^{EIM} = \text{idx}$
  - 6: **for**  $i = 2$  to  $\ell^{EIM}$  **do**
  - 7: Solve  $\Phi_{\{\mathbf{p}^{EIM}\}c_j} = \bar{\mathcal{N}}(y(t_j), p(t_j), q(t_j), \mu)_{\{\mathbf{p}^{EIM}\}}$  for  $j = 0, \dots, N_t$
  - 8:  $k \leftarrow \arg \max_{j=0, \dots, N_t} \|\bar{\mathcal{N}}(y(t_j), p(t_j), q(t_j), \mu) - \Phi c_j\|_\infty$
  - 9:  $\xi \leftarrow \bar{\mathcal{N}}(y(t_k), p(t_k), q(t_k), \mu)$
  - 10:  $\text{idx} \leftarrow \arg \max_{j=1, \dots, N_x} |(\xi - \Phi c_k)_{\{j\}}|$
  - 11:  $\phi_i \leftarrow (\xi - \Phi c_k) / (\xi - \Phi c_k)_{\{\text{idx}\}}$
  - 12:  $\Phi \leftarrow [\Phi, \phi_i]$  and  $\mathbf{p}^{EIM} \leftarrow [\mathbf{p}^{EIM}, \text{idx}]$
  - 13: **end for**
  - 14: **return**  $\Phi$  and  $\mathbf{p}^{EIM}$
- 

In Figure 3.3 the decay of the singular values and a comparison to the eigenvalues are given. The observations are the same as previously described for the POD method. For a comparison the first six basis functions and interpolation points obtained by each of the two methods are shown in Figure 3.4. It can be seen that they are quite different which is due to their different properties. The selection for the interpolation points in both algorithms is similar and is based on a greedy algorithm. The idea is to successively select spatial points to limit the growth of an error bound. The indices are constructed inductively from the input data. For more details we refer the reader to [3, 4]. When looking at Figure 3.4 one can see that both methods select very similar interpolation points although they are not the same.

---

**Algorithm 2** (The discrete empirical interpolation method (DEIM))
 

---

**Require:**  $\ell^{EIM}$  and snapshots  $E$

- 1: Compute POD basis  $\Phi = [\phi_1, \dots, \phi_{EIM}]$  for  $E$
  - 2:  $\text{idx} \leftarrow \arg \max_{j=1, \dots, N_x} |(\phi_1)_{\{j\}}|$
  - 3:  $U = [\phi_1]$  and  $\mathbf{p}^{EIM} = \text{idx}$
  - 4: **for**  $i = 2$  to  $\ell^{EIM}$  **do**
  - 5:  $u \leftarrow \phi_i$
  - 6: Solve  $U_{\{\mathbf{p}^{EIM}\}c} = u_{\{\mathbf{p}^{EIM}\}}$
  - 7:  $r \leftarrow u - Uc$
  - 8:  $\text{idx} \leftarrow \arg \max_{j=1, \dots, N_x} |(r)_{\{j\}}|$
  - 9:  $U \leftarrow [U, u]$  and  $\mathbf{p}^{EIM} \leftarrow [\mathbf{p}^{EIM}, \text{idx}]$
  - 10: **end for**
  - 11: **return**  $\Phi$  and  $\mathbf{p}^{EIM}$
-

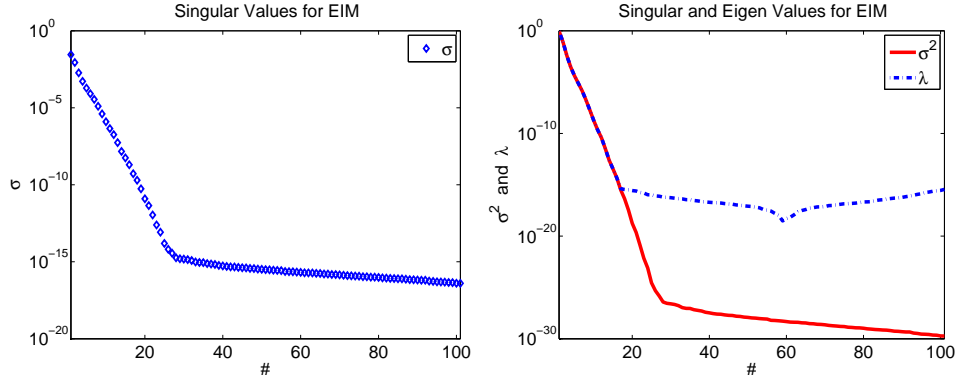


FIGURE 3.3. Decay of the singular values for the POD in DEIM (left) and comparison of the decay of singular and eigen values (right).

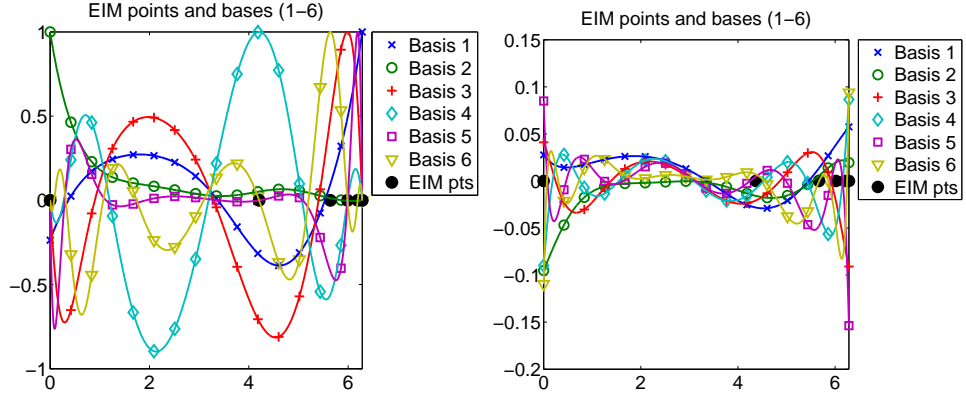


FIGURE 3.4. The first six basis functions utilized in EIM using the greedy approach (left) and the POD approach (right) together with the associated interpolation points.

#### 4. A-PRIORI ERROR ESTIMATES

In this section we present an a-priori error estimate for the difference between the solution to (2.4) and the solution to (3.4). The proof depends essentially on properties of the nonlinear function  $\mathcal{N}$ .

For an arbitrary chosen  $\mu \in \mathcal{P}_{ad}$  let  $z = \mathcal{S}(\mu)$  denote the unique solution to (2.4). In the context of Section 3 we choose  $\mathcal{H} = V$ ,  $\mathcal{H} = V_a$ ,  $\mathcal{H} = V_b$  to compute POD bases for the  $y$ ,  $p$  and  $q$  variables, respectively. We assume that Assumption 1 holds, i.e., there exists a unique solution  $z^\ell = z^\ell(\mu)$  to (3.4). In addition, we make use of the following hypothesis.

**Assumption 2.** *There exists a positive constant  $y_{\min}$  such that  $z(t, x)$  and  $z^\ell(t, x)$  belong to  $\mathcal{Z}_{ad}$  f.a.a.  $(t, x) \in Q$ . Moreover,  $\|z^\ell\|_{L^\infty(Q)^3} \leq C$  for a constant  $C > 0$  independent of  $\ell$ .*

We want to estimate the error

$$\int_0^T \|y(t) - y^\ell(t)\|_V^2 + \|p(t) - p^\ell(t)\|_V^2 + \|q(t) - q^\ell(t)\|_V^2 dt.$$

**Remark 4.1.** The proof of the a-priori error estimate works also for  $\Omega \subset \mathbb{R}^d$ ,  $d \in \{1, 2, 3\}$ . For that reason we use a notation which is appropriate also for the three-dimensional case.  $\diamond$

To introduce appropriate Ritz projections we define the symmetric, bounded, coercive bilinear forms

$$\begin{aligned} a^y(\varphi, \psi) &= \int_\Omega c_1 \nabla \varphi \cdot \nabla \psi + \varphi \psi \, dx && \text{for } \varphi, \psi \in V, \\ a^p(\varphi, \psi) &= \int_\Omega c_2 \nabla \varphi \cdot \nabla \psi \, dx && \text{for } \varphi, \psi \in V_a, \\ a^q(\varphi, \psi) &= \int_\Omega c_3 \nabla \varphi \cdot \nabla \psi \, dx && \text{for } \varphi, \psi \in V_b. \end{aligned}$$

Recall that

$$\langle \varphi, \psi \rangle_{V_{a,b}} = \int_\Omega \nabla \varphi \cdot \nabla \psi \, dx \quad \text{for } \varphi, \psi \in V$$

is an inner product on  $V_a$  and  $V_b$ . Moreover, the piecewise constant coefficient functions  $c_i$  are strictly positive. Hence, there exists a constant  $c_V \geq 1$  such that

$$(4.1) \quad \begin{aligned} a^y(\varphi, \varphi) &\geq c_V \|\varphi\|_V^2 \text{ for all } \varphi \in V, & a^p(\varphi, \varphi) &\geq c_V \|\varphi\|_V^2 \text{ for all } \varphi \in V_a, \\ a^q(\varphi, \varphi) &\geq c_V \|\varphi\|_V^2 \text{ for all } \varphi \in V_b, \end{aligned}$$

Let us define the Ritz projection  $\mathcal{P}^{\ell^y} : V \rightarrow V^{\ell^y}$  by

$$(4.2) \quad a^y(\mathcal{P}^{\ell^y} \varphi, \psi) = a^y(\varphi, \psi) \quad \text{for all } \psi \in V^{\ell^y}.$$

Analogously, we introduce the mapping  $\mathcal{P}^{\ell^p} : V_a \rightarrow V^{\ell^p}$  and  $\mathcal{P}^{\ell^q} : V_b \rightarrow V^{\ell^q}$ . Using the Ritz projections we decompose the error in the  $y$  component as

$$(4.3a) \quad y(t) - y^\ell(t) = y(t) - \mathcal{P}^{\ell^y} y(t) + \mathcal{P}^{\ell^y} y(t) - y^\ell(t) = \varrho^{\ell^y}(t) + \vartheta^{\ell^y}(t)$$

where  $\varrho^{\ell^y}(t) = y(t) - \mathcal{P}^{\ell^y} y(t)$  and  $\vartheta^{\ell^y}(t) = \mathcal{P}^{\ell^y} y(t) - y^\ell(t)$ . Analogously, let

$$(4.3b) \quad p(t) - p^\ell(t) = (p(t) - \mathcal{P}^{\ell^p} p(t)) + (\mathcal{P}^{\ell^p} p(t) - p^\ell(t)) = \varrho^{\ell^p}(t) + \vartheta^{\ell^p}(t),$$

$$(4.3c) \quad q(t) - q^\ell(t) = (q(t) - \mathcal{P}^{\ell^q} q(t)) + (\mathcal{P}^{\ell^q} q(t) - q^\ell(t)) = \varrho^{\ell^q}(t) + \vartheta^{\ell^q}(t).$$

The following result for the projection error follows from [17, Lemma 3].

**Lemma 4.2.** *The approximation error terms  $\varrho^{\ell^y}$ ,  $\varrho^{\ell^p}$  and  $\varrho^{\ell^q}$  satisfy*

$$\begin{aligned} \int_0^T \|\varrho^{\ell^y}(t)\|_V^2 dt &= \int_0^T \|y(t) - \mathcal{P}^{\ell^y} y(t)\|_V^2 dt \leq C \sum_{i=\ell^y+1}^{\infty} \lambda_i^y, \\ \int_0^T \|\varrho^{\ell^p}(t)\|_V^2 dt &= \int_0^T \|p(t) - \mathcal{P}^{\ell^p} p(t)\|_V^2 dt \leq C \sum_{i=\ell^p+1}^{\infty} \lambda_i^p, \\ \int_0^T \|\varrho^{\ell^q}(t)\|_V^2 dt &= \int_0^T \|q(t) - \mathcal{P}^{\ell^q} q(t)\|_V^2 dt \leq C \sum_{i=\ell^q+1}^{\infty} \lambda_i^q \end{aligned}$$

with  $C = \max(1, \|c_1\|_{L^\infty(\Omega)}, \|c_2\|_{L^\infty(\Omega)}, \|c_3\|_{L^\infty(\Omega)})/c_V$ .

In the following lemma we present an error estimate for the discretization error terms  $\vartheta^{\ell^p}(t)$  and  $\vartheta^{\ell^q}(t)$ . The proof is based on (2.4), (3.4) and the properties of the Ritz projections  $\mathcal{P}^{\ell^p}$ ,  $\mathcal{P}^{\ell^q}$ . For a proof we refer the reader to the appendix.

**Lemma 4.3.** *Suppose that Assumptions 1 and 2 hold. Then there exists a constant  $C > 0$  independent of  $\ell$  and  $t \in [0, T]$  so that*

$$(4.4) \quad \|\vartheta^{\ell^p}(t)\|_V^2 + \|\vartheta^{\ell^q}(t)\|_V^2 \leq C(\|\varrho^{\ell^y}(t)\|_V^2 + \|\varrho^{\ell^p}(t)\|_V^2 + \|\varrho^{\ell^q}(t)\|_V^2 + \|\vartheta^{\ell^y}(t)\|_H^2)$$

*f.a.a.  $t \in [0, T]$ .*

The first three terms on the right-hand side of (4.4) shall be bounded by using Lemma 4.2. From the parabolic equation an estimate is derived for the term  $\|\vartheta^{\ell^y}(t)\|_H^2$ . Let us mention that it is essential that  $\vartheta^{\ell^y}(t)$  occurs in the  $H$ - and not in the  $V$ -norm.

Now we turn to an estimate for the difference  $\vartheta^{\ell^y}(t) = \mathcal{P}^{\ell^y} y(t) - y^\ell(t)$ . The proof is also given in the appendix.

**Lemma 4.4.** *Suppose that Assumptions 1 and 2 hold. Then there exists a constant  $C > 0$  independent of  $\ell$  and  $t \in [0, T]$  so that*

$$(4.5) \quad \begin{aligned} & \|\vartheta^{\ell^y}\|_{L^\infty(0,T;H)}^2 + \|\vartheta^{\ell^y}\|_{L^2(0,T;V)}^2 \\ & \leq C(\|\vartheta^{\ell^y}(0)\|_H^2 + \|\mathcal{P}^{\ell^y} y_t - y_t\|_{L^2(0,T;H)}^2) \\ & \quad + C\left(\sum_{i=\ell^y+1}^{\infty} \lambda_i^y + \sum_{i=\ell^p+1}^{\infty} \lambda_i^p + \sum_{i=\ell^q+1}^{\infty} \lambda_i^q\right). \end{aligned}$$

From Lemmas 4.2, 4.3 and 4.4 we infer the existence of a constant  $C > 0$  such that

$$\begin{aligned} & \int_0^T \|y(t) - y^\ell(t)\|_V^2 + \|p(t) - p^\ell(t)\|_V^2 + \|q(t) - q^\ell(t)\|_V^2 dt \\ & \leq C(\|\mathcal{P}^{\ell^y} y_\circ - y^\ell(0)\|_H^2 + \|\mathcal{P}^{\ell^y} y_t - y_t\|_{L^2(0,T;H)}^2) \\ & \quad + C\left(\sum_{i=\ell^y+1}^{\infty} \lambda_i^y + \sum_{i=\ell^p+1}^{\infty} \lambda_i^p + \sum_{i=\ell^q+1}^{\infty} \lambda_i^q\right). \end{aligned}$$

**Remark 4.5.** (1) The term  $\|\mathcal{P}^{\ell^y} y_\circ - y^\ell(0)\|_H^2$  expresses how well the initial condition is approximated by the POD basis. It follows from [13, Remark 3.3] that

$$\|\mathcal{P}^{\ell^y} y_\circ - y^\ell(0)\|_H^2 \leq \|(\mathcal{P}^{\ell^y} - \mathcal{T}^{\ell^y})y_\circ\|_H^2 \xrightarrow{\ell \rightarrow \infty} 0,$$

where  $\mathcal{T}^{\ell^y} \varphi = \sum_{i=1}^{\ell^y} \langle \varphi, \psi_i^y \rangle_H \psi_i^y$  for  $\varphi \in H$ .

(2) Note that the error depends on the  $L^2(0, T; H)$  norm of the difference  $y_t - \mathcal{P}^{\ell^y} y_t$ . To avoid this dependence we have to include time derivatives into our snapshot set; see [17, 18].  $\diamond$

## 5. NUMERICAL EXPERIMENTS

In this section we present numerical results to the methods introduced in the previous sections. We choose here two settings. In the first run we demonstrate the effectiveness of the POD Galerkin scheme to solve the nonlinear elliptic parabolic system. In the second run we demonstrate a possible extension so that this

advantage can also be utilized in multiple solving of the PDE which can occur for example in the course of an optimization strategy.

5.1. **Run 1.** Let us start with setting up the problem. The nonlinearity has been introduced in (2.2). For this run the parameter  $\mu$  is set to  $(1, -1)$ . With this setting the PDE is very similar with the model to simulate lithium ion batteries introduced in [10, 25]. As an initial value for the parabolic equation we choose  $y_\circ(x) = 2 + \sin(x)$ . The diffusion coefficients  $c_1$ ,  $c_2$  and  $c_3$  are set to 1. For the domain  $\Omega$  the one dimensional interval  $[0, 2\pi]$  is chosen. The time interval  $[0, T]$ ,  $T = 1$ , is discretized in 100 equidistant steps. It is also possible to use an adaptive method for the time steps or even incorporate an optimal snapshot location algorithm to improve the quality of the generated snapshots [20]. For the discretization in space we choose 1000 discretization points and second order finite elements. This results in 1999 degrees of freedom. For the time an implicit Euler method is used for the full system of PDEs. To solve the full system a Newton method is utilized. It is important to solve the full system at once to obtain the most accurate results. When splitting the system into an elliptic and a parabolic part and solving them alternately the achieved results are less accurate and the variables  $y$ ,  $p$  and  $q$  may not satisfy the original problem to desired accuracy.

Next we discuss the settings for the simulation utilizing the POD method. We will report on the performance of the simulation using POD with and without the use of the EIM or DEIM method. Let us start with giving the dimension of the basis for the POD. Looking at Figure 3.1 we can see that the singular values decrease rapidly. Here we choose  $\ell^y = 12$ ,  $\ell^p = 10$  and  $\ell^q = 10$ . Further we choose  $\ell^{EIM} = 25$ . Comparing to Figure 3.3 this choice is reasonable for the DEIM method. For the EIM method we choose the same value for  $\ell^{EIM}$  as for the DEIM method in order to compare the computational performance.

To measure the accuracy of the POD method with respect to the finite element method the average relative error is introduced in the form

$$(5.1) \quad \varepsilon_y = \frac{1}{N_t + 1} \sum_{i=0}^{N_t} \frac{\|y^h(t_i) - y^\ell(t_i)\|_{L^2(\Omega)}}{\|y^h(t_i)\|_{L^2(\Omega)}},$$

where  $N_t$  is the number of time steps and  $t_i$  the discretization points in time. Note that we can define the average relative error in this way since we choose an equidistant discretization in time for our test. The average relative error of the variables  $p$  and  $q$  are introduced analogously. In Table 5.1 we compare the average relative errors. It can be observed that the empirical interpolation methods perform very well and there is no difference in accuracy when comparing EIM and DEIM.

|                 | ROM                     | ROM-EIM                 | ROM-DEIM                |
|-----------------|-------------------------|-------------------------|-------------------------|
| $\varepsilon_y$ | $1.6765 \times 10^{-7}$ | $1.6763 \times 10^{-7}$ | $1.6762 \times 10^{-7}$ |
| $\varepsilon_p$ | $2.8723 \times 10^{-7}$ | $2.7560 \times 10^{-7}$ | $2.7467 \times 10^{-7}$ |
| $\varepsilon_q$ | $9.7545 \times 10^{-8}$ | $9.4332 \times 10^{-8}$ | $9.1929 \times 10^{-8}$ |

TABLE 5.1. Comparison of the average relative errors when solving the ROM with and without EIM or DEIM.



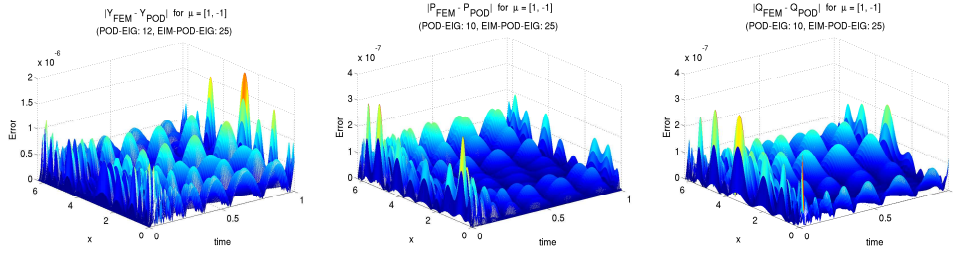


FIGURE 5.1. Absolute error between the finite element solution and the POD solution using DEIM for  $y$  (left),  $p$  (center) and  $q$  (right).

Further in Figure 5.1 the absolute error between the solution obtained by the finite element method and the POD method is shown. It can be seen that the POD method delivers nearly the same results as the finite element method and the error is distributed equally over the whole space-time domain.

The computational performance of POD method compared to the finite element method is summarized in Table 5.2. The speed up when using one of the two interpolation methods is significant. The achieved results are very similar as can be seen in Table 5.1. What should be noted is that the computation of the EIM is more expensive but leads to a more efficient ROM due to the properties of the interpolation matrix. So in case of multiple evaluations the EIM has a performance advantage for our particular problem.

|          | FEM   | POD  | EIM  | DEIM | ROM  | ROM-EIM | ROM-DEIM |
|----------|-------|------|------|------|------|---------|----------|
| CPU time | 18.20 | 0.20 | 0.19 | 0.03 | 6.03 | 0.24    | 0.48     |

TABLE 5.2. Summary of the performance for the finite element and POD method measured in seconds.

Additionally to the average relative error over the whole domain  $\Omega$  we have a look at the average relative boundary error. This is in particular interesting in the application where one can only measure certain quantities at the boundary. Therefore it is important that the POD approximation is accurate on the boundary. Let us define the average relative error for the boundary as

$$(5.2) \quad \bar{\varepsilon}_p(b) = \frac{1}{N_t + 1} \sum_{i=0}^{N-t} \frac{|p^h(t_i, b) - p^\ell(t_i, b)|}{|p^h(t_i, b)|}.$$

This error is interesting for the variables  $p(t, b)$  and  $q(t, a)$ . When comparing to the modeling of lithium ion batteries these two variables correspond to the potentials which can only be measured at the boundaries. In Table 5.3 we summarize the results. It can be seen that the POD approach delivers good results also when only looking at one particular point of the domain  $\Omega$ , in our case the boundary.

|                          | ROM                     | ROM-EIM                 | ROM-DEIM                |
|--------------------------|-------------------------|-------------------------|-------------------------|
| $\bar{\varepsilon}_p(b)$ | $3.3426 \times 10^{-7}$ | $3.2678 \times 10^{-7}$ | $3.3285 \times 10^{-7}$ |
| $\bar{\varepsilon}_q(a)$ | $6.6072 \times 10^{-8}$ | $9.0798 \times 10^{-8}$ | $7.4682 \times 10^{-8}$ |

TABLE 5.3. Comparison of the average relative errors on the boundary when solving the ROM with and without EIM or DEIM.

5.2. **Run 2.** In this run we focus on the case that the parameter  $\mu$  is changed. This is the case when using the POD method for optimization or parameter estimation. For this experiment we introduce two sets, one sample set on which finite element method is used and one test set to evaluate the reliability of the simulation using the POD method. The settings for the finite element discretization and solver are as described in Run 1. For the POD method we increase the number of basis functions to  $\ell^y = 18$ ,  $\ell^p = 22$  and  $\ell^q = 20$ . For the empirical interpolation method we use  $\ell^{EIM} = 40$  interpolation points. Next we introduce the sample and test set. We choose the two disjoint sets

$$M_{sample} = \{1, 2\} \times \{-2, -1\}$$

and

$$M_{test} = \{0.5, 1.5, 2.5, 3\} \times \{-3, -2.5, -1.5, -0.5\}.$$

This gives us 4 possible combinations for the sample set on which the PDE will be solved using the finite element discretization. The test set will give us 16 possible combinations on which the reduced order model will be solved using the POD method together with EIM or DEIM.

We use the average relative error introduced in (5.1) to measure the accuracy of the POD method with respect to the finite element method. In Figure 5.2 the results for the relative errors are summarized. In the right plot the average relative errors for the three variables is shown for the four parameters in the sample set. It can be seen that the POD method manages to approximate the solutions very well. In the left plot we report on the 16 parameters of the test set. Also for parameters which were not used to generate the POD basis we are able to achieve a sufficient accuracy. In Table 5.4 we summarize the performance of the POD method with respect to the computational time. Here we again get similar results as in the previous run. We can compute the time saved by using the POD method. Since we used the generated basis 16 times we saved approximately 17.5 seconds for each time we solved the system of nonlinear elliptic-parabolic PDEs using the POD method. On the other hand we have the time needed to generate the POD basis and four times solving the system using the finite element method. Further the basis for our interpolation method has to be computed. We observe for our test examples that for multiple evaluations of the system we were able to speed up the simulation with a factor of 35 when using EIM or DEIM. Without using an interpolation method only a factor of two can be achieved.

In Figure 5.3 we compare the average relative boundary error given in (5.2) for the different parameters  $\mu$ . Also here we see that the POD solution approximates the boundary points very accurate. As motivated previously this is important for certain applications.

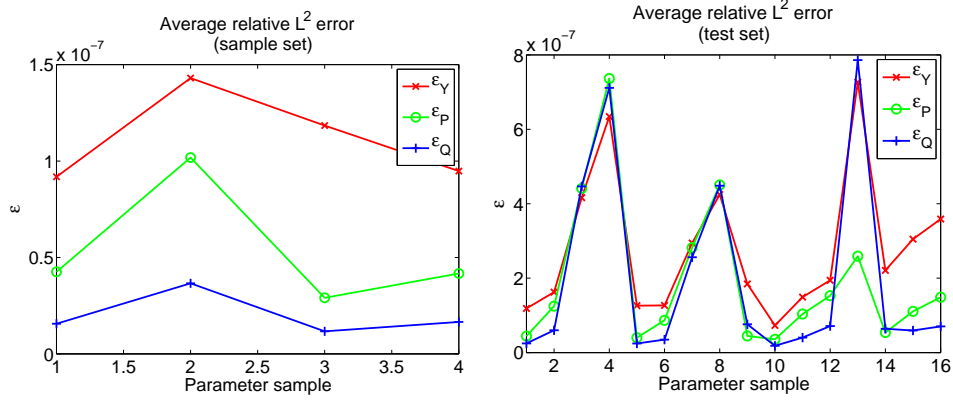


FIGURE 5.2. Average relative  $L^2$  error for the sample (left) and the test (right) set.

|          | FEM             | POD  | EIM  | DEIM | ROM                | ROM-EIM            | ROM-DEIM           |
|----------|-----------------|------|------|------|--------------------|--------------------|--------------------|
| CPU time | $\sim 18^{(*)}$ | 0.54 | 0.74 | 0.09 | $\sim 7.50^{(**)}$ | $\sim 0.30^{(**)}$ | $\sim 0.60^{(**)}$ |

TABLE 5.4. Summary of the performance for the finite element and POD method measured in seconds ((\*) for each sample pair, (\*\*)) for each test pair).

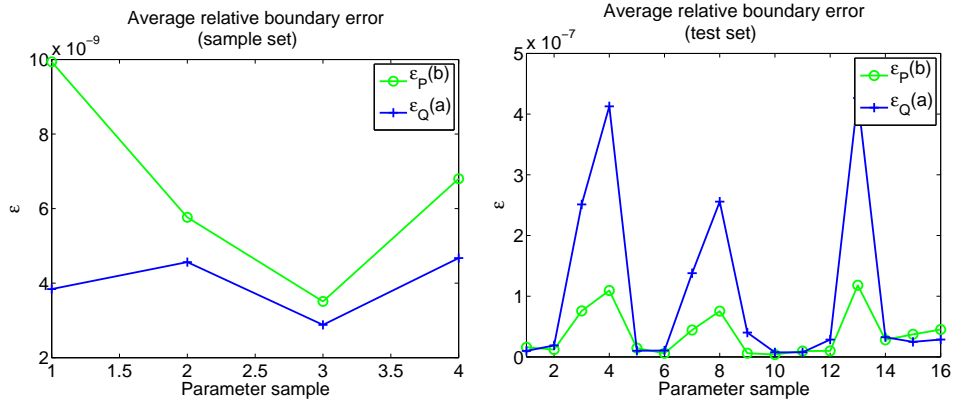


FIGURE 5.3. Average relative boundary error for the sample (left) and the test (right) set.

## 6. CONCLUSIONS

In the present paper a coupled semilinear elliptic-parabolic system is approximated by a POD Galerkin scheme. It turns out that the combination of the POD scheme with an interpolation of the nonlinearity leads to a fast and reliable reduced-order model. Theoretically, the POD Galerkin scheme can be proved to satisfy an a-priori error estimate, whereas the discretization error depends on the decay of the POD eigenvalues. The error estimate is based on the expensive evaluation of

the nonlinear term in the reduced-order model. However, by using the techniques from [5] we plan to extend the a-priori analysis to the case, where the discrete interpolation method is applied.

#### APPENDIX

**Proof of Lemma 4.3.** Recall that the mapping  $\mathcal{N}$  is continuously differentiable on  $\mathcal{Z}_{ad} \times \mathcal{P}_{ad}$ . Applying (4.3b), the property of the Ritz mapping  $\mathcal{P}^{\ell p}$ , (2.4b), (3.4b) and the integral form of the mean value theorem we obtain that

$$\begin{aligned} a^p(\vartheta^{\ell p}(t), \psi) &= a^p(\mathcal{P}^{\ell p} p(t), \psi) - a^p(p^\ell(t), \psi) = a^p(p(t), \psi) - a^p(p^\ell(t), \psi) \\ &= \langle \mathcal{N}(z^\ell(t); \mu) - \mathcal{N}(z(t); \mu), \psi \rangle_H \\ &= \int_0^1 \left\langle \frac{\partial \mathcal{N}}{\partial y}(\zeta^\ell(t, s); \mu)(y^\ell(t) - y(t)) + \frac{\partial \mathcal{N}}{\partial p}(\zeta^\ell(t, s); \mu)(p^\ell(t) - p(t)), \psi \right\rangle_H ds \\ &\quad + \int_0^1 \left\langle \frac{\partial \mathcal{N}}{\partial q}(\zeta^\ell(t, s); \mu)(q^\ell(t) - q(t)), \psi \right\rangle_H ds \end{aligned}$$

for all  $\psi \in V^{\ell p}$  and f.a.a.  $t \in [0, T]$ , where  $\zeta^\ell(t, s) = z(t) + s(z^\ell(t) - z(t))$ ,  $(t, s) \in [0, T] \times [0, 1]$ , parametrizes the straight line between  $z(t)$  and  $z^\ell(t)$ . Choosing  $\psi = \vartheta^{\ell p}(t)$  and using (4.1), (2.3) we find that

$$\begin{aligned} (A.1) \quad c_V \|\vartheta^{\ell p}(t)\|_V^2 &\leq - \int_0^1 \left\langle \frac{\partial \mathcal{N}}{\partial y}(\zeta^\ell(t, s); \mu)(\varrho^{\ell y}(t) + \vartheta^{\ell y}(t)), \vartheta^{\ell p}(t) \right\rangle_H ds \\ &\quad - \int_0^1 \left\langle \frac{\partial \mathcal{N}}{\partial p}(\zeta^\ell(t, s); \mu)(\varrho^{\ell p}(t) + \vartheta^{\ell p}(t)), \vartheta^{\ell p}(t) \right\rangle_H ds \\ &\quad + \int_0^1 \left\langle \frac{\partial \mathcal{N}}{\partial p}(\zeta^\ell(t, s); \mu)(\varrho^{\ell q}(t) + \vartheta^{\ell q}(t)), \vartheta^{\ell p}(t) \right\rangle_H ds \end{aligned}$$

f.a.a.  $t \in [0, T]$ . Analogously, we proceed for  $\vartheta^{\ell q}(t)$ . From (4.3c), (2.1c), (3.4c) and (2.3) we deduce that

$$\begin{aligned} a^q(\vartheta^{\ell q}(t), \varphi) &= \langle \mathcal{N}(z(t); \mu) - \mathcal{N}(z^\ell(t); \mu), \varphi \rangle_H \\ &= \int_0^1 \left\langle \frac{\partial \mathcal{N}}{\partial y}(\zeta^\ell(t, s); \mu)(y(t) - y^\ell(t)) + \frac{\partial \mathcal{N}}{\partial p}(\zeta^\ell(t, s); \mu)(p(t) - p^\ell(t)), \varphi \right\rangle_H ds \\ &\quad - \int_0^1 \left\langle \frac{\partial \mathcal{N}}{\partial p}(\zeta^\ell(t, s); \mu)(q(t) - q^\ell(t)), \varphi \right\rangle_H ds \end{aligned}$$

for all  $\varphi \in V^{\ell q}$  and f.a.a.  $t \in [0, 1]$ . Taking  $\varphi = \vartheta^{\ell q}$  and using (4.1) it follows that

$$\begin{aligned} (A.2) \quad c_V \|\vartheta^{\ell q}(t)\|_V^2 &\leq \int_0^1 \left\langle \frac{\partial \mathcal{N}}{\partial y}(\zeta^\ell(t, s); \mu)(\varrho^{\ell y}(t) + \vartheta^{\ell y}(t)), \vartheta^{\ell q}(t) \right\rangle_H ds \\ &\quad + \int_0^1 \left\langle \frac{\partial \mathcal{N}}{\partial p}(\zeta^\ell(t, s); \mu)(\varrho^{\ell p}(t) + \vartheta^{\ell p}(t)), \vartheta^{\ell q}(t) \right\rangle_H ds \\ &\quad - \int_0^1 \left\langle \frac{\partial \mathcal{N}}{\partial p}(\zeta^\ell(t, s); \mu)(\varrho^{\ell q}(t) + \vartheta^{\ell q}(t)), \vartheta^{\ell q}(t) \right\rangle_H ds \end{aligned}$$

f.a.a.  $t \in [0, 1]$ . Due to Assumption 2 we have  $\zeta^\ell(t, x) \in \mathcal{Z}_{ad}$  f.a.a.  $(t, x) \in Q$  and  $\|\zeta^\ell\|_{L^\infty(Q)^3} \leq C$  with a constant independent of  $\ell$ . Therefore, there exists a

constant  $c_{\mathcal{N}} > 0$  independent of  $\ell$  such that

$$\operatorname{esssup}_{t \in [0, T]} \left( \max \left\{ \int_0^1 \left\| \frac{\partial \mathcal{N}}{\partial y}(\zeta^\ell(t, s)) \right\|_{L^\infty(\Omega)} ds, \int_0^1 \left\| \frac{\partial \mathcal{N}}{\partial p}(\zeta^\ell(t, s)) \right\|_{L^\infty(\Omega)} ds \right\} \right) \leq c_{\mathcal{N}}.$$

Adding (A.1), (A.2) and utilizing (2.3) we obtain that

$$\begin{aligned} & c_V (\|\vartheta^{\ell^p}(t)\|_V^2 + \|\vartheta^{\ell^q}(t)\|_V^2) \\ & \leq \int_0^1 \left\langle \frac{\partial \mathcal{N}}{\partial y}(\zeta^\ell(t, s); \mu), (\varrho^{\ell^y}(t) + \vartheta^{\ell^y}(t))(\vartheta^{\ell^q}(t) - \vartheta^{\ell^p}(t)) \right\rangle_H ds \\ & \quad + \int_0^1 \left\langle -\frac{\partial \mathcal{N}}{\partial p}(\zeta^\ell(t, s); \mu), (\vartheta^{\ell^p}(t) - \vartheta^{\ell^q}(t))^2 \right\rangle_H ds \\ & \quad + \int_0^1 \left\langle \frac{\partial \mathcal{N}}{\partial p}(\zeta^\ell(t, s); \mu), \varrho^{\ell^p}(t)\vartheta^{\ell^q}(t) + \varrho^{\ell^q}(t)\vartheta^{\ell^p}(t) \right\rangle_H ds \\ & \leq c_{\mathcal{N}} (\|\varrho^{\ell^y}(t)\|_H + \|\vartheta^{\ell^y}(t)\|_H) (\|\vartheta^{\ell^q}(t)\|_H + \|\vartheta^{\ell^p}(t)\|_H) \\ & \quad + c_{\mathcal{N}} (\|\varrho^{\ell^p}(t)\|_H \|\vartheta^{\ell^q}(t)\|_H + \|\varrho^{\ell^q}(t)\|_H \|\vartheta^{\ell^p}(t)\|_H). \end{aligned}$$

Since  $V$ ,  $V_a$  and  $V_b$  are continuously embedded into  $H$ , there exists a constant  $c_H > 0$  such that  $\|\varphi\|_H \leq c_H \|\varphi\|_V$  for all  $\varphi \in V$ . Consequently,

$$\begin{aligned} & c_V (\|\vartheta^{\ell^p}(t)\|_V^2 + \|\vartheta^{\ell^q}(t)\|_V^2) \\ & \leq C_1 (\|\varrho^{\ell^y}(t)\|_V \|\vartheta^{\ell^q}(t)\|_V + \|\vartheta^{\ell^y}(t)\|_H \|\vartheta^{\ell^q}(t)\|_V + \|\varrho^{\ell^y}(t)\|_V \|\vartheta^{\ell^p}(t)\|_V) \\ & \quad + C_1 (\|\vartheta^{\ell^y}(t)\|_H \|\vartheta^{\ell^p}(t)\|_V + \|\varrho^{\ell^p}(t)\|_V \|\vartheta^{\ell^q}(t)\|_V + \|\varrho^{\ell^q}(t)\|_V \|\vartheta^{\ell^p}(t)\|_V) \end{aligned}$$

f.a.a.  $t \in [0, T]$  and with  $C_1 = c_{\mathcal{N}} \max(c_H, c_H^2)$ . Using Young's inequality [2, p. 28] we obtain

$$\begin{aligned} & c_V (\|\vartheta^{\ell^p}(t)\|_V^2 + \|\vartheta^{\ell^q}(t)\|_V^2) \\ & \leq \frac{C_1^2}{2\varepsilon} (2\|\varrho^{\ell^y}(t)\|_V^2 + 2\|\vartheta^{\ell^y}(t)\|_H^2 + \|\varrho^{\ell^p}(t)\|_V^2 + \|\varrho^{\ell^q}(t)\|_V^2) \\ & \quad + \frac{3\varepsilon}{2} (\|\vartheta^{\ell^q}(t)\|_V^2 + \|\vartheta^{\ell^p}(t)\|_V^2) \end{aligned}$$

for an arbitrary  $\varepsilon > 0$ . Choosing  $\varepsilon = c_V/3$  and setting  $C_2 = 3C_1^2/c_V$  we find that

$$(A.3) \quad \|\vartheta^{\ell^p}(t)\|_V^2 + \|\vartheta^{\ell^q}(t)\|_V^2 \leq C_2 (\|\vartheta^{\ell^y}(t)\|_H^2 + \|\varrho^{\ell^y}(t)\|_V^2 + \|\varrho^{\ell^p}(t)\|_V^2 + \|\varrho^{\ell^q}(t)\|_V^2)$$

f.a.a.  $t \in [0, T]$ .  $\square$

**Proof of Lemma 4.4.** Using (4.3a), (4.2), (2.4a), (3.4a) we find

$$\begin{aligned} & \frac{d}{dt} \langle \vartheta^{\ell^y}(t), \psi \rangle_H + a^y \langle \vartheta^{\ell^y}(t), \psi \rangle = \langle \mathcal{P}^{\ell^y} y_t(t) - y_t^\ell(t), \psi \rangle_H + a^y \langle \mathcal{P}^{\ell^y} y(t) - y^\ell(t), \psi \rangle \\ & = \langle \mathcal{N}(z^\ell(t); \mu) - \mathcal{N}(z(t); \mu) + y(t) - y^\ell(t) + \mathcal{P}^{\ell^y} y_t(t) - y_t(t), \psi \rangle_H \\ & = \int_0^1 \left\langle \frac{\partial \mathcal{N}}{\partial y}(\zeta^\ell(t, s); \mu)(y^\ell(t) - y(t)) + \frac{\partial \mathcal{N}}{\partial p}(\zeta^\ell(t, s); \mu)(p^\ell(t) - p(t)), \psi \right\rangle_H ds \\ & \quad - \int_0^1 \left\langle \frac{\partial \mathcal{N}}{\partial p}(\zeta^\ell(t, s); \mu)(q^\ell(t) - q(t)) + y(t) - y^\ell(t) + \mathcal{P}^{\ell^y} y_t(t) - y_t(t), \psi \right\rangle_H ds, \end{aligned}$$

where  $\zeta^\ell(t, s) = z(t) + s(z^\ell(t) - z(t))$ ,  $(t, s) \in [0, T] \times [0, 1]$ , parametrizes the straight line between  $z(t)$  and  $z^\ell(t)$ . Recall that we have introduced the constant  $c_{\mathcal{N}}$  in the

proof of Lemma 4.3. Choosing  $\psi = \vartheta^{\ell^y}(t)$ ,  $C_1 = \max(c_N \max(1, c_H), c_H, 1)/2$  and applying (4.1) we find that

$$\begin{aligned} & \frac{1}{2} \frac{d}{dt} \|\vartheta^{\ell^y}(t)\|_H^2 + c_V \|\vartheta^{\ell^y}(t)\|_V^2 \\ & \leq 2C_1 (\|\varrho^{\ell^y}(t)\|_V \|\vartheta^{\ell^y}(t)\|_H + \|\vartheta^{\ell^y}(t)\|_H^2 + \|\varrho^{\ell^p}(t)\|_V \|\vartheta^{\ell^y}(t)\|_H) \\ & \quad + 2C_1 (\|\varrho^{\ell^q}(t)\|_V \|\vartheta^{\ell^y}(t)\|_H + \|\vartheta^{\ell^q}(t)\|_V \|\vartheta^{\ell^y}(t)\|_H + \|\vartheta^{\ell^p}(t)\|_V \|\vartheta^{\ell^y}(t)\|_H) \\ & \quad + 2C_1 (\|\varrho^{\ell^y}(t)\|_V \|\vartheta^{\ell^y}(t)\|_H + \|\vartheta^{\ell^y}(t)\|_H^2 + \|\mathcal{P}^{\ell^y} y_t(t) - y_t(t)\|_H \|\vartheta^{\ell^y}(t)\|_H) \\ & \leq C_1^2 (2\|\varrho^{\ell^y}(t)\|_V^2 + \|\varrho^{\ell^p}(t)\|_V^2 + \|\vartheta^{\ell^p}(t)\|_V^2 + \|\varrho^{\ell^q}(t)\|_V^2 + \|\vartheta^{\ell^q}(t)\|_V^2) \\ & \quad + C_1^2 (11\|\vartheta^{\ell^y}(t)\|_H^2 + \|\mathcal{P}^{\ell^y} y_t(t) - y_t(t)\|_H^2). \end{aligned}$$

From (A.3) we infer that

$$\begin{aligned} & \frac{d}{dt} \|\vartheta^{\ell^y}(t)\|_H^2 + 2c_V \|\vartheta^{\ell^y}(t)\|_V^2 \\ & \leq C_2 (\|\varrho^{\ell^y}(t)\|_V^2 + \|\varrho^{\ell^p}(t)\|_V^2 + \|\varrho^{\ell^q}(t)\|_V^2 + \|\vartheta^{\ell^y}(t)\|_H^2 + \|\mathcal{P}^{\ell^y} y_t(t) - y_t(t)\|_H^2), \end{aligned}$$

where  $C_2 = 22C_1^2$ . Integrating over time and using Lemma 4.2 we get

$$\begin{aligned} & \|\vartheta^{\ell^y}(t)\|_H^2 + 2 \int_0^t c_V \|\vartheta^{\ell^y}(s)\|_V^2 ds \\ & \leq C_2 \int_0^T (\|\varrho^{\ell^y}(t)\|_V^2 + \|\varrho^{\ell^p}(t)\|_V^2 + \|\varrho^{\ell^q}(t)\|_V^2 + \|\vartheta^{\ell^y}(t)\|_H^2 + \|\mathcal{P}^{\ell^y} y_t(t) - y_t(t)\|_H^2) dt \\ & \leq C_2 \left( \sum_{i=\ell^y+1}^{\infty} \lambda_i^y + \sum_{i=\ell^p+1}^{\infty} \lambda_i^p + \sum_{i=\ell^q+1}^{\infty} \lambda_i^q \right) \\ & \quad + C_2 \left( \|\mathcal{P}^{\ell^y} y_\circ - y^\ell(0)\|_H^2 + \|\mathcal{P}^{\ell^y} y_t - y_t\|_{L^2(0,T;H)}^2 \right) \end{aligned}$$

which gives

$$\begin{aligned} \|\vartheta^{\ell^y}(t)\|_H^2 & \leq C_2 \left( \sum_{i=\ell^y+1}^{\infty} \lambda_i^y + \sum_{i=\ell^p+1}^{\infty} \lambda_i^p + \sum_{i=\ell^q+1}^{\infty} \lambda_i^q \right) \\ & \quad + C_2 \left( \|\mathcal{P}^{\ell^y} y_\circ - y^\ell(0)\|_H^2 + \|\mathcal{P}^{\ell^y} y_t - y_t\|_{L^2(0,T;H)}^2 \right) \end{aligned}$$

f.a.a.  $t \in [0, T]$  and

$$\begin{aligned} \|\vartheta^{\ell^y}(t)\|_{L^2(0,T;V)}^2 & \leq C_2 \left( \sum_{i=\ell^y+1}^{\infty} \lambda_i^y + \sum_{i=\ell^p+1}^{\infty} \lambda_i^p + \sum_{i=\ell^q+1}^{\infty} \lambda_i^q \right) \\ & \quad + C_2 \left( \|\mathcal{P}^{\ell^y} y_\circ - y^\ell(0)\|_H^2 + \|\mathcal{P}^{\ell^y} y_t - y_t\|_{L^2(0,T;H)}^2 \right). \end{aligned}$$

□

## REFERENCES

- [1] R.A. Adams. *Sobolev Spaces*. Pure and Applied Mathematics, Vol. 65. Academic Press, New York-London, 1975.
- [2] H.W. Alt. *Lineare Funktionalanalysis. Eine anwendungsorientierte Einführung*. Springer-Verlag, Berlin, 1992.

- [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. *Comptes Rendus Mathematique*, 339(9):667-672, 2004.
- [4] S. Chaturantabut, and D. C. Sorensen. Application of POD and DEIM on a Dimension Reduction of Nonlinear Miscible Viscous Fingering in Porous Media. *Technical Report*, TR09-25, RICE University, 2009.
- [5] S. Chaturantabut, and D. C. Sorensen. A state space estimate for POD-DEIM Nonlinear Model Reduction. *Technical Report*, TR10-32, RICE University, 2010.
- [6] S. Chaturantabut, and D. C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM J. Sci. Comput.*, 32(5):2737-2764, 2010.
- [7] R. Dautray and J.-L. Lions. *Mathematical Analysis and Numerical Methods for Science and Technology. Volume 5: Evolution Problems I*. Springer-Verlag, Berlin, 1992.
- [8] P. Deuffhard. *Newton Methods for Nonlinear Problems. Affine Invariance and Adaptive Algorithms*. Series Computational Mathematics 35, Springer, Berlin, 2004.
- [9] P. Deuffhard and A. Hohmann. *Numerische Mathematik 1. Eine algorithmisch orientierte Einführung*. 4. überarbeitete und erweiterte Auflage. de Gruyter, Berlin, New York, 2008.
- [10] M. Doyle, T. Fuller and J. Newman. *Modeling of galvanostatic charge and discharge of the lithium ion battery/polymer/insertion cell*. *J. Chem. Soc.*, 140(6):1526-1533, 1993.
- [11] L.C. Evans. *Partial Differential Equations*. American Math. Society, Providence, Rhode Island, 2002.
- [12] M. Grepl. *Certified reduced basis method for nonaffine time-varying and nonlinear parabolic partial differential equations*. Submitted, 2011.
- [13] M. Hinze and S. Volkwein. Error estimates for abstract linear-quadratic optimal control problems using proper orthogonal decomposition. *Comput. Optim. Appl.*, 39:319-345, 2008.
- [14] P. Holmes, J.L. Lumley, and G. Berkooz. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge Monographs on Mechanics, Cambridge University Press, 1996.
- [15] D. Hömberg and S. Volkwein. Control of laser surface hardening by a reduced-order approach using proper orthogonal decomposition. *Mathematical and Computer Modeling*, 38:1003-1028, 2003.
- [16] M. Kahlbacher and S. Volkwein. Galerkin proper orthogonal decomposition methods for parameter dependent elliptic systems. *Discussiones Mathematicae: Differential Inclusions, Control and Optimization*, 27:95-117, 2007.
- [17] K. Kunisch and S. Volkwein. Galerkin proper orthogonal decomposition methods for parabolic problems. *Numerische Mathematik*, 90:117-148, 2001.
- [18] K. Kunisch and S. Volkwein. Galerkin proper orthogonal decomposition methods for a general equation in fluid dynamics. *SIAM J. Numer. Anal.*, 40:492-515, 2002.
- [19] K. Kunisch and S. Volkwein. Proper orthogonal decomposition for optimality systems. *ESAIM: M2AN*, 42:1-23, 2008.
- [20] K. Kunisch and S. Volkwein. Optimal snapshot location for computing POD basis functions. *ESAIM: M2AN*, 44:509-259, 2010.
- [21] A.T. Patera and G. Rozza. *Reduced Basis Approximation and A Posteriori Error Estimation for Parametrized Partial Differential Equations*. MIT Pappalardo Graduate Monographs in Mechanical Engineering, 2006.
- [22] P. Popov, Y. Vutov, S. Mergenov, and O. Iliev. Finite volume discretization of equations describing nonlinear diffusion in li-ion batteries. In Proceedings of the 7th International Conference, NMA 2010, Borovets, Bulgaria, August 20-24, 2010, I. Dimov, S. Dimova, N. Kolkovska (Eds.), LNCS, 6046:338-346, 2011.
- [23] E.W. Sachs and M. Schu. A-priori error estimates for reduced order models in finance. Submitted, 2011.
- [24] L. Sirovich. Turbulence and the dynamics of coherent structures. Parts I-II. *Quart. Appl. Math.*, XVI:561-590, 1987.
- [25] J. Wu, J. Xu, and H. Zou. On the well posedness of mathematical model for lithium-ion battery systems. *Methods and Applications of Analysis*, 13:275-298, 2006.

OLIVER LASS, UNIVERSITÄT KONSTANZ, FACHBEREICH MATHEMATIK UND STATISTIK, UNIVERSITÄTSSTRASSE 10, D-78457 KONSTANZ, GERMANY  
*E-mail address:* `Oliver.Lass@uni-konstanz.de`

STEFAN VOLKWEIN, UNIVERSITÄT KONSTANZ, FACHBEREICH MATHEMATIK UND STATISTIK, UNIVERSITÄTSSTRASSE 10, D-78457 KONSTANZ, GERMANY  
*E-mail address:* `Stefan.Volkwein@uni-konstanz.de`