POD-DEIM for Efficient Reduction of a Dynamic 2D Catalytic Reactor Model

Jens Bremer[a], Pawan Goyal[a], Lihong Feng[a], Peter Benner[a,b], and Kai Sundmacher[a,b]

[a]Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstraße 1, 39106 Magdeburg, Germany

[b]Otto-von-Guericke University, Universitätsplatz 2, 39106 Magdeburg, Germany

April 6, 2017

Abstract

Many computational difficulties in dealing with chemical process models often result from spatially distributed states as well as nonlinear correlations (e.g., for transport coefficients or reaction kinetics). Surrogate models with sufficient accuracy represent one remedy to this problem. Featuring a lower number of states, model order reduction (MOR) generates considerably less complex models and leads to faster model evaluations. Especially for nonlinear systems, snapshot-based MOR techniques are considered to be one of the most promising methods. In this study, we apply proper orthogonal decomposition together with the discrete empirical interpolation method (POD-DEIM) to a dynamic, two-dimensional reactor model for catalytic carbon dioxide methanation. Motivated by renewable energy integration, we consider this reactor in two different dynamic scenarios: Disturbed continuous operation and start-up. It can be shown that the reduced order model (ROM) is accurate and, furthermore, the solution of the FOM is accelerated at least by one order of magnitude.

Keywords: Nonlinear model reduction; Proper orthogonal decomposition; Empirical interpolation methods; Catalytic reactor; Methanation

1 Introduction

During the last decades, an increasing usage of highly complex mathematical models, supported by the rapid development of computational capacities, can be observed. For the identification of further potential in energy efficiency and cost reduction, these complex models appear to be very promising, especially in the field of process systems engineering. However, large state-spaces, or nonlinear correlations for the description of underlying chemical and physical phenomena can cause a challenging level of complexity. Beyond that, the level of complexity enhanced even more due to the widespread use of dynamic models. Consequently, the application of these models for simulation, optimization, and process control is still a challenging task (Biegler et al., 2014).

As an example of current interest, the integration of renewable resources to chemical and energy conversion systems requires a deeper understanding of various processes and their dynamic behavior. In this context, Fig. 1 illustrates a promising process route for the conversion of renewable, electrical energy. Due to their volatile nature, input streams coming from renewable energy are often challenging to deal with (Güttel, 2013). Hence, start-up, shut-down but also disturbed continuous operations are gaining more interest (Adams and Barton, 2009; Güttel, 2013). Furthermore, the major process units are characterized by local diffusion, thermal conduction and multi-step reaction mechanisms. To capture all these features, dynamic process models with fine spatial resolution are required. Nevertheless, the resulting large state-space dimensions of order $\mathcal{O}(10^3)$ – $\mathcal{O}(10^5)$ still exacerbate the usage of these models for optimization, control or even simulation. Model order reduction (MOR) can be considered as an essential tool to mitigate this situation and to allow for fast computation (Baur et al., 2014; Marquardt, 2002).

As an exemplary unit of an entire process for renewable energy integration, this study is focused on a catalytic reactor to investigate the potential of MOR. This unit is usually realized by a tubular reactor either filled with catalyst particles or coated with catalyst at the inner wall. There are many
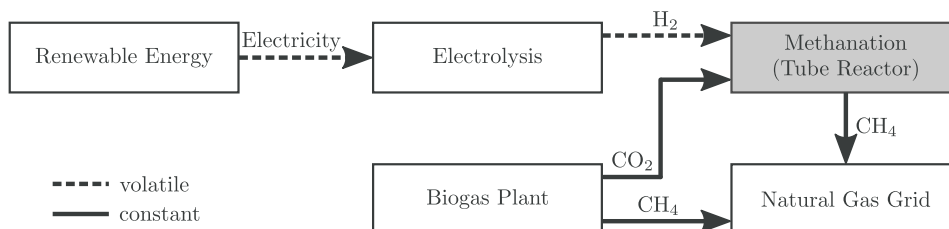

Figure 1: Exemplary process route for a renewable conversion of electrical energy to methane. (Bremer et al., 2016)

studies about this kind of reactors available. For instance, to control the hot-spot formation during start-up, Bremer et al. (2017) developed a rigorous, two-dimensional, packed-bed carbon dioxide methanation reactor model. They reported run times of several minutes per simulation run. Other two and one-dimensional reactor models are mentioned with CPU times of several minutes to hours (Pantoleontos et al., 2012; Adams and Barton, 2009). This computational burden still prevents detailed investigations on the reactor dynamics for various applications. In our perspective, MOR shows a vast potential for improvement in computational time for these reactor models and, thus, reveals new insights for process design and dynamic operation.

In this work, we investigate a catalytic wall carbon dioxide methanator in two different dynamic scenarios: Disturbed continuous operation and start-up. Since, methanation is a strongly exothermic reaction, heat management is of major interest Rönsch et al. (2016). Thus, key control variables for heat management are the cooling temperature at the reactor wall and the inlet gas composition. The aim is to develop a ROM which is capable of dealing with different cooling and carbon dioxide feeding policies, coming from a disturbed hydrogen inlet (see Fig. 1).

2   Full-Order Reactor Model

Considering a tubular geometry of the catalytic wall reactor, the full-order model (FOM) is derived based on mass and energy balances. Instead of taking the momentum balance into account, we assume a fully-developed laminar velocity profile. Furthermore, pressure drop in the tube is neglected due to small Reynolds numbers. The further main assumptions are:

- No change in φ-direction, due to symmetric geometry.
- Axial diffusion is neglected, due to small Reynolds numbers (< 2300).
- Incompressible gas flow with ideal gas behavior.
- $CH_4, CO, CO_2, H_2O, H_2$ and N_2 are the considered species.
- CO_2 methanation described by rate expressions proposed by Xu and Froment (1989), see Fig. 2.
- The mass transport into the thin catalytic layer at the wall is significantly faster than to the catalytic reaction.
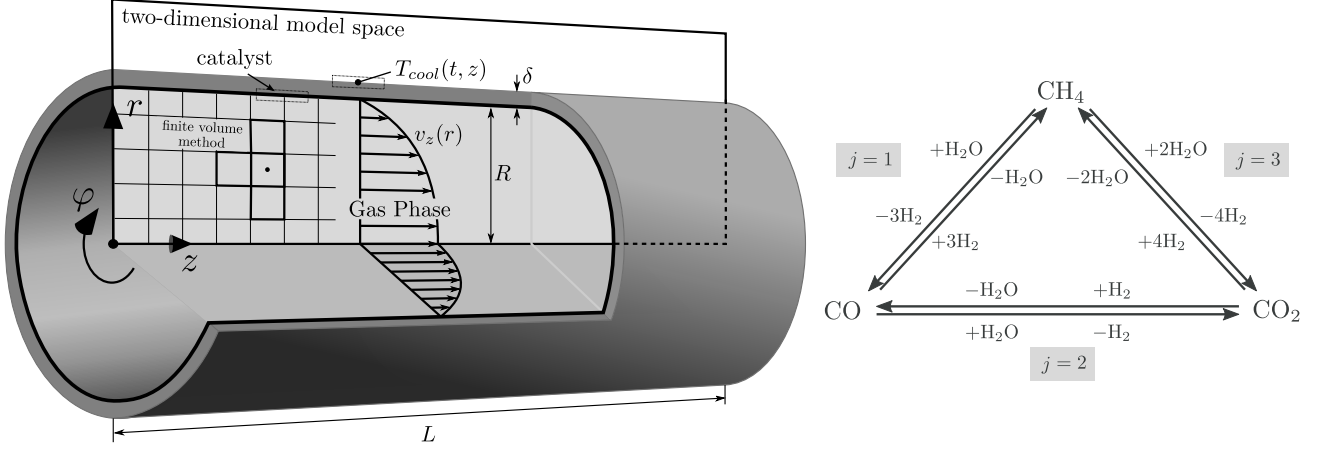- The catalytic reaction is, furthermore, assumed to be in quasi-steady state.

Figure 2: Left: An illustration of the reactor model (Bremer et al., 2016); Right: The considered reaction network proposed by Xu and Froment (1989).

Accordingly, we obtain a two-dimensional, heterogeneous, dynamic reactor model as illustrated in Fig. 2. The dynamics of the reactor is governed by the following set of PDEs:

$$\frac{\partial \rho_\alpha}{\partial t} = -v_z \frac{\partial \rho_\alpha}{\partial z} + \mathcal{D}_{r,\alpha} \left[\frac{\partial^2 \rho_\alpha}{\partial r^2} + \frac{1}{r} \frac{\partial \rho_\alpha}{\partial r} \right], \quad \alpha = 1 \ldots 6 \quad \text{(species)}, \tag{1}$$

$$\frac{\partial T}{\partial t} = -v_z \frac{\partial T}{\partial z} + \frac{\lambda_r}{\rho c_p} \left[\frac{\partial^2 T}{\partial r^2} + \frac{1}{r} \frac{\partial T}{\partial r} \right], \tag{2}$$

where ρ_α relates to the mass concentration of each component α, $\mathcal{D}_{r,\alpha}$ corresponds to the radial gas diffusion coefficient of each component; T represents the gas temperature, λ_r is the radial thermal gas conductivity; ρ is the gas mixture mass density; c_p stands for the heat capacity of the gas mixture, and v_z relates to a given laminar velocity profile. In order to fully determine the above system of PDEs, boundary conditions, with respect to the reactor's symmetry and the catalytic reaction at the inner wall, are given by

$$\rho_\alpha|_{z=0} = \rho_{\alpha,\text{in}} \, , \qquad \frac{\partial \rho_\alpha}{\partial r}\bigg|_{r=0} = 0, \qquad \frac{\partial \rho_\alpha}{\partial r}\bigg|_{r=R} = \frac{M_\alpha \sum_j \nu_{\alpha,j} \, \tilde{r}_j}{\mathcal{D}_{r,\alpha}} \Theta_{\text{cat}}, \tag{3}$$

$$T|_{z=0} = T_{\text{in}} \, , \qquad \frac{\partial T}{\partial r}\bigg|_{r=0} = 0, \qquad \frac{\partial T}{\partial r}\bigg|_{r=R} = -\frac{\Theta_{\text{cat}}}{\lambda_r} \sum_j (\Delta_R H_j) \tilde{r}_j + \frac{k_w}{\lambda_r} (T_{\text{cool}} - T), \tag{4}$$

where $\nu_{\alpha,j}$ is the stoichiometric coefficient of species α and reaction j, Θ_{cat} stands for the volume to surface area ratio of the catalyst layer; M_α is the molar mass of component α, and T_{cool} stands for the cooling temperature at the outer reactor wall. Since T_{cool} is a function of time and the axial coordinate, the respective control domain would be needlessly large. Instead, we only focus on the cooling temperature at the inlet and outlet sides of the reactor. In between, the cooling temperature

4

Table 1: Reactor specifications.

reactor radius	R	=	0.01	m
reactor length	L	=	15	m
wall thickness	δ	=	0.02	m
catalyst layer thickness	δ_{cat}	=	0.002	m
catalyst density	ρ_{cat}	=	2355.2	kg/m^3
catalyst volume-surface ratio	Θ_{cat}	=	0.44	m
heat transfer coefficient	k_w	=	120	W/(m^2 K)

is assumed to obey a linear relationship, given by:

$$T_{\text{cool}}(t, z) = \left(1 - \frac{z}{L}\right) T_{\text{cool,in}}(t) + \frac{z}{L} T_{\text{cool,out}}(t).$$

The reaction heat $\Delta_R H_j$ is obtained by the component enthalpies weighted over the respective stoichiometric coefficients. The rate equation $\tilde{r}_j$ is adapted from Xu and Froment (1989), which requires the conversion from kmol/(kg$_{\text{cat}}$ h) to mol/(m$^3_{\text{cat}}$ s) done by

$$\tilde{r}_j = r_j \cdot \rho_{\text{cat}} \cdot \frac{1000}{3600},$$

where r_j is explained more detailed in Appendix A. The wall heat transfer coefficient k_w and the remaining specifications are listed in Tab. 1. For an accurate description of the radial heat and mass transport coefficients, several correlations are taken from the literature. Using binary diffusion coefficients, Kee et al. (2005) provide the correlation for radial gas diffusion coefficients $\mathcal{D}_{r,\alpha}$. The gas heat capacity c_p relies on polynomial correlations for component heat capacities presented in VDI (2010), and the thermal gas conductivity λ_r follows the mixing rule explained in Poling et al. (2001). The dynamic component viscosities and the thermal component conductivities, which are needed for the mixing rule, are again extracted from VDI (2010). More details on the used physical correlations, but explained for a different reactor type, are comprehensively stated in Bremer et al. (2017).

Operation scenarios

Referring to our exemplary process route for renewable energy conversion, two major scenarios are selected to investigate the properties and performance of the reduced-order models: A disturbed steady-state operation (scenario 1) and the reactor start-up (scenario 2). Both scenarios are charac-

Table 2: Reactor operation parameters

superficial gas velocity	v_z	=	0.5	m/s
inlet gas temperature	T_{in}	=	500	K
inlet pressure	p_{in}	=	$5 \cdot 10^5$	Pa
H$_2$ inlet mole fraction range	$x_{H_2,in}$	$\in$	[0.7 0.9]	
CO$_2$ inlet mole fraction range	$x_{CO_2,in}$	$\in$	[0.1 0.3]	
remaining components inlet mole fraction	$x_{\alpha,in}$	=	10^{-5}	
inlet cooling temperature range	$T_{cool,in}$	$\in$	[600 650]	K
outlet cooling temperature range	$T_{cool,out}$	$\in$	[650 700]	K
start-up time horizon	t_f	=	100	s

terized by operation parameters highlighted in Tab. 2.

Scenario 1 initially deals with a reactor operating at steady-state, which is characterized by $x_{H_2,in} = 0.8$, $x_{CO_2,in} = 0.2$ and $T_{cool,in} = T_{cool,out} = 650\,K$. This steady-state scenario is then disturbed by step changes of hydrogen and carbon dioxide inlet mole fractions, as well as cooling temperatures (see ranges in Tab. 2). In contrast, for scenario 2 the reactor bulk gas phase is considered to be equal to the inlet gas condition. Thus, initially the reactor is neither heated up nor any conversion occurred. In order to fulfill the above mentioned assumption of a fully developed fluid profile, we consider the reactor to be initially perfused with non-reactive gas as best practice, to ensure a safe and fast startup of the entire process system (including axillary units, e.g., compressors and heat exchangers). During the start-up, the four parameters are again disturbed by step changes within the same range as used for scenario 1. Furthermore, the parameter ranges of both scenarios are especially important to bound training and test sets for ROM generation and evaluation, which is explained later in more detail.

Nonlinear State-Space Representation

We discretize the governing PDEs, using the finite volume method (FVM), such that the resulting ODEs approximate the essential dynamic information of the system. Therefore, we use a piecewise constant axial approximation of the convective terms, following the upwind scheme, and a piecewise linear radial approximation for the conductive transport terms with central differences. However, within one finite volume (FV), all physical coefficients (e.g., diffusion coefficient $\mathcal{D}_{r,\alpha}$, thermal conductivity λ_r) are assumed to be constant. We choose $n_z = 25$ axial and $n_r = 25$ radial equally distributed FVs. Consequently, the FVM discretization yields to the following nonlinear system of

order $n = 25 \cdot 25 \cdot 7 = 4375$:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}_1(\boldsymbol{x}(t))\,\boldsymbol{x}(t) + \boldsymbol{A}_2\,\boldsymbol{x}(t) + \boldsymbol{B}_1(\boldsymbol{x}(t))\,\boldsymbol{u}_1(t) + \boldsymbol{B}_2\,\boldsymbol{u}_2(t) + \boldsymbol{f}(\boldsymbol{x}(t)),$$
$$\boldsymbol{x}(t_0) = \boldsymbol{x}_0, \tag{5}$$

where $\boldsymbol{x}(t) \in \mathbb{R}^n$ is the state vector, containing all six component concentrations and temperatures at all finite volumes. The input vector $\boldsymbol{u}_1(t) \in \mathbb{R}^2$ contains the cooling temperatures $T_{\text{cool,in}}$ and $T_{\text{cool,out}}$; the input vector $\boldsymbol{u}_2(t) \in \mathbb{R}^{7n_r}$ relates to the inlet mole fractions at all inlet nodes. The nonlinear function $\boldsymbol{A}_1 : \mathbb{R}^n \to \mathbb{R}^{n \times n}$ originates from the radial gas diffusion, thermal gas conductivity and heat capacity; $\boldsymbol{B}_1 : \mathbb{R}^n \to \mathbb{R}^{n \times 2}$ appears due to thermal gas conductivity at the wall boundary, and $\boldsymbol{f} : \mathbb{R}^n \to \mathbb{R}^n$ mainly contains the reaction rate expressions. The linear parts $\boldsymbol{A}_2 \in \mathbb{R}^{n \times n}$ and $\boldsymbol{B}_2 \in \mathbb{R}^{n \times 7n_r}$ mainly relate to the predefined gas velocity and geometric parameters. This partitioning into nonlinear and linear parts is essential for the MOR, as described in the next section.

3 Model Order Reduction

Our main goal is to compute a low-dimensional models for the catalytic reactor, described in the previous section. The governing equations of the considered catalytic reactor are strongly nonlinear due to the above mentioned physical correlations (e.g., for transport coefficients and rate expressions). For better understanding of the underlying reactor dynamics, we need to discretize the governing PDEs very fine over the spatial domain. This leads to a large-scale nonlinear system of ODEs, whose optimization and control studies impose a huge computational burden. Thus, we seek to determine a ROM describing the dynamics of the FOM with an acceptable accuracy.

Snapshot-based techniques are widely applied for nonlinear systems to determine ROMs and other surrogate models. Without any claim to completeness, many recent publications propose Artificial Neuronal Networks (ANNs), Kriging Models (KM), Gaussian Processes (GP), and Radial Basis Functions (RBFs) for efficient surrogate model construction. However, for dynamic systems based on discretized PDEs, these techniques often suffer from the curse of dimensionality. To address such systems, Proper Orthogonal Decomposition (POD) is efficient (see, e.g., Astrid et al. (2004); Bui-Thanh et al. (2004); Kunisch and Volkwein (2008); Agarwal and Biegler (2013); Li et al. (2014)) and well-established for PDE-constrained optimization problems (see the recent survey Benner et al. (2014)). It relies on the Galerkin projection (see, e.g.,Kunisch and Volkwein (2002)), aiming at

approximating the state vector $\boldsymbol{x}(t) \in \mathbb{R}^n$ by $\boldsymbol{V}\boldsymbol{x}_r(t)$, where $\boldsymbol{x}_r(t) \in \mathbb{R}^r$ is the state vector of the ROM and $\boldsymbol{V}$ is a semi-orthogonal matrix. For simplicity, in the remaining paper we skip the time dependency in vectors or matrices (e.g., we denote $\boldsymbol{x}(t)$ just by $\boldsymbol{x}$). In the following, we outline the basic steps of constructing the ROM for a given nonlinear system via the projection $\mathcal{P} = \boldsymbol{V}\boldsymbol{V}^T$:

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} + \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}) \tag{6}$$

$$\Downarrow$$

Galerkin projection s.t. $\boldsymbol{x} \approx \boldsymbol{V}\boldsymbol{x}_r$

$$\Downarrow$$

$$\dot{\boldsymbol{x}}_r = \boldsymbol{A_r}\boldsymbol{x}_r + \boldsymbol{B_r}\boldsymbol{u} + \boldsymbol{g_r}(\boldsymbol{x}_r, \boldsymbol{u}), \tag{7}$$

where the reduced matrices $\boldsymbol{A}_r$, $\boldsymbol{B}_r$ and $\boldsymbol{g}_r$ are $\boldsymbol{V}^T\boldsymbol{A}\boldsymbol{V}$, $\boldsymbol{V}^T\boldsymbol{B}$ and $\boldsymbol{V}^T\boldsymbol{g}(\boldsymbol{V}\boldsymbol{x}_r, \boldsymbol{u})$, respectively. POD determines the matrix $\boldsymbol{V}$ using the most dominant modes of the dynamical systems. That is, the solution of the FOM (6) at a sequence of time instances (snapshots), corresponding to different possible inputs and scenarios are collected into a snapshot matrix $\mathcal{X}$; then the singular value decomposition (SVD) of $\mathcal{X}$ is performed, i.e., $\mathcal{X} = \boldsymbol{U}_x\boldsymbol{\Sigma}_x\boldsymbol{V}_x^T$, where the first r vectors of $\boldsymbol{U}_x$ are the dominant modes, which are called the POD basis modes of $\mathcal{X}$. Then, $\boldsymbol{V}$ of rank r is constructed by the first r POD vectors, i.e., $\boldsymbol{V} := \boldsymbol{U}_x(:, 1:r)$.

Clearly, the dynamics of the ROM (7) lies in an r-dimensional subspace, where $r \ll n$ with n being the dimension of the full order ODE system (FOM). Furthermore, the reduced matrices $\boldsymbol{A}_r$ and $\boldsymbol{B}_r$ in (7) are constant and need to be computed only once which can be done in the off-line phase. Although, the nonlinear term $\boldsymbol{g}_r$ also lies in the r-dimensional subspace, the related computational cost is not reduced. This is due to the fact that we first need to compute the function $\boldsymbol{g}$ in (6) on the full spatial grid, and then prolongate back to the r-dimensional subspace. This process limits the efficiency of MOR.

To overcome the aforementioned issue of computing the nonlinear function in the ROM (7), the discrete empirical interpolation method (DEIM) is proposed by Chaturantabut and Sorensen (2010). This method avoids the computation of the nonlinear function on the full grid; instead, the nonlinear function needs to be evaluated at only a few selected grid points.

According to DEIM, we first collect the snapshots of the nonlinear function at different time-

Algorithm 1 Steps to obtain ROM via POD-DEIM.

1: Run full simulation and collect i snapshots of the state and the nonlinear function into two snapshot matrices:
$$\mathcal{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_i], \qquad \mathcal{G} = [\boldsymbol{g}(\boldsymbol{x}_1, \boldsymbol{u}), \boldsymbol{g}(\boldsymbol{x}_2, \boldsymbol{u}), \ldots, \boldsymbol{g}(\boldsymbol{x}_i, \boldsymbol{u})].$$
2: Perform SVD of snapshot matrix $\mathcal{X}$ so as to obtain POD basis $\boldsymbol{V}$ of the system.
3: Compute the POD basis $\boldsymbol{U}$ of the nonlinear part by performing SVD of $\mathcal{G}$.
4: Determine an index matrix $\boldsymbol{P}$ and approximate the nonlinear term via Eq. (8).
5: Construct ROM (7)-(8), consisting of linear and nonlinear parts.

---

instances corresponding to different possible inputs and scenarios. Then, we perform SVD on the collected snapshot matrix $\mathcal{G}$ to extract the POD basis vectors of $\mathcal{G}$, i.e., $\mathcal{G} = \boldsymbol{U}_g \boldsymbol{\Sigma}_g \boldsymbol{V}_g^T$. Analogous to the SVD of $\mathcal{X}$, the m POD basis vectors of $\mathcal{G}$ can be constructed as $\boldsymbol{U} = \boldsymbol{U}_g(:, 1 : m)$. The DEIM method uses a linear combination of the POD basis vectors in $\boldsymbol{U}$ to approximate the nonlinear function $\boldsymbol{g}$ (i.e., $\boldsymbol{g} \approx \boldsymbol{U}\boldsymbol{c}$). The coefficient vector $\boldsymbol{c}$ is computed via interpolation. The interpolation condition relating $\boldsymbol{g}$ and $\boldsymbol{U}\boldsymbol{c}$ is that these two quantities are equal to each other at the specially selected spatial grid points, which are called interpolation points. Next, we specify the above discussions on the DEIM method into mathematical formulas.

The DEIM method applied to $\boldsymbol{g}(\boldsymbol{V}\boldsymbol{x}_r, \boldsymbol{u})$ in (7) can be mathematically formulated as finding an index matrix $\boldsymbol{P} \in \mathbb{R}^{n \times m}$ such that $\boldsymbol{P}^T \boldsymbol{g}(\boldsymbol{V}\boldsymbol{x}_r, \boldsymbol{u}) = \boldsymbol{P}^T \boldsymbol{U}\boldsymbol{c}$. Assume that $\boldsymbol{P}^T \boldsymbol{U}$ is invertible, so that we obtain $\boldsymbol{c} = (\boldsymbol{P}^T \boldsymbol{U})^{-1} \boldsymbol{P}^T \boldsymbol{g}(\boldsymbol{V}\boldsymbol{x}_r, \boldsymbol{u})$. Finally, we achieve the following approximation of $\boldsymbol{g}(\boldsymbol{V}\boldsymbol{x}_r, \boldsymbol{u})$ via DEIM:

$$\boldsymbol{V}^T \boldsymbol{g}(\boldsymbol{V}\boldsymbol{x}_r, \boldsymbol{u}) \approx \underbrace{\boldsymbol{V}^T \boldsymbol{U}(\boldsymbol{P}^T \boldsymbol{U})^{-1}}_{=:\boldsymbol{Q} \in \mathbb{R}^{r \times m}} \boldsymbol{P}^T \boldsymbol{g}(\boldsymbol{V}\boldsymbol{x}_r, \boldsymbol{u}). \tag{8}$$

The columns of the index matrix $\boldsymbol{P}$ are unit vectors selecting the interpolation indices. This implicates that $\boldsymbol{P}^T \boldsymbol{g}(\boldsymbol{V}\boldsymbol{x}_r, \boldsymbol{u})$ gives a shortened vector extracted from $\boldsymbol{g}$, whose components are just the components of $\boldsymbol{g}$ at the $m \ll n$ spatial grid points selected by $\boldsymbol{P}$.

The matrix $\boldsymbol{Q}$ in the above formulation is constant and can be computed in advance. Hence, the approximation of the term $\boldsymbol{V}^T \boldsymbol{g}(\boldsymbol{V}\boldsymbol{x}_r, \boldsymbol{u})$ can be computed cheaply. However, the next question is how to select these grid points (i.e., $\boldsymbol{P}$) for the given nonlinear function. For this, a greedy procedure is proposed by Barrault et al. (2004), minimizing the function $\|\boldsymbol{g} - \boldsymbol{U}\boldsymbol{c}\|$ at each iteration. For a detailed discussion, we refer to Chaturantabut and Sorensen (2010). In the end, we summarize a brief procedure to compute the ROM of (6) via POD-DEIM in Alg. 1.

Remark 1 *We have shown the procedure to determine the ROM of* (6) *by using a single nonlinear function $\boldsymbol{g}$. However, following similar steps, the nonlinear terms $\boldsymbol{A}_1, \boldsymbol{B}_1$ and $\boldsymbol{f}$ in* (5) *can be treated*

in the exact same manner.

4 Results

Starting with the first step in Alg. 1, we perform 50 different simulations for the continuous operation and start-up scenario, separately. These simulations are based on the training sets of the parameters which are randomly distributed in the parameter ranges of $x_{H_2,in}$, $x_{CO_2,in}$, $T_{cool,in}$, $T_{cool,out}$ in Tab. 2. For the model implementation and simulation, MATLAB® in combination with the `ode15s` solver is used (Shampine and Reichelt, 1997). For an efficient computation, we provide exact Jacobians of the ODE system, reducing the CPU time for the simulations of FOM by factors in $\mathcal{O}(10) - \mathcal{O}(10^2)$. These Jacobians are obtained from ADiGator, which is a source transformation via operator overloading toolbox for the automatic differentiation of mathematical functions written in MATLAB®, developed by Patterson et al. (2013). Once all 50 simulations are available for each scenario, we collect the state vectors and the respective nonlinear vectors at all time instances of each simulation, chosen by the integrators adaptive stepsize controller, into their corresponding snapshot matrices (e.g., $\mathcal{X}$ and $\mathcal{G}$ in Alg. 1):

$$\mathcal{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_i] \in \mathbb{R}^{n \times i}, \qquad\qquad \mathcal{A} = [\boldsymbol{A}_1(\boldsymbol{x}_1)\,\boldsymbol{x}_1, \ldots, \boldsymbol{A}_1(\boldsymbol{x}_i)\,\boldsymbol{x}_i] \in \mathbb{R}^{n \times i}, \quad (9)$$

$$\mathcal{B} = [\boldsymbol{B}_1(\boldsymbol{x}_1)\boldsymbol{u}_{1,1}, \ldots, \boldsymbol{B}_1(\boldsymbol{x}_i)\boldsymbol{u}_{1,i}] \in \mathbb{R}^{n \times i}, \qquad \mathcal{F} = [\boldsymbol{f}(\boldsymbol{x}_1), \ldots, \boldsymbol{f}(\boldsymbol{x}_i)] \in \mathbb{R}^{n \times i}, \quad (10)$$

where i represents the i-th time instance or snapshot associated with one of the operation scenarios.

Considering approximately 200 integration steps for the entire time horizon of each simulation, the snapshot matrices consist of $200 \times 50 = 10'000$ snapshots. Due to our relatively simple random sampling method, we investigated that this large set of snapshots is necessary to capture the most relevant dynamics. Adding further snapshots (via more simulations or more time instances) does not feature a better ROM performance. There are certainly many better sampling strategies (e.g., adaptive or low-discrepancy sampling methods) that could reduce the offline time, and this will be explored for our application in the future. Nevertheless, we do not expect this to have a significant impact on the online performance.
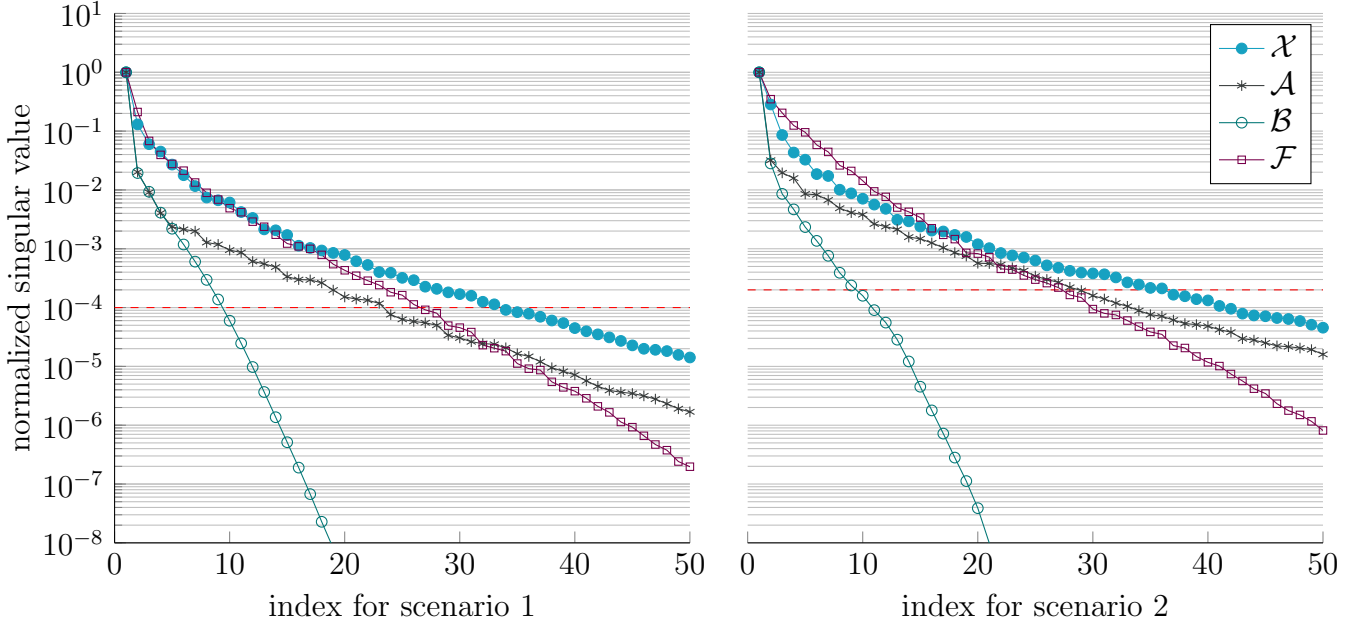
Figure 3: Normalized singular value decay for scenario 1 (continuous operation - $r = 34$, $m_A = 23$, $m_B = 9$, $m_f = 27$) and scenario 2 (start-up - $r = 36$, $m_A = 29$, $m_B = 9$, $m_f = 27$).

Following Alg. 1, we perform the SVD to obtain the relevant bases and index matrices:

$$
\begin{aligned}
\text{svd}(\mathcal{X}) &\xrightarrow{POD} \boldsymbol{V} \in \mathbb{R}^{n \times r}, & \text{svd}(\mathcal{A}) &\xrightarrow{DEIM} \boldsymbol{Q}_A \in \mathbb{R}^{r \times m_A}, \boldsymbol{P}_A \in \mathbb{R}^{n \times m_A}, \\
\text{svd}(\mathcal{B}) &\xrightarrow{DEIM} \boldsymbol{Q}_B \in \mathbb{R}^{r \times m_B}, \boldsymbol{P}_B \in \mathbb{R}^{n_z \times m_B}, & \text{svd}(\mathcal{F}) &\xrightarrow{DEIM} \boldsymbol{Q}_f \in \mathbb{R}^{r \times m_f}, \boldsymbol{P}_f \in \mathbb{R}^{n \times m_f},
\end{aligned}
\tag{11}
$$

where $\boldsymbol{V}$ is the POD basis, and $\boldsymbol{P}_A$, $\boldsymbol{P}_B$, and $\boldsymbol{P}_f$ are the index matrices obtained by applying DEIM to the nonlinearities $\boldsymbol{A}_1$, $\boldsymbol{B}_1$, and $\boldsymbol{f}$, respectively. $\boldsymbol{Q}_A$, $\boldsymbol{Q}_B$, and $\boldsymbol{Q}_f$ correspond to $\boldsymbol{Q}$ in Eq. (8). Furthermore, we plot the singular values for the collected state vectors and nonlinear functions in Fig. 3.

We see that the singular value decay differs significantly between both scenarios. The decay of scenario 2 is much slower than in sceanrio 1, which means that the solution of the FOM corresponding to scenario 2 stays in a subspace with larger dimension than the solution corresponding to scenario 1. Most likely, this behavior is directly connected to the extent of transient state changes within one scenario. Especially during start-up (scenario 2), the system states undergo considerable changes (e.g., due to gas flow heat up and conversion progress). This observation is often associated to transport-dominated systems, which also include our reactive flow problem. Usually these systems are challenging to deal with in POD applications, since their dynamical behavior cannot be captured accurately by a few dominant modes (Reiss et al., 2015). Nevertheless, scenario 1 initially starts with a gas phase fully developed in temperature and concentration, such that the dominance of transport
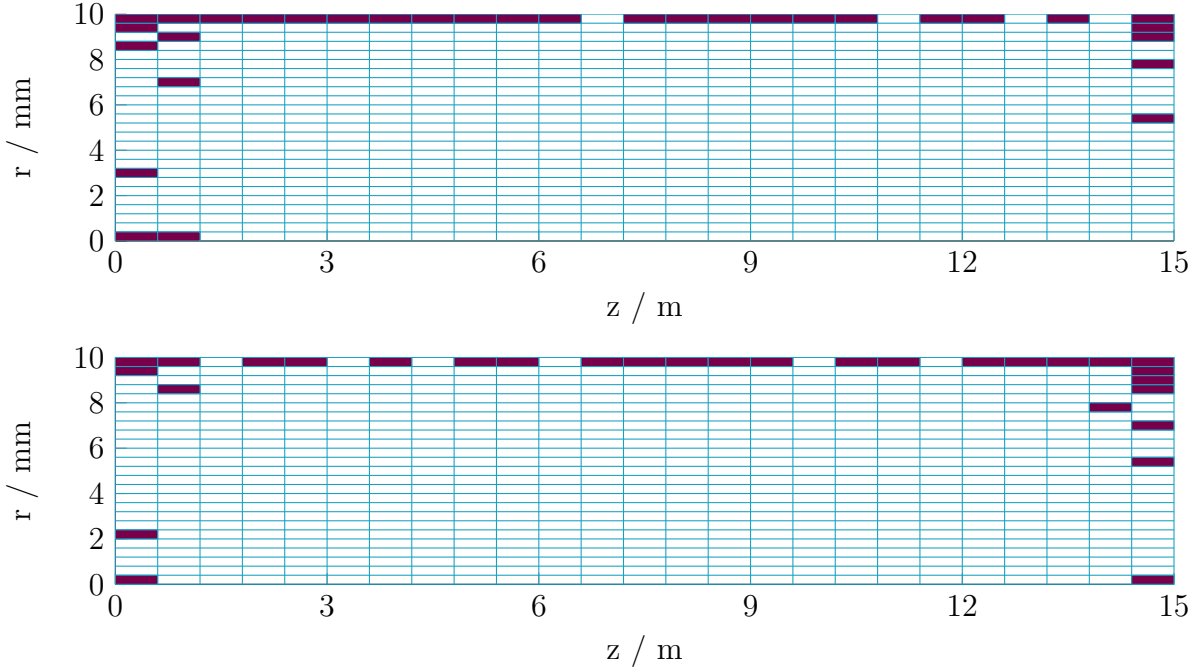
Figure 4: Selected finite volumes (FVs) of the 25×25 spatial grid for the evaluation of nonlinearities. Top: Scenario 1; Bottom: Scenario 2.

effects in Eqs. 1 and 2 is mitigated. As a result, the selection of appropriate dimensions for the projection bases also differs in both scenarios. As illustrated by the red dashed horizontal lines in Fig. 3, we select our bases via a normalized singular value threshold, compromising ROM complexity and accuracy: 10^{-4} (scenario 1) and $2 \cdot 10^{-4}$ (scenario 2).

Beside the orders of the ROMs, both scenarios also vary in their index matrices computed by the DEIM. Since we are dealing with several nonlinear parts, we get different index matrices $\boldsymbol{P}$ for each part. However, our nonlinear equations are often not sufficiently describable by the selected nonlinear indexes. For instance, the heat capacity in one FV can only be evaluated considering all its chemical species and its temperature, which might not all have been fully selected by the index matrices. As a consequence, we merge and extend the index matrices $\boldsymbol{P}_A$, $\boldsymbol{P}_B$ and $\boldsymbol{P}_f$, corresponding to each scenario, to allow for a sufficient evaluation of all relevant nonlinearities. The finally chosen indexes for both scenarios are highlighted in Fig. 4.

We observe that only $\sim 5\,\%$ (32 of 625 - scenario 1, 30 of 625 - scenario 2) of all FVs are relevant for evaluating all nonlinearities. Considering that the computational costs for nonlinearities are the most relevant in case of the FOM evaluation ($\sim 80\,\%$), the reduction due to the DEIM proves to be of great significance.

Finally, we obtain the following ROM of order $r = 36$ (scenario 1) and $r = 34$ (scenario 2):

$$\dot{\boldsymbol{x}}_r = \boldsymbol{Q}_A \boldsymbol{P}_A^T \boldsymbol{A}_1(\boldsymbol{V}\boldsymbol{x}_r)\,\boldsymbol{V}\boldsymbol{x}_r + \boldsymbol{A}_{2r}\boldsymbol{x}_r + \boldsymbol{Q}_B \boldsymbol{P}_B^T \boldsymbol{B}_1(\boldsymbol{V}\boldsymbol{x}_r)\,\boldsymbol{u}_1 + \boldsymbol{B}_{2r}\,\boldsymbol{u}_2 + \boldsymbol{Q}_f \boldsymbol{P}_f^T \boldsymbol{f}(\boldsymbol{V}\boldsymbol{x}_r),$$

$$\boldsymbol{x_r}(t_0) = \boldsymbol{V}^T \boldsymbol{x}_0. \tag{12}$$

Thus, the size of the ROM state vectors $\boldsymbol{x}_r(t) \in \mathbb{R}^{34-36}$ are significantly smaller compared to the FOM state vector $\boldsymbol{x}(t) \in \mathbb{R}^{4375}$. In order to validate the obtained ROMs (12), error bounds are often used for linear or weakly nonlinear systems. Defining an error bound for strongly nonlinear systems is rather difficult and not focus of this study. Nevertheless, there are some studies available (e.g., Zhang et al. (2015); Chaturantabut and Sorensen (2012)), which will be considered in our context in the future. In this study, we validate the obtained ROMs (12) via a numerical study as outlined in the following.

ROM Validation

For validation purposes, we perform further simulations again based on random samples of the parameters $x_{\mathrm{H_2,in}}$, $x_{\mathrm{CO_2,in}}$, $T_{\mathrm{cool,in}}$, $T_{\mathrm{cool,out}}$ in Tab. 2, but different from the training sets, that are used to generate the snapshot matrices. Each simulation is done with the FOM and ROM, respectively. After collecting 20 simulations for each scenario, we compare the results with respect to an absolute transient state error $\boldsymbol{e}(t) \in \mathbb{R}^n$ and a relative time averaged error ε_i of the i-th state

$$\boldsymbol{e} = \boldsymbol{x} - \boldsymbol{V}\,\boldsymbol{x}_r,$$

$$\varepsilon_i = \frac{1}{t_f} \int_0^{t_f} \frac{\|e_i\|}{\|x_i\|}\,dt, \qquad i = 1, \ldots, n.$$

Based on the relative time averaged error ε_i, we define the best and the worst case within each scenario and compare the FOM and ROM simulation with respect to temperature, methane concentration and carbon dioxide concentration. Therefore, we select nodes closest to the catalytic wall at $z/L = 0.3$ and $z/L = 1$.

The results for the best and the worst case of scenario 1 are illustrated in Figs. 5 and 6, respectively. As we see from both cases, the transient behavior is accurately reproduced by the ROM from the initiation of the disturbance (at $t = 0$) until steady state. Furthermore, for all ROM simulations, no
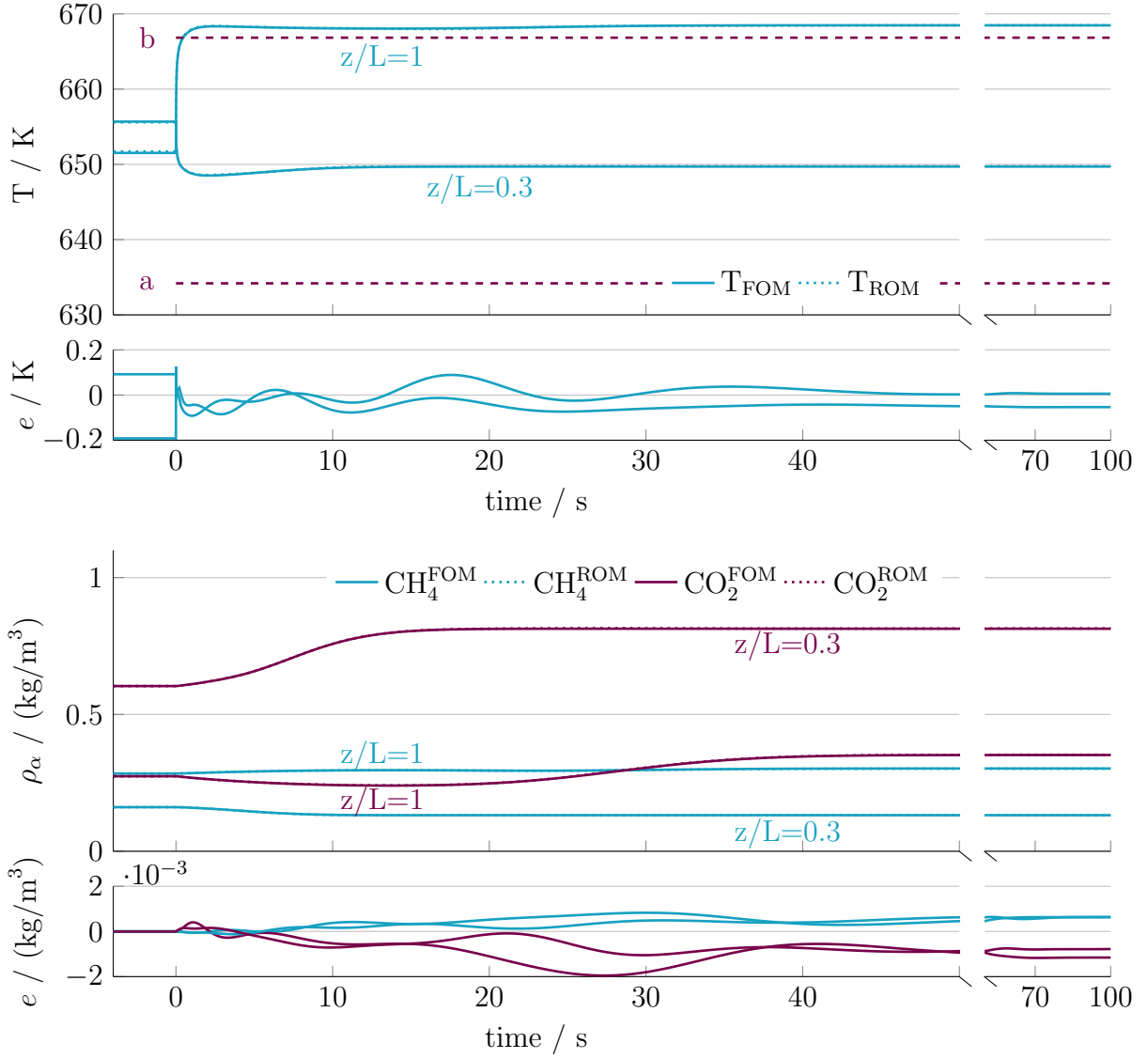
Figure 5: Best test case for FOM and ROM comparison of scenario 1; $x_{H_2,in} = 0.791$, $x_{CO_2,in} = 0.225$, $T_{cool,in} = 634\,K$ (a), $T_{cool,out} = 667\,K$ (b), FOM CPU time $= 13\,s$, ROM CPU time $= 1\,s$.

instabilities are observed, such that the following condition holds:

$$\lim_{t \to \infty} \frac{d\boldsymbol{e}}{dt} \approx 0. \tag{13}$$

This condition turned out to be very sensitive with respect to our selected singular value threshold. It is observed that choosing lower thresholds leads to violations of the condition (13). However, higher thresholds inevitably involve a loss of ROM precision. Besides, we see that as more severe the disturbance is, the less accurate the ROM becomes. Indeed, these low-dimensional models are restricted to the given parameter ranges in Tab. 2, such that using the ROM for other parameter ranges might lead to inaccurate solutions. CPU times for both cases reveal the major advantage of our derived ROMs. We achieve an acceleration of at least one order of magnitude, thanks to the
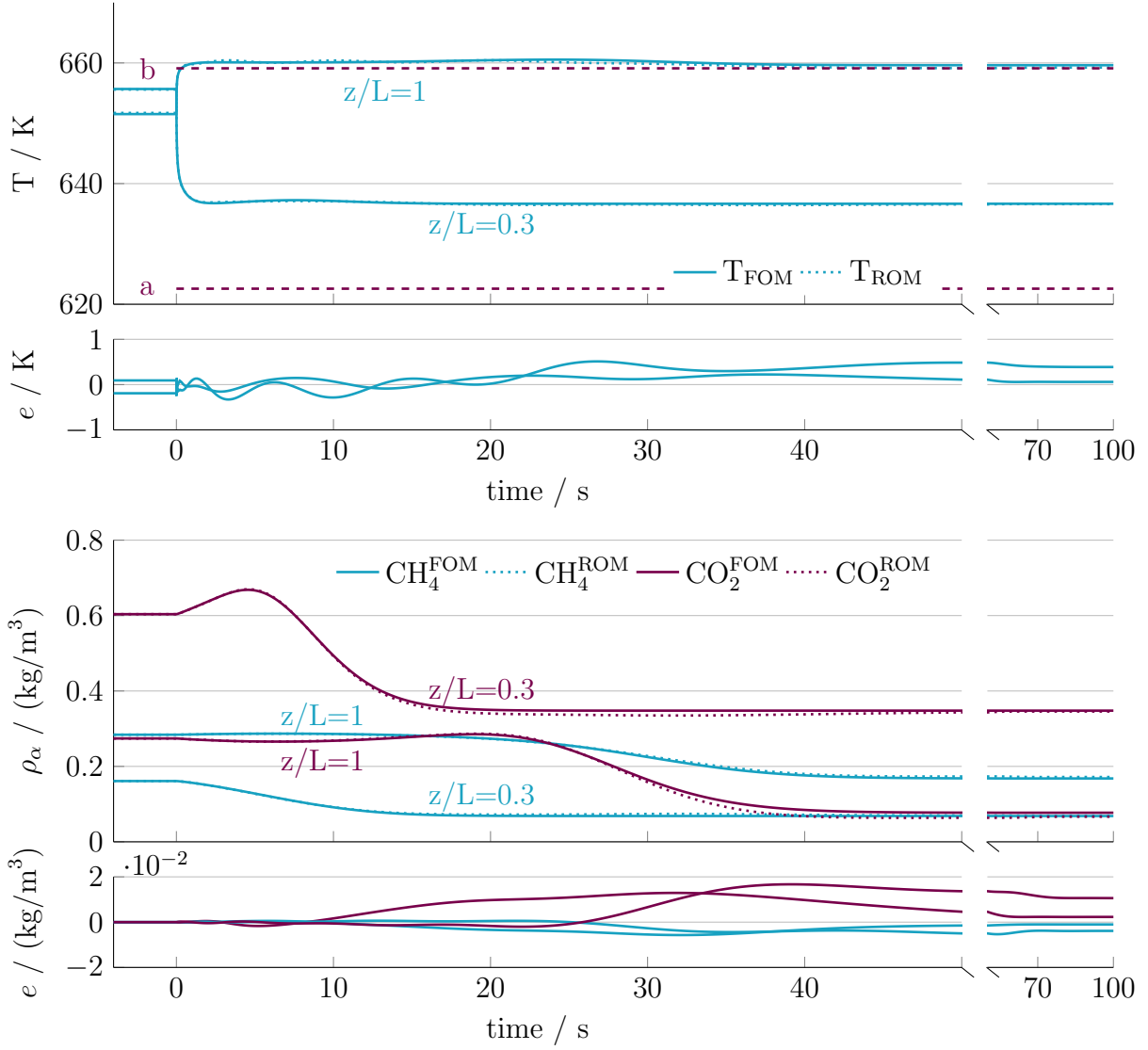
Figure 6: Worst test case for FOM and ROM comparison of scenario 1; $x_{H_2,in} = 0.85$, $x_{CO_2,in} = 0.102$, $T_{cool,in} = 623\,K$ (a), $T_{cool,out} = 659\,K$ (b), FOM CPU time $= 23.5\,s$, ROM CPU time $= 1.5\,s$.

DEIM.

For scenario 2, the best and worst case are illustrated in Figs. 7 and 8. Again, the CPU time needed for simulating the ROM is at least one order of magnitude lower. Furthermore, the ROM also reflects the FOM solution very well, though less accurate. Still no instabilities are observed for any test case, and thus, the condition (13) holds. Similar to scenario 1, more severe disturbances lead to less accurate ROM solutions. However, the absolute error should be treated with caution, since small time delays of steep transient fronts (as seen in Fig. 7) easily cause a larger absolute error at a specific time point, even if the general trend is preserved. Especially in our start-up scenario (scenario 2), these fronts occur very often, due to the fast reaction kinetics evoked by the elevated reactor jacket temperatures. For this reason, the relative time averaged error ε is more relevant, and thus, part of our discussions below.
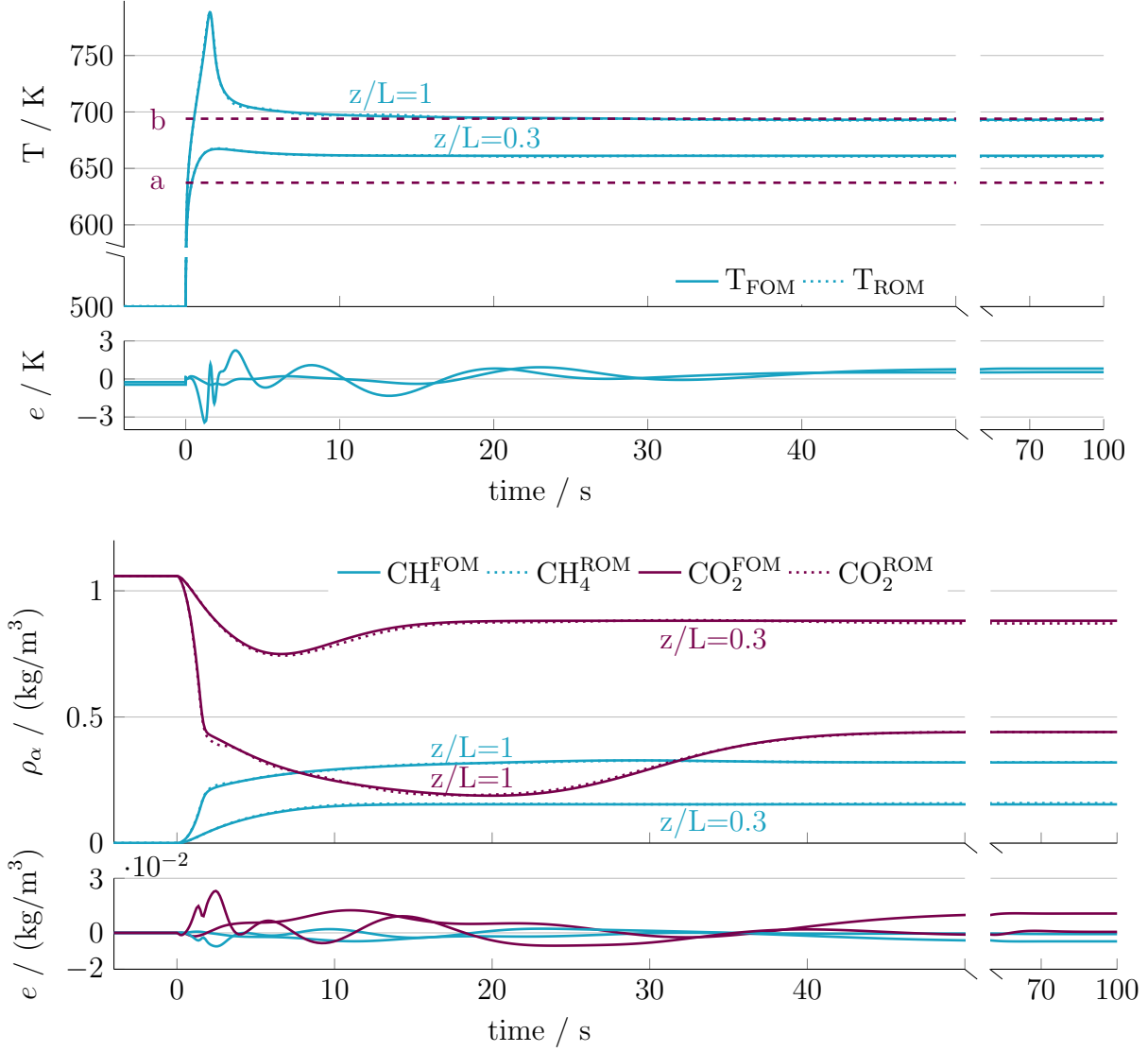
Figure 7: Best test case for FOM and ROM comparison of scenario 2; $x_{H_2,in} = 0.72$, $x_{CO_2,in} = 0.25$, $T_{cool,in} = 637\,K$ (a), $T_{cool,out} = 694\,K$ (b), FOM CPU time $= 49.6\,s$, ROM CPU time $= 2.7\,s$.

To summarize the validation of the ROMs for both scenarios, Fig. 9 contains a statistical survey of all considered test cases. This figure clearly reflects our expectations relating to the singular value decay. Thus, the ROM for scenario 1 is more accurate and more precise. Nevertheless, we see that on average both ROMs perform with an error of less than 4 %. Compared to the FOM uncertainties due to our modeling assumptions, such an error already represents a sufficient ROM reliability. Especially the ROM temperature predictions are highly accurate with error bounds of less than 0.2 %. In contrast, the carbon monoxide concentration reveals the highest errors combined with outliers of above 15 %. However, it is worthwhile mentioning that the reactor is very selective w.r.t methane production due to our operation conditions. Consequently, the by-product carbon monoxide only appears in small quantities, such that the absolute prediction error remains negligible.

Summarizing the discussions from above, Tab. 3 lists the explicit error values together with CPU
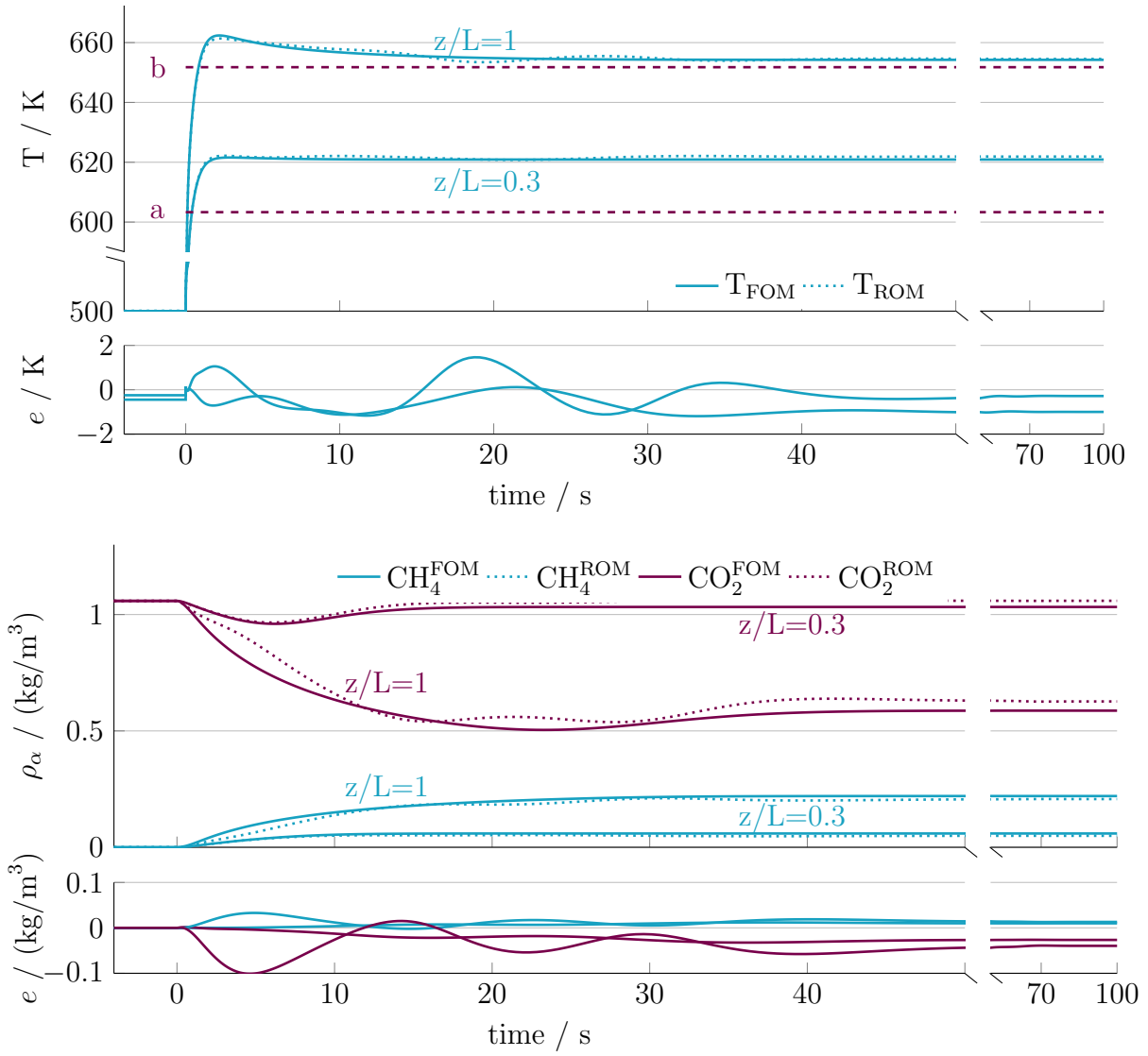
Figure 8: Worst test case for FOM and ROM comparison of scenario 2; $x_{H_2,in} = 0.74$, $x_{CO_2,in} = 0.227$, $T_{cool,in} = 603\,K$ (a), $T_{cool,out} = 652\,K$ (b), FOM CPU time $= 25.9\,s$, ROM CPU time $= 1.7\,s$.
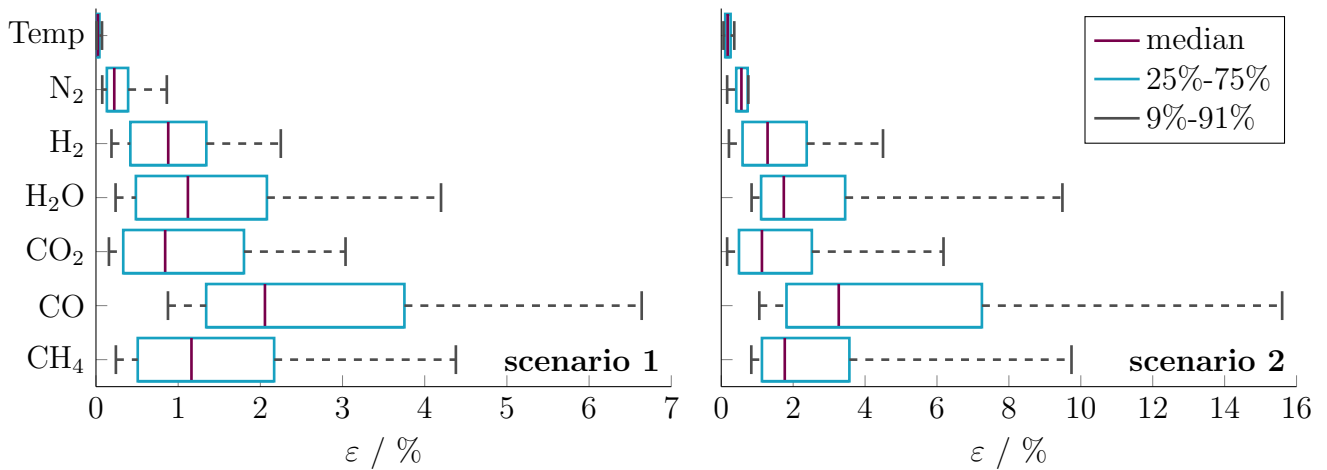

Figure 9: Relative time averaged error sorted by components and temperature; each bar relates to all 25×25 FVs in all 20 test cases.

Table 3: Results of ROM validation considering 20 test cases in scenario 1 (S1) and scenario 2 (S2).

model	no. states	avg. CPU-time	median of ε / %						
			$\bar{\varepsilon}_{CH4}$	$\bar{\varepsilon}_{CO}$	$\bar{\varepsilon}_{CO2}$	$\bar{\varepsilon}_{H2O}$	$\bar{\varepsilon}_{H2}$	$\bar{\varepsilon}_{N2}$	$\bar{\varepsilon}_{T}$
FOM-S1	4375	19.5 s	-	-	-	-	-	-	-
ROM-S1	34	1.3 s	1.16	2.06	0.84	1.12	0.88	0.22	0.02
FOM-S2	4375	39.8 s	-	-	-	-	-	-	-
ROM-S2	36	2.4 s	1.77	3.27	1.13	1.74	1.29	0.56	0.18

times and system orders. Beside the already mentioned ROM accuracy, we also see that the order of the FOM as well as the average CPU time are significantly reduced. As shown above, the biggest contribution to the acceleration goes to the DEIM and the reduced order of the ROM is not the major impact for the computational speed-up. As a matter of fact, without DEIM, the projection of the FOM fully destroys its sparse nature which slightly mitigates the advantage of having a lower-dimensional system, and leads to a ROM which might be even slower than the FOM. Furthermore, a diligent treatment of the nonlinear parts saves the most computational burden. Therefore, our proposed decomposition in several nonlinear parts with separate bases generation appears to be a promising strategy, leading to ROMs 15 to 16 times faster than their original FOMs.

For further improvement, a more sophisticated strategy for snapshot selection during the ROM construction could be very effective to capture the major system dynamics with less sampling effort. Although our random selection for training cases includes already a large number of snapshots, it might still miss important cases. Benner et al. (2015) proposed such an adaptive selection strategy applied to batch chromatography, where they especially report a reduction in offline computation. More relevant in terms of the online computation might be a better balancing of the POD and DEIM bases. Instead of using only one singular value threshold for all bases, a separate treatment for each basis may lead to faster and more accurate ROMs.

5   Conclusion

We have shown that POD-DEIM is a powerful tool to reduce the complexities of highly nonlinear, spatially distributed models for catalytic reactors. The reduced computational time of more than one order of magnitude opens a broad range of new applications for detailed reactor models (e.g., nonlinear model predictive control). Furthermore, we have seen that the performance of POD-DEIM does not exclusively depend on the model itself, but also on the operational conditions (scenarios).

Although our ROMs are capable of capturing the essential system dynamics in pre-defined parameter ranges, they are of different accuracy for different scenarios. Consequently, more expensive and less accurate ROM predictions or even instabilities could arise for certain scenarios. By using our approach for separate consideration of the leading nonlinearities, this tendency could be mitigated. To further address this matter and to improve the efficiency of the ROM, methods using local reduced bases and more sophisticated approaches for snapshot and bases selection will be studied in the future.

6 Acknowledgment

References

T. A. Adams, P. I. Barton, 2009. A dynamic two-dimensional heterogeneous model for water gas shift reactors. International Journal of Hydrogen Energy 34 (21), 8877–8891.

A. Agarwal, L. T. Biegler, 2013. A trust-region framework for constrained optimization using reduced order modeling. Optimization and Engineering 14 (1), 3–35.

P. Astrid, S. Weiland, K. Willcox, T. Backx, Dec 2004. Missing point estimation in models described by proper orthogonal decomposition. In: 2004 43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No.04CH37601). Vol. 2. pp. 1767–1772 Vol.2.

M. Barrault, Y. Maday, N. C. Nguyen, A. T. Patera, 2004. An empirical interpolationmethod: application to efficient reduced-basis discretization of partial differential equations. Comptes Rendus Mathematique 339 (9), 667–672.

U. Baur, P. Benner, L. Feng, 2014. Model order reduction for linear and nonlinear systems: A system-theoretic perspective. Archives of Computational Methods in Engineering 21 (4), 331–358.

P. Benner, L. Feng, S. Li, Y. Zhang, 2015. Reduced-order modeling and ROM-based optimization of batch chromatography. In: Numerical Mathematics and Advanced Applications-ENUMATH 2013. Springer, pp. 427–435.

P. Benner, E. Sachs, S. Volkwein, 2014. Model order reduction for pde constrained optimization. In: G. Leugering, P. Benner, S. Engell, A. Griewank, H. Harbrecht, M. Hinze, R. Rannacher, S. Ulbrich (Eds.), Trends in PDE Constrained Optimization. Springer International Publishing, Basel, pp. 303–326.

L. T. Biegler, Y.-d. Lang, W. Lin, Jan. 2014. Multi-scale optimization for process systems engineering. Computers & Chemical Engineering 60 (0), 17–30.

J. Bremer, P. Goyal, L. Feng, P. Benner, K. Sundmacher, 2016. Nonlinear model order reduction for catalytic tubular reactors. In: 26th European Symposium on Computer Aided Process Engineering. Vol. 38 of Computer Aided Chemical Engineering. Elsevier, pp. 2373 – 2378.

J. Bremer, K. H. G. Rätze, K. Sundmacher, 2017. CO_2 methanation: Optimal start-up control of a fixed-bed reactor for power-to-gas applications. AIChE Journal 63 (1), 23–31.

T. Bui-Thanh, M. Damodaran, K. E. Willcox, 2004. Aerodynamic data reconstruction and inverse design using proper orthogonal decomposition. AIAA journal 42 (8), 1505–1516.

S. Chaturantabut, D. C. Sorensen, 2010. Nonlinear model reduction via discrete empirical interpolation. SIAM Journal on Scientific Computing 32 (5), 2737–2764.

S. Chaturantabut, D. C. Sorensen, 2012. A state space error estimate for pod-deim nonlinear model reduction. SIAM Journal on numerical analysis 50 (1), 46–63.

A. El-Sibai, L. Rihko-Struckmann, K. Sundmacher, 2015. Synthetic methane from CO_2: Dynamic optimization of the Sabatier process for power-to-gas applications. In: 12th International Symposium on Process Systems Engineering (PSE) & 25th European Symposium on Computer Aided Process Engineering (ESCAPE). pp. 1157–1162.

R. Güttel, 2013. Study of unsteady-state operation of methanation by modeling and simulation. Chemical Engineering & Technology 36 (10), 1675–1682.

R. Kee, M. Coltrin, P. Glarborg, 2005. Chemically Reacting Flow: Theory and Practice. Wiley.

K. Kunisch, S. Volkwein, 2002. Galerkin proper orthogonal decomposition methods for a general equation in fluid dynamics. SIAM Journal on Numerical Analysis 40 (2), 492–515.

K. Kunisch, S. Volkwein, 2008. Proper orthogonal decomposition for optimality systems. ESAIM: M2AN 42 (1), 1–23.

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

X. Li, B. Yang, Y. Zhang, 2013. Dynamics and control study on the low temperature methanation reactor with mass and heat recycle. Journal of Process Control 23 (10), 1360–1370.

W. Marquardt, 2002. Nonlinear model reduction for optimization based control of transient chemical processes. In: AIChE Symposium Series. New York; American Institute of Chemical Engineers; 1998, pp. 12–42.

G. Pantoleontos, E. S. Kikkinides, M. C. Georgiadis, 2012. A heterogeneous dynamic model for the simulation and optimisation of the steam methane reforming reactor. International Journal of Hydrogen Energy 37 (21), 16346–16358.

M. A. Patterson, M. Weinstein, A. V. Rao, May 2013. An efficient overloaded method for computing derivatives of mathematical functions in MATLAB. ACM Trans. Math. Softw. 39 (3), 17:1–17:36.

B. E. Poling, J. M. Prausnitz, J. P. O'Connell, 2001. The Properties of Gases and Liquids. McGraw-Hill.

J. Reiss, P. Schulze, J. Sesterhenn, 2015. The shifted proper orthogonal decomposition: A mode decomposition for multiple transport phenomena. arXiv preprint arXiv:1512.01985.

S. Rönsch, J. Schneider, S. Matthischke, M. Schlüter, M. Götz, J. Lefebvre, P. Prabhakaran, S. Bajohr, 2016. Review on methanation - From fundamentals to current projects. Fuel 166, 276 – 296.

L. F. Shampine, M. W. Reichelt, 1997. The MATLAB ODE suite. SIAM Journal on Scientific Computing 18 (1), 1–22.

VDI, 2010. VDI Heat Atlas. VDI-Buch. Springer Berlin Heidelberg.

J. Xu, G. F. Froment, 1989. Methane steam reforming, methanation and water-gas shift: I. intrinsic kinetics. AIChE Journal 35 (1), 88–96.

Y. Zhang, L. Feng, S. Li, P. Benner, 2015. An efficient output error estimation for model order reduction of parametrized evolution equations. SIAM Journal on Scientific Computing 37 (6), B910–B936.

Appendix A

The reaction mechanism for carbon dioxide methanation used in this works follows the Langmuir-Hinshelwood rate equations developed by Xu and Froment (1989) for a nickel-alumina catalyst. These rate equations have been widely used for simulation models (Li et al., 2013; El-Sibai et al., 2015; Rönsch et al., 2016). Accordingly, the rate equations in Eqs. (3) and (4) originate from

$$r_1(T, p_\alpha) = \frac{b_1(T)}{p_{H_2}^{2.5}} \left(p_{CH_4} p_{H_2O} - \frac{p_{H_2}^3 p_{CO}}{K_1(T)} \right) \bigg/ (DEN(T))^2, \tag{14}$$

$$r_2(T, p_\alpha) = \frac{b_2(T)}{p_{H_2}} \left(p_{CO} p_{H_2O} - \frac{p_{H_2} p_{CO_2}}{K_2(T)} \right) \bigg/ (DEN(T))^2, \tag{15}$$

$$r_3(T, p_\alpha) = \frac{b_3(T)}{p_{H_2}^{3.5}} \left(p_{CH_4} p_{H_2O}^2 - \frac{p_{H_2}^4 p_{CO_2}}{K_3(T)} \right) \bigg/ (DEN(T))^2, \tag{16}$$

where p_α is the partial pressure of the respective component in bar, evaluated by

$$p_\alpha = \frac{\rho_\alpha}{\rho} \frac{M}{M_\alpha} p.$$

DEN is a dimensionless parameter defined as

$$DEN(T) = 1 + B_{CO}(T) p_{CO} + B_{H_2}(T) p_{H_2} + B_{CH_4}(T) p_{CH_4} + \frac{B_{H_2O}(T) p_{H_2O}}{p_{H_2}},$$

where B_α is the respective adsorption constant for CH_4, CO, H_2O and H_2 and b_β are rate coefficients following the Arrhenius equation

$$B_\alpha(T) = \mathcal{A}_\alpha \exp \left(-\frac{\Delta H_\alpha}{\mathcal{R}T} \right),$$

$$b_\beta(T) = \mathcal{A}_\beta \exp \left(-\frac{E_\beta}{\mathcal{R}T} \right).$$

Table 4: Kinetic parameters to calculate reaction rates; each pre-exponential factor contains a multiplier of 1.225 (see Xu and Froment (1989))

α	Formula	$\mathcal{A}_\alpha$	$[\mathcal{A}_\alpha]$	ΔH_α	$[\Delta H_\alpha]$	β	$\mathcal{A}_\beta$	$[\mathcal{A}_\beta]$	E_β	$[E_\beta]$
1	CH_4	8.15×10^{-4}	bar^{-1}	-38.28	$\frac{kJ}{mol}$	1	5.176×10^{15}	$\frac{kmol\,bar^{0.5}}{kg_{cat}\,h}$	240.10	$\frac{kJ}{mol}$
2	CO	10.08×10^{-5}	bar^{-1}	-70.65	$\frac{kJ}{mol}$	2	2.395×10^{6}	$\frac{kmol}{kg_{cat}\,h\,bar}$	67.13	$\frac{kJ}{mol}$
4	H_2O	2.17×10^{5}		88.68	$\frac{kJ}{mol}$	3	1.250×10^{15}	$\frac{kmol\,bar^{0.5}}{kg_{cat}\,h}$	243.90	$\frac{kJ}{mol}$
5	H_2	7.50×10^{-9}	bar^{-1}	-82.90	$\frac{kJ}{mol}$					

$\mathcal{A}_\beta$, E_β, $\mathcal{A}_\alpha$, ΔH_α are constant kinetic parameters stated in Tab. 4. The temperature dependent equilibrium constants K_1, K_2 and K_3 are based on Gibbs energy minimization and the law of mass action as shown in Poling et al. (2001).