# MODEL ORDER REDUCTION TECHNIQUE FOR LARGE SCALE FLOW COMPUTATIONS

#### M. Isoz

## Institute of Thermomechanics of the Czech Academy of Sciences, Dolejskova 5, 182 00 Prague, Czech Republic

#### Abstract

Current progress in numerical methods and available computational power combined with industrial needs promote the development of more and more complex models. However, such models are, due to their complexity, expensive from the point of view of the data storage and the time necessary for their evaluation. The model order reduction (MOR) seeks to reduce the computational complexity of large scale models. We present an application of MOR to the problems originating in the finite volume (FV) discretization of incompressible Navier-Stokes equations. Our approach to MOR is based on the proper orthogonal decomposition (POD) with Galerkin projection. Moreover, the problems arising from the nonlinearities present in the original model are adressed within the framework of the discrete empirical interpolation method (DEIM). We provide a link between the POD-DEIM based MOR and OpenFOAM, which is an open-source CFD toolbox capable of solving even industrial scale problems. The availability of a link between OpenFOAM and POD-DEIM based MOR enables a direct order reduction for large scale systems originating in the industrial practice.

**Keywords:** Proper orthogonal decomposition, Discrete empirical interpolation method, Computational fluid dynamics, OpenFOAM.

### 1 Introduction

The modern mathematical models appearing in engineering can impose difficulties when it comes to their numerical solution and day-to-day usability. Because of the advances in numerical mathematics and available computing power, these models tend to be more and more complex. As a result, evaluation of the modern numerical models often requires powerful computers. Furthermore, the calculations are usually expensive from the point of view of the computational time. Nevertheless, industrial practice often calls for parametric studies and optimizations performed using these models.

The model order reduction (MOR) is a useful tool for accelerating calculations connected to parametric studies or optimizations of complex systems. In this paper, we consider MOR based on the proper orthogonal decomposition (POD) with Galerkin projection, which is well described for example by Pinnau [1] or Volkwein [2]. The problems arising from the nonlinearities present in the original model are addressed within the framework of the discrete empirical interpolation method (DEIM) of Chaturantabut and Sorensen [3].

Because of its applicability to a wide range of engineering problems, the POD-DEIM based model order reduction is a quickly developing branch of applied mathematics. As examples of recent developments and applications, we would like to mention the work of Drmac and Gugercin [4] on a priori error bounds for discrete empirical interpolation and the article of Lozovskiy et al. [5] on application of POD based ROM for shallow water flows.

The presented reduced order model (ROM) creation technique represents an *a posteriori* approach to MOR [6]. Hence, the solution of the full order model (FOM) has to be available for the ROM creation. Moreover, not only the solution of the system is necessary for the POD-DEIM based model order reduction, the method also needs an access to several other data structures, which are internal to the CFD solver and which are usually not readily available. This need to easily access the internal CFD solver variables often results in the use of in-house CFD solvers to calculate the FOM results. However, in order to enable a direct order reduction for large scale systems originating in the industrial practice a link between POD-DEIM based MOR and a robust enough CFD solver is needed. In this paper, we provide and validate a link between POD-DEIM based MOR and OpenFOAM [7]. Special emphasis is made on MOR for pressure-linked Navier-Stokes equations.

## 2 Fundamentals of model order reduction

The proper orthogonal decomposition is a projection method for reducing the dimensions of general large-scale ODE systems regardless of their origin [1]. However, within our work we will restrict our interest to the systems obtained from the semi-discretization of time dependent or parameter dependent partial differential equations (PDEs). Furthermore, given our interest in OpenFOAM, which is a finite volume method (FVM) based solver for the problems of the computational fluid dynamics (CFD), we will take a special interest in ROM of the large-scales ODE systems generated by the FV discretization of the Navier-Stokes equations.

A scalar nonlinear PDE for an unknown function  $y: \mathbb{R} \times \mathbb{R}^3 \to \mathbb{R}$  may be rewritten as

$$\dot{y} + \mathcal{L}(t, y) = 0, \qquad (1)$$

where the operator  $\mathcal{L}$  represents all the terms of the original PDE apart from the temporal derivative. After the FV semi-discretization of the equation (1) one obtains the system

$$\Delta\Omega^h \dot{y}^h + \mathcal{L}^h(t, y^h) = 0, \quad y^h : \mathbb{R} \to \mathbb{R}^m,$$
<sup>(2)</sup>

where  $\mathcal{L}^{h}(t, y^{h})$  is the FV spatial discretization operator corresponding to the operator  $\mathcal{L}$  and  $\Delta\Omega^{h} := \operatorname{diag}(\delta\Omega_{i}^{h}) \in \mathbb{R}^{m \times m}$  is a diagonal matrix in which the symbol  $\delta\Omega_{i}^{h}$  represents a volume of one element of the computational domain discretization. Note that from now on, we work solely with  $y^{h}$ , the spatially discretized approximation of y. The superscript h will be omitted in the following.

In the OpenFOAM software, the operator  $\mathcal{L}^{h}(t, y)$  has the structure  $\mathcal{L}^{h}(t, y) = -\tilde{A}(t)y - \tilde{b}(t, y)$ , where the linear, implicitly discretized, members are lumped in the term  $\tilde{A}(t)y$  and the explicitly discretized nonlinearities are used for the construction of the vector  $\tilde{b}(t, y)$ . The size of the matrix  $\tilde{A}$  and of the vector  $\tilde{b}$  is determined by the number of the cells in the FV discretization mesh, m. Because the matrix  $\Delta \Omega^{h}$  is diagonal, its inversion is cheap and equation (2) can be recast as a (large) system of ODEs,

$$\dot{y} = A(t)y + b(t,y), \quad y(0) = y_0, \quad A(t) = (\Delta\Omega^h)^{-1}\tilde{A}(t), \ b(t,y) = (\Delta\Omega^h)^{-1}\tilde{b}(t,y).$$
 (3)

POD is a projection method, its main objective is to find a subspace approximating a given set of data in an optimal least-square sense. In our case, the data is generated by sampling the solution of the full order model (3) at given times,  $\{y_j := y(t_j)\}_{j=1}^n$ ,  $t_j \in (0,T]$ . These samples are called snapshots. The details on the theory of POD may be found for example in [1]. We will restrict our description to a sketch of the process of the reduced order model construction.

Let us denote the space containing the solution of the system (3) and its orthogonal basis as  $V = \operatorname{span}\{\psi_j\}_{j=1}^d$ . Then it is possible to rewrite the solution of (3) as

$$y(t) = \sum_{j=1}^{d} \eta_j \,\psi_j, \,\forall t \in [0,T], \quad \eta_j(t) := \langle y(t), \psi_j \rangle, \quad d = \dim(V),$$

$$\tag{4}$$

where by  $\langle \cdot, \cdot \rangle$  we denote an inner product in the  $L^2$  space. The Fourier coefficients  $\eta_j$ ,  $j = 1, \ldots, d$ , are functions that map [0, T] into  $\mathbb{R}$ .

We arrange the members of the sum in (4) in descending order by the amount of information on the original system they carry, take the first  $l \leq d$  members of and introduce the ansatz

$$y^{\ell}(t) = \sum_{j=1}^{\ell} \eta_j^{\ell} \psi_j, \,\forall t \in [0, T], \quad \eta_j^{\ell}(t) := \langle y^{\ell}(t), \psi_j \rangle, \quad l \le d,$$

$$(5)$$

which is an approximation of y(t) provided  $\ell < d$ . Inserting (5) into (3) and assuming that the equality holds after projection of V on the  $\ell$ -dimensional subspace  $V^{\ell} = \operatorname{span}\{\psi_j\}_{j=1}^{\ell}$  we obtain the system

$$\dot{\eta}^{\ell} = A^{\ell} \eta^{\ell} + f^{\ell}(t, \eta^{\ell}), \,\forall t \in (0, T], \quad \eta^{\ell}(0) = \eta_0^{\ell},$$
(6)

where we defined the reduced system matrix

$$A^{\ell} := (a_{ij}^{\ell}) \in \mathbb{R}^{l \times l}, \quad a_{ij}^{\ell} = \langle A\psi_j, \psi_i \rangle, \tag{7}$$

the ROM nonlinearities  $f^{\ell} = (f_i^{\ell})^{\mathrm{T}} : [0,T] \to \mathbb{R}^{\ell}, f_i^{\ell}(t,\eta) = \left\langle f\left(t, \sum_{j=1}^{\ell} \eta_j \psi_j\right), \psi_i \right\rangle$ , and the initial condition  $\eta^{\ell}(0) = \eta_0^{\ell} = (\langle y_0, \psi_1 \rangle, \dots, \langle y_0, \psi_l \rangle)^{\mathrm{T}}$ . The dimension of the newly defined system (6) is  $\ell \leq d \leq m$ .

The quality of the approximation is largely dependent on the choice of basis functions  $\{\psi_j\}_{j=1}^{\ell}$ . For the sake of brevity, let us only state (the proof may be found in [2]) that the columns of the matrix  $\Psi = [\psi_1, \ldots, \psi_{\ell}] \in \mathbb{R}^{m \times \ell}$  calculated via the POD method are a suitable basis for the discrete representation of the space  $V^{\ell}$ . For a detailed description of POD algorithm, see e.g. [1, 2].

Furthermore, to make ROM completely independent of the full system dimension, it is necessary to address two issues. The first issue is the time dependence of the matrix A, which would cause the need to recalculate the matrix  $A^{\ell}$  for each ROM evaluation.

A way to resolve the time dependence of the matrix A is to sample the system matrices the same way as the full system solution and to interpolate between the full system matrix snapshots. If one uses the linear interpolation, it is possible to write the approximate system matrix as

$$\hat{A}(t) := \varpi(t)A_{i-1} + (1 - \varpi(t))A_i, \quad \varpi(t) = \frac{t - t_{i-1}}{t_i - t_{i-1}}, \quad i = 1, \dots, n.$$
(8)

Substituting the approximation (8) of the matrix A into  $A^{\ell}$  matrix definition (7), we define an approximate time dependent matrix of the reduced system as

$$\hat{A}^{\ell}(t) := \Psi^{\mathrm{T}}\hat{A}(t)\Psi = \varpi(t)\Psi^{\mathrm{T}}A_{i-1}\Psi + (1-\varpi(t))\Psi^{\mathrm{T}}A_{i}\Psi = \varpi(t)A_{i-1}^{\ell} + (1-\varpi(t))A_{i}^{\ell}$$
(9)

and the reduced order model, once it is created, stays fully independent on the full system dimension.

The second problem arises when you look closely at the nonlinearities in (6). One may notice that to evaluate the non-linearity in the reduced order model  $f^{\ell}(t, \eta^{\ell})$ , it is necessary to evaluate the function f at  $(t, y^{\ell})$  and  $y^{\ell}(t) = \sum_{j=1}^{\ell} \eta_j^{\ell}(t) \psi_j \in \mathbb{R}^m$ . This significantly increases the cost of the evaluation of ROM. In this work, we address this problem via the discrete empirical interpolation method of Chaturantabut and Sorensen [3].

DEIM is a combination of the greedy algorithm and POD. The reduction of the computational cost of the system nonlinearity evaluation is achieved by reducing the size of the argument of the function f (assuming it is point-wise evaluable). The details of the procedure may be found for example in the aforementioned article by Chaturantabut and Sorensen. The outputs of DEIM may be used to approximate the nonlinearity in ROM by

$$f^{\ell}(t,\eta^{\ell}) \approx \tilde{f}(t,\eta^{\ell}) := \Psi^{\mathrm{T}} \Phi(P^{\mathrm{T}} \Phi)^{-1} f(t,P^{\mathrm{T}} \Psi \eta^{\ell}), \qquad (10)$$

where the nonlinearity argument  $P^{\mathrm{T}}\Psi\eta^{\ell}$  is in  $\mathbb{R}^p$ ,  $p \leq m$ . We would like also to emphasize that to use DEIM, the nonlinearity samples  $\{f_j := f(t_j, y_j)\}_{j=1}^n$  need to be included in the solution snapshots.

## 3 Reduced order model construction for pressure-linked Navier-Stokes equations

The application of the POD-DEIM based model order reduction to the systems originating in the FV discretization of the incompressible Navier-Stokes equations is not completely straightforward. In the incompressible Navier-Stokes equations,

$$\dot{u} + \nabla \cdot (u \otimes u) - \nabla \cdot (\nu \nabla u) + \nabla p = f, \nabla \cdot u = 0,$$
(11)

the continuity equation  $\nabla \cdot u = 0$  is pressure free. Thus, their discretization ultimately leads to a system of linear algebraic equations of the form,

$$\begin{pmatrix} M & N^T \\ N & 0 \end{pmatrix} \begin{pmatrix} u^h \\ p^h \end{pmatrix} = \begin{pmatrix} f^h \\ 0 \end{pmatrix},$$
(12)

where we denoted the discrete representations of the considered functions by the superscript h. The matrix N coincides with a discrete representation of the  $\nabla$  operator. The matrix M is slightly more difficult. The Navier-Stokes equations for an incompressible isothermal flow (11) are nonlinear. Hence, the nonlinear convective term  $\nabla \cdot u \otimes u$ , needs to be linearized during the construction of the matrix M. If we apply the Newton linearization to the nonlinear convective term,

$$\nabla \cdot u^{j} \otimes u^{j} \approx u^{j-1} \nabla u^{j} + u^{j} \nabla u^{j-1}, \quad j \dots \text{ current time step/iteration},$$
(13)

we can define a linear operator

$$\mathcal{M}(u^{j-1}, u^j) := \dot{u}^j + \nabla \cdot (\nu \nabla u^j) + \mathcal{P}(u^{j-1}, u^j), \qquad (14)$$

where  $\mathcal{P}$  represents the Newton linearization operator. Then, the matrix M is a discrete representation of the operator  $\mathcal{M}$ .

The matrices M and N are, as results of the FV discretization, large and sparse. The system (12) is a so called saddle point problem and as such, it cannot be directly solved by the available methods of numerical linear algebra.

However, if one assumes the matrix M to be regular, it is possible to explicitly express the velocity from the first row of the system (12) and to substitute for it in the second row. Doing so, the following system for one unknown  $p^h \in \mathbb{R}^m$  is obtained

$$NM^{-1}N^{\mathrm{T}}p^{h} = NM^{-1}f, \quad u^{h} = M^{-1}\left(f - N^{\mathrm{T}}p^{h}\right).$$
 (15)

Nevertheless, as M is a large sparse matrix, its inverse is usually not obtainable and the system (15) needs to be solved iteratively by alternatively updating the values of  $p^h$  and  $u^h$  (see e.g. [8] or any description of SIMPLE or PISO algorithms).

The resulting iteration scheme for solution of pressure-linked Navier-Stokes equations usually has the following structure,

$$ND^{-1}N^{\mathrm{T}}p^{h,n} = ND^{-1}(f - (U+L)u^{h,o}),$$
  
$$u^{h,n} = D^{-1}\left(f - (U+L)u^{h,o} - N^{\mathrm{T}}p^{h,n}\right)$$
(16)

where we denoted the current iteration by the superscript o and the following iteration by n. For the purposes of the iterations (16), the matrix M was split as M = D + L + U, where D, L and U are diagonal, lower triangular and upper triangular matrices, respectively.

The reduced order model generated from the solution to (11) computed by any solver based on a SIMPLE-like pressure-velocity coupling (16) needs to take into account the above outlined solution procedure. The natural variable for the solution techniques for the incompressible Navier-Stokes equations based on the solution of the system (15) is the pressure. Thus, it would seem reasonable to base the reduced order model directly on relations (16).

To define the base system for the construction of ROM for the pressure we propose to split the operator  $\mathcal{M}$  introduced in (14) as

$$\mathcal{M} = \mathcal{M}_t + \mathcal{M}_x, \quad \mathcal{M}_t(u^j) := \dot{u}^j, \quad \mathcal{M}_x(u^{j-1}, u^j) = \nabla \cdot (\nu \nabla u^j) + \mathcal{P}(u^{j-1}, u^j).$$
(17)

Furthermore, we split also the matrix M obtained by the FV discretization of  $\mathcal{M}$ ,

$$M = M_t + M_x = M_t + D_x + L_x + U_x.$$
(18)

We propose to define the system in the form (3) as

$$\dot{p} \approx A(t)p + b(t,p), \quad A(\mu) := ND_x^{-1}N^{\mathrm{T}}, \ b(t,p) := ND_x^{-1}(f - (U_x + L_x)u^h),$$
(19)

where  $u^h$  is the solution for the velocity obtained from the SIMPLE iterations. Please note, that the pressure derivative is defined only approximately because at the moment we do not have the proof of equivalence. Also let us emphasize that due to the linearization (13), the coefficients of the matrix  $M_x$  are dependent on the current velocity field. Consequently, the matrix A is time dependent.

## 4 Validation of the proposed approach

In order to validate the proposed method, we performed a series of numerical tests. We started with a test of the link between POD-DEIM based MOR and OpenFOAM solver on a system which can be readily expressed in the form (1). Then, we proceeded to the validation of the presented approach to the ROM construction for the incompressible Navier-Stokes equations.

As a problem that can be expressed directly in the form (1) , we selected the advection of passive scalar  $\alpha$  that comes from the Volume-of-fluid (VOF) simulations of multiphase flows [9]. The principle of the VOF method is to solve the Navier-Stokes equations for a single hypothetical fluid with properties changing in the computation domain. The fluid properties are calculated as a weighted average of the properties of the constitutive phases marked by A and B, respectively. The weight function corresponds to the volume fraction of the tracked phase, i.e. A, at a given point in space-time continuum.

More specifically, the used advection equation for the passive scalar  $\alpha$  is

$$\partial_t \alpha + \nabla \cdot (u\alpha) + \nabla \cdot (u_r \alpha (1 - \alpha)) = 0, \qquad (20)$$

where  $u_r$  is the interface compression velocity defined by

$$u_r = u_A - u_B \,. \tag{21}$$

After the spatial discretization of (20), one obtains the system in the form (2),

$$\dot{y}_{\alpha} = -\left(\Delta\Omega^{h}\right)^{-1} \mathcal{L}_{\alpha}^{h} = A_{\alpha}(t)y_{\alpha} + b_{\alpha}(t, y_{\alpha}), \quad \dot{y}_{\alpha} = \partial_{t}\alpha, \quad y_{\alpha} := \alpha,$$
(22)

for which it is possible to construct the reduced order model directly by the methods presented in section 2. Please note that we use a shortened notation. For example the matrix  $A_{\alpha}$  does not explicitly depend on time, but it is a function of the time-dependent velocity field. However, the velocity is not considered a system unknown because we are interested only in the advection of the variable  $\alpha$ .



Figure 1: Qualitative comparison of the results of the CFD simulation and ROM results for the case of a multiphase flow down an inclined plate. The results of the ROM are shown in the middle of the figure. The only variable of interest is the tracked phase volume fraction,  $\alpha$ . Depicted is the gas-liquid interface, which is an isosurface with  $\alpha = 0.5$ . On the right side of the figure, we give a quantitative comparison of CFD and ROM results. The areas of the computational domain in which the difference between the two results was higher than 10% of max ( $\alpha$ ) = 1.0 are displayed. The CFD gas-liquid interface is shown for easier orientation in the figure.

In Fig. 1, we show a qualitative comparison between the results of an OpenFOAM CFD simulation and the ROM for the case of a gravity-driven multiphase flow down a plate inclined by an angle  $\varphi = 60^{\circ}$  to the horizontal. The dimensions of the computational domain are  $6 \times 5 \times 0.7$  cm.

The CFD simulation was performed on approximately 1 million cells. The reduced order model consisted of 46 ordinary differential equations. On an examination of Fig. 1, it can be seen that the

ROM describes well the general features of the flow. However, it tends to smoothen the gas-liquid interface. Nonetheless, there is a rather good qualitative agreement between the ROM and CFD simulation.

To further evaluate on the agreement between the CFD simulation and ROM, we calculated an absolute difference of the fields  $\alpha^{CFD}$  and  $\alpha^{ROM}$ . Because the tracked phase volume fractions are from [0, 1], the calculated absolute difference coincides with the relative difference,

$$\left|\alpha^{CFD} - \alpha^{ROM}\right| = \varepsilon_{\alpha}^{R} := \frac{\left|\alpha^{CFD} - \alpha^{ROM}\right|}{\max \alpha - \min \alpha} \quad \text{as} \quad \alpha \in [0, 1].$$
<sup>(23)</sup>

In the right side of Fig. 1 we depict only the parts of the computational domain in which  $\varepsilon_{\alpha}^{R} \geq 0.1$  as for the most of the domain the difference is substantially lower. It is visible, that the biggest differences between the fields were encountered at the parts of the computational domain with complex gas-liquid interface evolution. Or, in other words, with large gradient of  $\alpha$  function.

To get a notion of the difference of the solutions in the whole computational domain, let us define the spatial average of  $\varepsilon_{\alpha}^{R}$  as,

$$\overline{\varepsilon}_{\alpha}^{R} = \frac{\sum_{i=1}^{m} \varepsilon_{\alpha,i}^{R}}{m}, \qquad (24)$$

where m is the number of cells in the computational mesh. The value  $\overline{\varepsilon}_{\alpha}^{R}$  for the case depicted in Figures 1 is  $\overline{\varepsilon}_{\alpha}^{R} = 0.0088$ .



Figure 2: Evolution of the mesh mean value of difference between the CFD and ROM, denoted as  $\bar{\varepsilon}^R_{\alpha}$ , in time. Please note, that we selected only 25 representative values out of 200 taken snapshots.

In Fig. 2 we show the evolution of  $\overline{\varepsilon}_{\alpha}^{R}$  with simulation time for the studied case. The applied initial condition corresponds to a dry plate. Hence, at the beginning, the simulation is highly dynamic and we would need to use a large number of snapshots to account for this effect. However, at  $t \approx 0.3$  s the flow becomes quasi-periodic and the agreement between the CFD and ROM is significantly improved. Furthermore, comparing Figures 1 and 2, it is possible to conclude that although the local difference between CFD and ROM might go up to 30%, the difference between the  $\alpha$  fields on the whole mesh lies well bellow 5%. Hence, we conclude that the implementation of the link between POD-DEIM based MOR and OpenFOAM was succesfull and the approach is usable for creation of ROMs based on equations expressible in the form (1).

After the validation of the link between the MOR and OpenFOAM, we tested the proposed approach to the ROM creation for pressure-linked incompressible Navier-Stokes equations. As a test case, we selected a creation of the reduced order model for a transient laminar flow in the vicinity of a cylindrical obstacle. Such a flow develops an instability that leads to the formation of the famous von Karman vortex street [10]. A qualitative comparison of the results of the full model (FVM simulation of the full Navier-Stokes equations on approximately 18000 cells) and the created ROM (system of 7 ODEs) is depicted in Fig. 3 on the next page.

In agreement with the definition (23), we define a mesh averaged relative difference of the fields  $y^{CFD}$  and  $y^{ROM}$  as,

$$\overline{\varepsilon}_{y}^{R} = \frac{\sum_{i=1}^{m} \varepsilon_{y,i}^{R}}{m}, \quad \varepsilon_{y,i} = \frac{\left\| y^{CFD} - y^{ROM} \right\|}{\max_{\forall (i,j)} y - \min_{\forall (i,j)} y}, \quad i = 1, 2, \dots, m, \, j = 1, 2, \dots, n \,, \tag{25}$$

where n is the number of available snapshots of the full solution, m is the original model dimension and by  $\max_{\forall (i,j)} y$  and  $\min_{\forall (i,j)} y$  we denote the maximum and minimum value of the corresponding matrix of snapshots,  $Y = [y_1^{CFD}, \ldots, y_n^{CFD}]$ .



Figure 3: Qualitative comparison of the results of the CFD simulation and ROM results for the case of the flow around a cylindrical obstacle (von Karman vortex street). Results of the ROM are depicted in the top part of the figure, results of the full model are depicted in the bottom. The left part of the image is colored according to the pressure field, the right part according to the velocity magnitude.



Figure 4: Evolution of the mesh mean value of difference between the CFD and ROM, denoted as  $\bar{\varepsilon}^R$ , in time. Please note, that only every third measured point is marked by a corresponding symbol.

In Fig. 4 we depict the time dependence of  $\overline{\varepsilon}_p^R$  and  $\overline{\varepsilon}_{\|u\|}^R$ . It can be seen, that the difference between the full order model and reduced order model for pressure, which is our primary variable, oscillates around 0.2% for the whole simulation. The magnitude of the difference between the velocities is slightly larger, around 0.6%. However, this is to be expected because the velocity field  $u^{ROM}$  is obtained from  $p^{ROM}$  via an approximate form of the relation (15). The necessity of approximating the relation (15) and its final form is imposed by the applied method for the finite volume solution of the incompressible Navier-Stokes equations and will not be discussed here. Instead, the interested reader is pointed towards the relevant literature on the topic, e.g. [8, 11].

## 5 Conclusions

We proposed and validated an approach to use the proper orthogonal decomposition and the discrete empirical interpolation method for the model order reduction of systems arising from the finite volume spatial discretization of the incompressible Navier-Stokes equations. The presented approach is specifically designed for the pressure-based Navier-Stokes equations solution methods (e.g. SIMPLE, SIMPLEC or PISO algorithms). We were able to successfully link the proposed method with the OpenFOAM software. In the future, we plan to improve the mathematical background of the proposed approach to the model order reduction of the Navier-Stokes equations. Also, we would like to concentrate on the model order reduction for parametric studies of systems in a steady state. Finally, let us emphasize that all the computations connected to the ROM creation and evaluation are performed independently of the FOM solution. Therefore, all the capabilities of the OpenFOAM library, e.g. parallelization, are available for the FOM evaluation.

#### Acknowledgment

This work was supported by the European Regional Development Fund under Grant No.

 $CZ.02.1.01/0.0/0.0/15_003/0000493$  (Centre of Excellence for Nonlinear Dynamic Behaviour of Advanced Materials in Engineering). Moreover, the author would like to thank Michael Hinze from the University of Hamburg for many helpful discussions on the proper orthogonal decomposition and related topics.

## References

- R. Pinnau. Model reduction via Proper Orthogonal Decomposition. In W. H. A. Schilders, H. A. van der Vorst, and J. Rommes, editors, *Model order reduction: theory, research aspects* and applications, chapter 2, pages 95–109. Springer Berlin Heidelberg, Berlin, Germany, 2008.
- [2] S. Volkwein. *Proper orthogonal decomposition: theory and reduced-order modelling*. University of Konstanz, Konstanz, Germany, 1 edition, 2013.
- [3] S. Chaturantabut and D. C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM J. Sci. Comp.*, 32:2737–2764, 2010.
- [4] Z. Drmac and S. Gugercin. A new selection operator for the discrete empirical interpolation method – improved a priori error bounds and extensions. SIAM J. Sc. Comp., 38:A631–A648, 2016.
- [5] A. Lozovskiy, M. Farthing, C. Kees, and E. Gildin. POD-based model reduction for stabilized finite element approximations of shallow water flows. J. Comp. App. Math., 302:50–70, 2016.
- [6] A. Cosimo, A. Cardona, and S. Idelsohn. General treatment of essential boundary conditions in reduced order morels for non-linear problems. *Adv. Model. and Simul.*, 3:14, 2016.
- [7] OpenCFD. OpenFOAM: The Open Source CFD Toolbox. User Guide Version 1.4, OpenCFD Limited. Reading UK, 2007.
- [8] H. C. Elman and G. H. Golub. Inexact and preconditioned Uzawa algorithms for saddle point problems. SIAM J. Num. Anal., 31:1645–1661, 1994.
- [9] C. W. Hirt and B. D. Nichols. Volume of fluid (VOF) method for thy dynamics of free boundaries. J. Comp. Phys., 39:201–225, 1981.
- [10] T. von Karman. Aerodynamics. First McGraw-Hill paperback edition. McGraw-Hill, United States, 1963. ISBN 07-067602-x.
- [11] F. Moukalled, M. Darwish, and L. Mangani. The finite volume method in computational fluid dynamics: an advanced introduction with OpenFOAM and Matlab. Springer-Verlag, Berlin, Germany, 1 edition, 2016. ISBN 978-3-319-16874-6.