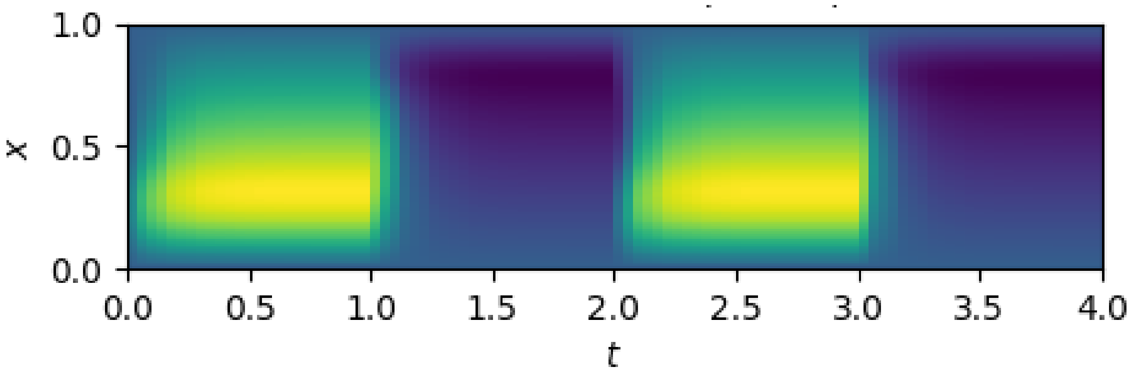
December 12, 2022

ENS Paris-Saclay

Laboratoire de Mécanique Paris-Saclay

Progress report

Space-time goal-oriented error control for incremental proper orthogonal decomposition based reduced order modeling with dual-weighted residuals



Author:
Julian Roth
Hendrik Fischer

Advisors:
Prof. Dr. Thomas Wick
Prof. Dr. Amélie Fau
Prof. Dr. Ludovic Chamoin

Table of Contents

Abstract

A long time ago in a galaxy far, far away, Amelie had an idea...
How to write comments:

Write about the motivation behind this work

- Comments by Hendrik with `\hfischer{some comment}`
- Comments by Julian with `\jroth{some comment}`
- Comments by Amelie with `\afau{some comment}`
- Comments by Ludovic with `\lchamoin{some comment}`
- Comments by Thomas with `\twick{some comment}`

There are also some other useful commands, e.g. you can use `\question{question to HF & JR (or all)}`, which gets a colorful box and shows up in the ToDo-list in the beginning of the PDF file.

This is how to ask questions.

TW for Julian and Hendrik: What is the major novelty in comparison to the literature? I found in Section 3: 'In this work, we focus on another objective, namely the enrichment of the reduced basis depending on the temporal evolution of the quantities of interest.' Could you outline in three bullet points the novelties? Is it space-time with POD? Is it the goal-oriented temporal adaptivity for POD and/or space-time? Is it also fully monolithic settings in case we go to coupled problems such as porous media (see at the end discussions from Amelie and myself)? Is it also the tensor-product space-time ansatz with POD, which is novel? Done

Highlights of our work

- Novel Tensor-Product Space-Time POD-ROM (only similar paper [19]).
- Error controlled POD-ROM by means of basis enrichment utilizing on DWR estimates.
- Incremental on-the-fly basis generation by incremental POD.

1 Problem formulation and discretization

In this work, we consider the abstract parabolic problem

$$\begin{aligned} \partial_t u + \mathcal{A}(u) &= f \qquad \text{in } I \times \Omega, \\ u &= u_D \qquad \text{on } I \times \partial\Omega, \\ u(0) &= u^0 \qquad \text{in } \Omega, \end{aligned} \tag{1}$$

with possibly nonlinear elliptic operator $\mathcal{A}$ and sufficiently regular right-hand side f and initial value u^0. In the problem description, $I := (0, T)$ denotes the temporal domain and $\Omega \subset \mathbb{R}^d$ with $d \in \{1, 2, 3\}$ denotes the spatial domain. Choosing a suitable analytical spatial function space V and analytical temporal functional space X, we can define the continuous spatio-temporal variational formulation as:

Find $u \in u_D + X(V)$ such that

$$A(u)(\varphi) := ((\partial_t u, \varphi)) + ((\mathcal{A}(u), \varphi)) + (u(0), \varphi(0)) = ((f, \varphi)) + (u^0, \varphi(0)) =: F(\varphi) \qquad \forall \varphi \in X(V),$$

where we use the notation

$$(f, g) := (f, g)_{L^2(\Omega)} := \int_\Omega f \cdot g \ \mathrm{d}\boldsymbol{x}, \qquad ((f, g)) := (f, g)_{L^2(I, L^2(\Omega))} := \int_I (f, g) \ \mathrm{d}t.$$

In this notation, $f \cdot g$ represents the Euclidean inner product if f and g are scalar- or vector-valued and it stands for the Frobenius inner product if f and g are matrices.

Remark 1.1. *Some partial differential equations (PDE) which fall into this framework are the heat equation:*

Find $u \in u_D + X(V) := u_D + \{v \mid v \in L^2(I, H^1_0(\Omega)), \partial_t v \in L^2(I, (H^1_0(\Omega))^*)\}$ *such that*

$$A(u)(\varphi) := ((\partial_t u, \varphi)) + ((\nabla_x u, \nabla_x \varphi)) + (u(0), \varphi(0)) = ((f, \varphi)) + (u^0, \varphi(0)) =: F(\varphi) \qquad \forall \varphi \in X(V),$$

and the incompressible isothermal Navier-Stokes equations:

Find $\boldsymbol{U} := \begin{pmatrix} \boldsymbol{v} \\ p \end{pmatrix} \in \begin{pmatrix} \boldsymbol{v}_D \\ 0 \end{pmatrix} + X(V)$ *such that*

$$\begin{aligned} A(\boldsymbol{U})(\boldsymbol{\Phi}) &:= ((\partial_t \boldsymbol{v}, \boldsymbol{\Phi}^v)) - ((p, \nabla_x \cdot \boldsymbol{\Phi}^v)) + \nu((\nabla_x \boldsymbol{v}, \nabla_x \boldsymbol{\Phi}^v)) + (((\boldsymbol{v} \cdot \nabla_x)\boldsymbol{v}, \boldsymbol{\Phi}^v)) + ((\nabla_x \cdot \boldsymbol{v}, \Phi^p)) + (\boldsymbol{v}(0), \boldsymbol{\Phi}^v(0)) \\ &= (\boldsymbol{v}^0, \boldsymbol{\Phi}^v(0)) =: F(\boldsymbol{\Phi}) \qquad \forall \boldsymbol{\Phi} := \begin{pmatrix} \boldsymbol{\Phi}^v \\ \Phi^p \end{pmatrix} \in X(V). \end{aligned}$$

1.1 Tensor-product space-time FEM discretization

We follow our recent work on space-time adaptivity for the Navier-Stokes equations [30] and use tensor-product space-time finite elements (FEM) with discontinuous finite elements in time (dG) and continuous finite elements in space (cG). Using the tensor-product of the temporal and spatial basis functions is a special case of the broad class of space-time finite element methods [23]. We will now explain tensor-product space-time FEM at the example of the heat equation, where the function spaces can be found in [34] and the slabwise tensor-product space-time implementation is being outlined in [36]. However, we assume that the spatial mesh remains fixed, which simplifies the analysis and in the end also the implementation.

1.1.1 Discretization in time

Let $\bar{I} = [0, T] = \{0\} \cup \bigcup_{m=1}^{M} I_m$, with $I_m := (t_{m-1}, t_m]$ be a partitioning of time. Then, we define the semi-discrete space for the heat equation as

$$X_k^{\mathrm{dG}(r)}(V) := \left\{ u_k \in L^2(I, L^2(\Omega)) \,\middle|\, u_k\big|_{I_m} \in P_r(I_m, H_0^1(\Omega)),\ u_k(0) \in L^2(\Omega) \right\}$$

where the space-time function space $X(V)$ has been discretized in time with the discontinuous Galerkin method of order $r \in \mathbb{N}_0$ (dG(r)). A typical choice in our simulation for the temporal degree is $r = 1$. $P_r(I_m, Y)$ is the space of polynomials of order r, which map from the time interval I_m into the space Y. Since functions in $X_k^{\mathrm{dG}(r)}(V)$ can have discontinuities between the time intervals, we define the limits of f_k at time t_m from above and from below for a function f_k

$$f_{k,m}^{\pm} := \lim_{\epsilon \searrow 0} f_k(t_m \pm \epsilon), \tag{2}$$

and the jump of the function value of f_k at time t_m as

$$[f_k]_m := f_{k,m}^{+} - f_{k,m}^{-}. \tag{3}$$

The dG(r) time discretization for the case $r = 1$ is illustrated in Figure 1.

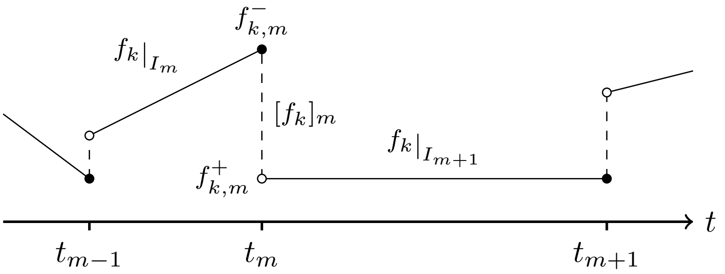


Figure 1: dG(1) time discretization

Note that in Figure 1 the location of the temporal degrees of freedom could be chosen to be at the ends of the time intervals, which would correspond to Gauss-Lobatto quadrature. Nevertheless, due to the discontinuity of the temporal discretization other quadrature rules can be used to define the location of the temporal DoFs. In particular, in our implementation we use Gauss-Legendre quadrature points in time, because we showed in [30] that this yields better results than the Gauss-Lobatto quadrature.

It has been derived in [33] that we obtain the backward Euler scheme by using suitable numerical quadrature for the temporal integrals of the dG(0) discretization. Therefore, it can be seen as a variant of that scheme. Additionally, the corresponding time-stepping scheme of dG(1) has been formulated therein. The dG(r) time discretization of the heat equation with the newly added temporal jump terms reads:

Find $u_k \in u_D + X_k^{\mathrm{dG}(r)}(V)$ such that

$$\begin{aligned} A_k(u_k)(\varphi_k) &:= \sum_{m=1}^{M} \int_{I_m} (\partial_t u_k, \varphi_k) + (\nabla_x u_k, \nabla_x \varphi_k)\ \mathrm{d}t + \sum_{m=0}^{M-1} ([u_k]_m, \varphi_{k,m}^{+}) + (u_{k,0}^{-}, \varphi_{k,0}^{-}) \\ &= ((f, \varphi_k)) + (u^0, \varphi_{k,0}^{-}) =: F_k(\varphi_k) \qquad \forall \varphi_k \in X_k^{\mathrm{dG}(r)}(V). \end{aligned}$$

1.1.2 Discretization in space

For the spatial discretization of the variational formulation, we use a fixed mesh $\mathbb{T}_h$, which consists of intervals in one dimension and in higher dimensions of quadrilateral or hexagonal elements. We can then use element-wise polynomial functions of up to order $s \in \mathbb{N}$ as our spatial function space, i.e.,

$$V_h^s := \left\{ v \in C(\bar{\Omega}) \middle| u\big|_K \in \mathcal{Q}_s(K) \quad \forall K \in \mathbb{T}_h \right\}$$

where $\mathcal{Q}_s(K)$ is being constructed by mapping tensor-product polynomials of degree s from the master element $(0,1)^d$ to the element K. The fully discrete function space for the heat equation is then given by

$$X_k^{\mathrm{dG}(r)}(V_h^s) := \left\{ u_{kh} \in L^2(I, L^2(\Omega)) \,\middle|\, u_{kh}|_{I_m} \in P_r(I_m, V_h^s),\; u_{kh}(0) \in V_h^s \right\}$$

and thus the fully discrete variational formulation reads:

Find $u_{kh} \in u_D + X_k^{\mathrm{dG}(r)}(V_h^s)$ such that

$$\begin{aligned} A_{kh}(u_{kh})(\varphi_{kh}) &:= \sum_{m=1}^{M} \int_{I_m} (\partial_t u_{kh}, \varphi_{kh}) + (\nabla_x u_{kh}, \nabla_x \varphi_{kh}) \,\mathrm{d}t + \sum_{m=0}^{M-1} ([u_{kh}]_m, \varphi_{kh,m}^+) + (u_{kh,0}^-, \varphi_{kh,0}^-) \\ &= (\!(f, \varphi_{kh})\!) + (u^0, \varphi_{kh,0}^-) =: F_{kh}(\varphi_{kh}) \qquad \forall \varphi_{kh} \in X_k^{\mathrm{dG}(r)}(V_h^s). \end{aligned}$$

1.1.3 Slabwise discretization

Finally, we want to remark that the fully discrete variational formulation does not need to be solved on the entire space-time cylinder $I \times \Omega$, but can also be solved sequentially on $L \in \mathbb{N}$ space-time slabs

$$Q_l := \left(\bigcup_{m=m_l}^{m_{l+1}-1} I_m \right) \times \Omega, \qquad 1 = m_1 < m_2 < .. < m_l < ... < m_L < m_{L+1} - 1 = M,$$

see also [36][Remark 2.1]. As mentioned previously, we can then get the space-time FEM basis on Q_l by taking the tensor-product of the spatial and the temporal finite element basis functions. This simplifies the finite element discretization of the abstract parabolic problem (1), since the main prerequisite is a FEM code for the stationary elliptic problem $\mathcal{A}(u) = f \quad \text{in } \Omega$. Furthermore, tensor-product space-time FEM allows for a greater flexibility in the choice of the temporal discretization, since changing the temporal degree of the space-time discretization can be performed simply by changing the polynomial degree of the temporal finite elements. Due to the tensor-product structure of the space-time FE basis, it is straightforward how proper orthogonal decomposition (POD) based reduced order modeling can be performed, since on an abstract level only the spatial finite element basis needs to be replaced by the spatial POD basis.

For the heat equation on the space-time slab $Q_l = (T_{l-1}, T_l) \times \Omega := (t_{m_l - 1}, t_{m_{l+1}-1}) \times \Omega$ with $p := m_{l+1} - m_l$ time intervals, we arrive at the linear equation system

$$\begin{pmatrix} A & & & & \mathbf{0} \\ B & A & & & \\ & B & A & & \\ & & \ddots & \ddots & \\ \mathbf{0} & & & B & A \end{pmatrix} \begin{pmatrix} U_{m_l} \\ U_{m_l+1} \\ U_{m_l+2} \\ \vdots \\ U_{m_{l+1}-1} \end{pmatrix} = \begin{pmatrix} F_{m_l} - BU_{m_l-1} \\ F_{m_l+1} \\ F_{m_l+2} \\ \vdots \\ F_{m_{l+1}-1} \end{pmatrix}$$

or in brevity

$$A_{Q_l} U_{Q_l} = F_{Q_l} \tag{4}$$

with

$$\begin{aligned} A &= C_k \otimes M_h + M_k \otimes K_h, \\ B &= -D_k \otimes M_h, \end{aligned}$$

where we use the spatial matrices

$$\begin{aligned} M_h &= \left\{ (\varphi_h^{(j)}, \varphi_h^{(i)}) \right\}_{i,j=1}^{\#\mathrm{DoFs}(\mathbb{T}_h)}, \\ K_h &= \left\{ (\nabla_x \varphi_h^{(j)}, \nabla_x \varphi_h^{(i)}) \right\}_{i,j=1}^{\#\mathrm{DoFs}(\mathbb{T}_h)} \end{aligned}$$

and the temporal matrices

$$M_k = \left\{ \int_{I_m} \varphi_k^{(j)} \cdot \varphi_k^{(i)} \; \mathrm{d}t \right\}_{i,j=1}^{\#\text{DoFs}(I_m)},$$

$$C_k = \left\{ \int_{I_m} \partial_t \varphi_k^{(j)} \cdot \varphi_k^{(i)} \; \mathrm{d}t + \varphi_{k,m-1}^{(j),+} \cdot \varphi_{k,m-1}^{(i),+} \right\}_{i,j=1}^{\#\text{DoFs}(I_m)},$$

$$D_k = \left\{ \varphi_{k,m-1}^{(j),-} \cdot \varphi_{k,m-1}^{(i),+} \right\}_{i,j=1}^{\#\text{DoFs}(I_m)}.$$

Note that $U_{m_l}, \dots, U_{m_{l+1}-1}$ are space-time vectors themselves, where $U_m \in \mathbb{R}^{\#\text{DoFs}(I_m)\cdot\#\text{DoFs}(\mathbb{T}_h)}$ is the coefficient vector of the solution u on the time interval I_m, i.e., for the dG(r) method in time with Gauss-Legendre quadrature points $t_1, \dots, t_{r+1}$ we have

$$U_{m_l} = \begin{pmatrix} U_{m_l}(t_1) \\ \vdots \\ U_{m_l}(t_{r+1}) \end{pmatrix}.$$

In particular, if we use space-time slabs which contain only one time interval, then we only need to solve the linear system

$$AU_m = F_m - BU_{m-1}$$

for each time interval I_m. For nonlinear problems, like the Navier-Stokes equations, the derivation of the linear equation system is more involved and the matrix A on the diagonal changes in each time interval.

2 Reduced order modeling

2.1 POD-ROM

The increase in computational power in the last decades makes it possible to exploit high performance computing for large numerical simulations. Nevertheless, in some scenarios high performance computing can be computationally expensive, in particular also having a large carbon footprint and enormous energy consumption. These circumstances motivate the application of model order reduction (MOR) techniques on the premise of a large computational speedup to satisfy these demands. In this work, we will mainly deal with projection-based reduced basis methods (RBM) [17, 18, 7, 24, 16, 32, 26] since this methodology aims at efficient treatments by providing both an approximate solution procedure and efficient error estimates [17]. Here, the critical observation is that instead of using projection spaces with general approximation properties (e.g. finite element method) problem-specific approximation spaces are chosen and then can be used for the discretization of the original problem [31]. Based on these spaces and the assumption that the solution evolves smoothly in a low-dimensional solution manifold (equivalent to a small Kolmogorov N-width [20, 6, 18]), a reduced order model (ROM) can be constructed that represents with sufficient accuracy the physical problem of interest using a significantly smaller number of degrees of freedom [31].

In order to construct the reduced spaces, we empirically explore the solution manifold by means of high-fidelity solutions of the full-order model developed in Sec. 1.1. To obtain the reduced basis functions, a proper orthogonal decomposition (POD) [24, 6, 22, 29, 35, 37, 11, 15, 10, 1] is conducted on the snapshots of the high-fidelity solution. The following Theorem 2.1 states that the POD basis is optimal in a least-squares sense. The proof is provided by Gubisch and Volkwein in [14].

Theorem 2.1 (POD basis). *Let $Y = [Y_1, \dots, Y_q] := [U_1(t_1), \dots, U_1(t_{r+1}), U_2(t_1), \dots, U_M(t_{r+1})] \in \mathbb{R}^{n\times q}$ with $q = M \cdot \#DoFs(I_m)$, $n = \#DoFs(\mathbb{T}_h)$ and rank $d \le \min(n, q)$. Moreover, let $Y = \Psi\Sigma\Phi^T$ be its singular value decomposition with $\Sigma = diag(\sigma_1, \dots, \sigma_d) \in \mathbb{R}^{d\times d}$ and orthogonal matrices $\Psi = [\boldsymbol{\psi}_1, \dots, \boldsymbol{\psi}_d] \in \mathbb{R}^{n\times d}$, $\Phi = [\boldsymbol{\phi}_1, \dots, \boldsymbol{\phi}_d] \in \mathbb{R}^{q\times d}$. Then for $1 \le N \le d$ the optimization problem*

$$\min_{\tilde{\boldsymbol{\psi}}_1,\dots,\tilde{\boldsymbol{\psi}}_N \in \mathbb{R}^n} \sum_{j=1}^{q} \left\| Y_j - \sum_{i=1}^{N} \left(Y_j, \tilde{\boldsymbol{\psi}}_i \right)_{\mathbb{R}^n} \tilde{\boldsymbol{\psi}}_i \right\|_{\mathbb{R}^n}^2 \quad s.t. \quad (\tilde{\boldsymbol{\psi}}_i, \tilde{\boldsymbol{\psi}}_j)_{\mathbb{R}^n} = \delta_{ij} \; \forall 1 \le i,j \le N \tag{$\mathbf{P}^N$}$$

is being solved by the left singular vectors $\{\boldsymbol{\psi}_i\}_{i=1}^N$ *and it holds that*

$$\sum_{j=1}^{q}\Big\| Y_j - \sum_{i=1}^{N}(Y_j,\boldsymbol{\psi}_i)_{\mathbb{R}^n}\,\boldsymbol{\psi}_i\Big\|^2_{\mathbb{R}^n} = \sum_{i=N+1}^{d}\sigma_i^2 = \sum_{i=N+1}^{d}\lambda_i. \tag{5}$$

Thus, the decay rate of the singular values plays an essential role in the feasibility of the POD approach. If the sum of the squared truncated singular values is sufficiently small for a relatively small N, we can utilize a linear combination of a few basis functions for a good approximation of elements living in the high-dimensional FE space. Although the error of an obtained rank-N approximation can be determined by Eq. 5, this does not yield an intuitive measure for rank determination. Thus, a widely used criterion to determine the quality of the POD basis heuristically refers to its retained energy or information content $\varepsilon(N)$, cf. [13, 14, 24]. The latter has no physical, energetic character and is defined by

$$\varepsilon(N) = \frac{\sum_{\mathrm{i}=1}^{N}\sigma_i^2}{\sum_{i=1}^{d}\sigma_i^2} = \frac{\sum_{i=1}^{N}\sigma_i^2}{\sum_{i=1}^{\mathrm{q}}||\boldsymbol{u}_i||^2}. \tag{6}$$

Next, the construction of the POD basis is presented. In Algorithm 1, we introduce different approaches dependent on the row to column ratio of the snapshot matrix. For this, we partly rely on the work of Gräßle et al. in [6].

Algorithm 1 POD basis generation in $\mathbb{R}^n$

```
    Input: Snapshots $\{Y_j\}_{j=1}^q \subset \mathbb{R}^n$ and energy threshold $\varepsilon \in [0,1]$.
    Output: POD basis $\{\boldsymbol{\psi}_i\}_{i=1}^N \subset \mathbb{R}^n$ and eigenvalues $\{\lambda_i\}_{i=1}^N$.
Set $Y = [Y_1, \dots, Y_q] \in \mathbb{R}^{n\times q}$.
if $n \approx q$ then
    Compute singular value decomposition $[\Psi, \Sigma, \Phi] = \mathrm{SVD}(Y)$.
    Compute $N = \min\{N \in \mathbb{N} \mid \varepsilon(N) \geq \varepsilon,\ \ 1 \leq N \leq d\}$.
    Set $\lambda_i = \Sigma_{ii}^2$ and $\boldsymbol{\psi}_i = \Psi_{\cdot,i} \in \mathbb{R}^n$ for $1 \leq i \leq N$.
else if $n \ll q$ then
    Compute eigenvalue decomposition $[\Psi, \Lambda] = \mathrm{Eig}(YY^T)$, where $YY^T \in \mathbb{R}^{n\times n}$.
    Compute $N = \min\{N \in \mathbb{N} \mid \varepsilon(N) \geq \varepsilon,\ \ 1 \leq N \leq d\}$.
    Set $\lambda_i = \Lambda_{ii}$ and $\boldsymbol{\psi}_i = \Psi_{\cdot,i} \in \mathbb{R}^n$ for $1 \leq i \leq N$.
else if $q \ll n$ then
    Compute eigenvalue decomposition $[\Phi, \Lambda] = \mathrm{Eig}(Y^TY)$, where $Y^TY \in \mathbb{R}^{q\times q}$.
    Compute $N = \min\{N \in \mathbb{N} \mid \varepsilon(N) \geq \varepsilon,\ \ 1 \leq N \leq d\}$.
    Set $\lambda_i = \Lambda_{ii}$ and $\boldsymbol{\psi}_i = Y\Phi_{\cdot,i}/\sqrt{\lambda_i} \in \mathbb{R}^n$ for $1 \leq i \leq N$.
```

2.2 Tensor-product space-time POD-ROM

In case of our space-time discretization, the spatial FEM spaces V_h^s are replaced by problem-specific low-dimensional spaces $V_N^s = \mathrm{span}\{\varphi_N^1, \dots, \varphi_N^N\}$ obtained by means of POD yielding the reduced variational formulation:

Find $u_{kN} \in u_{N_D} + X_k^{\mathrm{dG}(r)}(V_N^s)$ such that

$$\begin{aligned} A_{kN}(u_{kN})(\varphi_{kN}) &:= \sum_{m=1}^{M}\int_{I_m}(\partial_t u_{kN},\varphi_{kN}) + (\nabla_x u_{kN},\nabla_x\varphi_{kN})\,\mathrm{d}t + \sum_{m=0}^{M-1}([u_{kN}]_m,\varphi_{kN,m}^{+}) + (u_{kN,0}^{-},\varphi_{kN,0}^{-}) \\ &= (\!(f,\varphi_{kN})\!) + (u^0,\varphi_{kN,0}^{-}) =: F_{kN}(\varphi_{kN}) \qquad \forall\varphi_{kN} \in X_k^{\mathrm{dG}(r)}(V_N^s), \end{aligned}$$

with basis functions $\{\varphi_N^1, \dots, \varphi_N^N\}$ and reduced basis matrix

$$Z_N = \begin{bmatrix}\varphi_N^1 & \dots & \varphi_N^N\end{bmatrix} \in \mathbb{R}^{n\times N}. \tag{7}$$

2.2.1 Discretization

For the heat equation on the space-time slab $Q_l = (T_{l-1}, T_l) \times \Omega := (t_{m_l-1}, t_{m_{l+1}-1}) \times \Omega$ with $p := m_{l+1} - m_l$ time intervals, we arrive at the reduced linear equation system

$$\begin{pmatrix} A_N & & & & \mathbf{0} \\ B_N & A_N & & & \\ & B_N & A_N & & \\ & & \ddots & \ddots & \\ \mathbf{0} & & & B_N & A_N \end{pmatrix} \begin{pmatrix} U_{N_{m_l}} \\ U_{N_{m_l+1}} \\ U_{N_{m_l+2}} \\ \vdots \\ U_{N_{m_{l+1}-1}} \end{pmatrix} = \begin{pmatrix} F_{N_{m_l}} - B_N U_{N_{m_{l-1}}} \\ F_{N_{m_l+1}} \\ F_{N_{m_l+2}} \\ \vdots \\ F_{N_{m_{l+1}-1}} \end{pmatrix} \tag{8}$$

or in brevity

$$A_{N,Q_l} U_{N,Q_l} = F_{N,Q_l} \tag{9}$$

with

$$A_N = Z_N^T A Z_N, \tag{10}$$

$$B_N = Z_N^T B Z_N, \tag{11}$$

$$F_{N_{m_l}} = Z_N^T F_{m_l}. \tag{12}$$

3 Error estimator certified reduced order modeling

For further analysis, we consider homogeneous boundary conditions to simplify the presentation. Let a goal functional $J : X(V) + X_k^{\mathrm{dG}(r)}(V) \to \mathbb{R}$ of the form

$$J(u) = \int_0^T J_1(u(t)) \,\mathrm{d}t + J_2(u(T)) \tag{13}$$

be given, which represents our physical quantity of interest. Now, we want to reduce the difference between the quantity of interest of a fine solution u^{fine} and a coarse solution u^{coarse}, i.e.,

$$J(u^{\text{fine}}) - J(u^{\text{coarse}}) \tag{14}$$

subject to the constraint that the variational formulation of the parabolic problem (1) is being satisfied. Possible choices for the fine and the coarse solution could be $u^{\text{fine}} := u \in X(V)$, $u^{\text{coarse}} := u_k \in X_k^{\mathrm{dG}(r)}(V)$ to control the error caused by the temporal discretization or $u^{\text{fine}} := u_k \in X_k^{\mathrm{dG}(r)}(V)$, $u^{\text{coarse}} := u_{kh} \in X_k^{\mathrm{dG}(r)}(V_h)$, with $V_h := V_h^s$ for the heat equation and $V_h := V_h^{s_v/s_p}$ for the Navier-Stokes equations, to control the error caused by the spatial discretization. For more information on space-time error control, we refer the interested reader to [33, 36, 30] and for general information on spatial error control to [4, 5, 2, 12]. However, in this work we will later on restrict ourselves to the control of the error introduced by reduced order modeling and thus we will consider $u^{\text{fine}} := u_{kh}^{\text{FOM}} \in X_k^{\mathrm{dG}(r)}(V_h^{\text{FOM}})$, $u^{\text{coarse}} := u_{kh}^{\text{ROM}} \in X_k^{\mathrm{dG}(r)}(V_h^{\text{ROM}})$, with $V_h^{\text{ROM}} \subset V_h^{\text{FOM}} =: V_h$. First efforts of incorporating the dual-weighted residual (DWR) method in reduced-order modeling have been undertaken by Meyer and Matthies [25], where after computing some snapshots and creating the reduced basis, they used the DWR error estimator to determine which basis vectors have the largest error contribution and only use them for the reduced-order model. This can be thought of as a goal-oriented adaptive coarsening of the reduced basis.

Eventually we can also incorporate this into our project, since it seems straightforward. However, it might not be useful to remove reduced basis vectors too quickly, especially for transport dominated problems with long-term periodic behavior.

In this work, we focus on another objective, namely the enrichment of the reduced basis depending on the temporal evolution of the quantities of interest. This can be thought of as a goal-oriented adaptive refinement of the reduced basis, which we propose to accurately and efficiently compute the solution over the whole temporal domain.

Also explain how our approach is different from PGD and HiMod[28, 3, 27]

3.1 Space-time dual-weighted residual method

For the constrained optimization problem (14), we define the Lagrange functional for the fine problem as

$$\begin{aligned}\mathcal{L}_{\text{fine}} : X_k^{\text{dG}(r)}(V_h^{\text{FOM}}) \times X_k^{\text{dG}(r)}(V_h^{\text{FOM}}) &\to \mathbb{R},\\ (u^{\text{fine}}, z^{\text{fine}}) &\mapsto J(u^{\text{fine}}) - A(u^{\text{fine}})(z^{\text{fine}}) + F(z^{\text{fine}}),\end{aligned}$$

and for the coarse problem as

$$\begin{aligned}\mathcal{L}_{\text{coarse}} : X_k^{\text{dG}(r)}(V_h^{\text{ROM}}) \times X_k^{\text{dG}(r)}(V_h^{\text{ROM}}) &\to \mathbb{R},\\ (u^{\text{coarse}}, z^{\text{coarse}}) &\mapsto J(u^{\text{coarse}}) - A(u^{\text{coarse}})(z^{\text{coarse}}) + F(z^{\text{coarse}}).\end{aligned}$$

The stationary points $(u^{\text{fine}}, z^{\text{fine}})$ and $(u^{\text{coarse}}, z^{\text{coarse}})$ of the Lagrange functionals $\mathcal{L}_{\text{fine}}$ and $\mathcal{L}_{\text{coarse}}$ need to satisfy the Karush-Kuhn-Tucker first-order optimality conditions. Firstly, these stationary points are solutions to the equations

$$\begin{aligned}\mathcal{L}'_{\text{fine},z}(u^{\text{fine}}, z^{\text{fine}})(\delta z^{\text{fine}}) &= 0 \quad \forall \delta z^{\text{fine}} \in X_k^{\text{dG}(r)}(V_h^{\text{FOM}}),\\ \mathcal{L}'_{\text{coarse},z}(u^{\text{coarse}}, z^{\text{coarse}})(\delta z^{\text{coarse}}) &= 0 \quad \forall \delta z^{\text{coarse}} \in X_k^{\text{dG}(r)}(V_h^{\text{ROM}}).\end{aligned}$$

We call these equations the primal problems and their solutions u^{fine} and u^{coarse} the primal solutions. Secondly, the stationary points must also satisfy the equations

$$\begin{aligned}\mathcal{L}'_{\text{fine},u}(u^{\text{fine}}, z^{\text{fine}})(\delta u^{\text{fine}}) &= 0 \quad \forall \delta u^{\text{fine}} \in X_k^{\text{dG}(r)}(V_h^{\text{FOM}}),\\ \mathcal{L}'_{\text{coarse},u}(u^{\text{coarse}}, z^{\text{coarse}})(\delta u^{\text{coarse}}) &= 0 \quad \forall \delta u^{\text{coarse}} \in X_k^{\text{dG}(r)}(V_h^{\text{ROM}}).\end{aligned}$$

These equations are being called the adjoint or dual problems and their solutions z^{fine} and z^{coarse} are the adjoint solutions.

3.1.1 Primal problem

Taking the Gateaux derivatives of the Lagrange functionals $\mathcal{L}_{\text{fine}}$ and $\mathcal{L}_{\text{coarse}}$ with respect to the adjoint solution z, we arrive at the primal problem. Since the variational formulation of the PDE is linear in the test functions, we get

$$\begin{aligned}\mathcal{L}'_{\text{fine},z}(u^{\text{fine}}, z^{\text{fine}})(\delta z^{\text{fine}}) &= -A(u^{\text{fine}})(\delta z^{\text{fine}}) + F(\delta z^{\text{fine}}) = 0 \quad \forall \delta z^{\text{fine}} \in X_k^{\text{dG}(r)}(V_h^{\text{FOM}}),\\ \mathcal{L}'_{\text{coarse},z}(u^{\text{coarse}}, z^{\text{coarse}})(\delta z^{\text{coarse}}) &= -A(u^{\text{coarse}})(\delta z^{\text{coarse}}) + F(\delta z^{\text{coarse}}) = 0 \quad \forall \delta z^{\text{coarse}} \in X_k^{\text{dG}(r)}(V_h^{\text{ROM}}).\end{aligned}$$

We observe that the primal solution can be obtained by solving the original problem, e.g. heat or Navier-Stokes equations, forward in time.

3.1.2 Adjoint problem

Taking the Gateaux derivatives of the Lagrange functionals $\mathcal{L}_{\text{fine}}$ and $\mathcal{L}_{\text{coarse}}$ with respect to the primal solution u, we get

$$\begin{aligned}\mathcal{L}'_{\text{fine},u}(u^{\text{fine}}, z^{\text{fine}})(\delta u^{\text{fine}}) &= J'_u(u^{\text{fine}})(\delta u^{\text{fine}}) - A'_u(u^{\text{fine}})(\delta u^{\text{fine}}, z^{\text{fine}}) = 0 \quad \forall \delta u^{\text{fine}} \in X_k^{\text{dG}(r)}(V_h^{\text{FOM}}),\\ \mathcal{L}'_{\text{coarse},u}(u^{\text{coarse}}, z^{\text{coarse}})(\delta u^{\text{coarse}}) &= J'_u(u^{\text{coarse}})(\delta u^{\text{coarse}}) - A'_u(u^{\text{coarse}})(\delta u^{\text{coarse}}, z^{\text{coarse}}) = 0 \quad \forall \delta u^{\text{coarse}} \in X_k^{\text{dG}(r)}(V_h^{\text{ROM}}).\end{aligned}$$

Hence, to obtain the adjoint solution, we need to solve an additional equation, the adjoint problem

$$A'_u(u)(\delta u, z) = J'_u(u)(\delta u). \tag{15}$$

Note that even for nonlinear PDEs and goal functionals the adjoint problem is linear, since the semilinear form in the variational formulation of the PDE is linear in the test functions.

Remark 3.1. *For linear PDEs, like the heat equation, the left-hand side of the adjoint problem (15) simplifies to*

$$A'_u(u)(\delta u, z) = A(\delta u)(z).$$

For linear goal functionals, like the mean-value functional, the right-hand side of the adjoint problem (15) reduces to

$$J'_u(u)(\delta u) = J(\delta u).$$

In particular for a linear problem, i.e. linear PDE and goal functional, we have the adjoint problem

$$A(\delta u, z) = J(\delta u), \tag{16}$$

which does not depend on the primal solution u anymore.

By Remark (3.1), the adjoint problem for the heat equation reads

$$\begin{gathered} A(\delta u, z) = J'_u(u)(\delta u) \\ \Leftrightarrow \sum_{m=1}^{M} \int_{I_m} (\partial_t \delta u, z) + (\nabla_x \delta u, \nabla_x z) \ \mathrm{d}t + \sum_{m=0}^{M-1} ([\delta u]_m, z_m^+) + (\delta u_0^-(0), z_0^-(0)) = J'_u(u)(\delta u). \end{gathered}$$

We now use integration by parts in time to move the time derivative from the test function δu to the adjoint solution z and we get

$$\sum_{m=1}^{M} \int_{I_m} (\delta u, -\partial_t z) + (\nabla_x \delta u, \nabla_x z) \ \mathrm{d}t - \sum_{m=1}^{M} (\delta u_m^-, [z]_m) + (\delta u_M^-(T), z_M^-(T)) = J'_u(u)(\delta u).$$

For the Navier-Stokes equations the adjoint problem can be derived in a similar fashion as

$$\begin{aligned} \sum_{m=1}^{M} \int_{I_m} & (\boldsymbol{\delta v}, -\partial_t \boldsymbol{Z}^v) - (\delta p, \nabla_x \cdot \boldsymbol{Z}^v) + \nu(\nabla_x \boldsymbol{\delta v}, \nabla_x \boldsymbol{Z}^v) \ \mathrm{d}t \\ & + \sum_{m=1}^{M} \int_{I_m} ((\boldsymbol{\delta v} \cdot \nabla_x)\boldsymbol{v} + (\boldsymbol{v} \cdot \nabla_x)\boldsymbol{\delta v}, \boldsymbol{Z}^v) + (\nabla_x \cdot \boldsymbol{\delta v}, Z^p) \ \mathrm{d}t \\ & - \sum_{m=1}^{M} (\boldsymbol{\delta v}_m^-, [\boldsymbol{Z}^v]_m) + (\boldsymbol{\delta v}_M^-(T), \boldsymbol{Z}_M^{v,-}(T)) = J'_{\boldsymbol{U}}(\boldsymbol{U})(\boldsymbol{\delta U}). \end{aligned}$$

We notice that the adjoint problem now runs backward in time.

3.1.3 Error identity and temporal localization for linear problems

For the sake of simplicity, we assume that we are dealing with a linear PDE and goal functional. Then we have the error identity

$$J(u^{\text{fine}}) - J(u^{\text{coarse}}) = -A(u^{\text{coarse}})(z^{\text{fine}}) + F(z^{\text{fine}}) =: \eta. \tag{17}$$

The proof relies on the linearity of the goal functional and the definition of the adjoint and primal problems:

$$J(u^{\text{fine}}) - J(u^{\text{coarse}}) = J(u^{\text{fine}} - u^{\text{coarse}}) = A(u^{\text{fine}} - u^{\text{coarse}})(z^{\text{fine}}) = -A(u^{\text{coarse}})(z^{\text{fine}}) + F(z^{\text{fine}}).$$

In the DWR literature for spatial and temporal discretization error control this kind of error identity (17) would be useless, because for most applications z^{fine} is not feasible to compute exactly and replacing it by z^{coarse} yields bad error estimates. However, in our case $z^{\text{fine}} := z_{kh}^{\text{FOM}} \in X_k^{\text{dG}(r)}(V_h^{\text{FOM}})$ is the full-order model dual solution, which is computable but a bit expensive. Moreover, in our numerical experiments we will observe that using a reduced-order-model dual solution $z^{\text{coarse}} := z_{kh}^{\text{ROM}} \in$

$X_k^{\mathrm{dG}(r)}(V_h^{\mathrm{ROM}})$ still produces excellent error estimates for our problems, if the dual reduced basis is sufficiently large.

To localize the error in time, we just need to assemble the primal residual (17) slabwise. In particular, to localize the error to each time interval I_m, we simply need to assemble the primal residual on each time interval separately. More concretely, for the heat equation the error on the time interval can be computed from the primal linear equation system, the coarse primal solution and the fine dual solution by

$$\eta|_{I_m} = \sum_{i=1}^{\#\mathrm{DoFs}(I_m)} \left\{ (Z_m^{\mathrm{fine}})^T \left(-AU_m^{\mathrm{coarse}} + F_m - BU_{m-1}^{\mathrm{coarse}} \right) \right\}_i . \tag{18}$$

To test whether we need to use the fine dual solution for our error estimates or whether we can replace it by a coarse dual solution, we use the effectivity index as a measure of the quality of our error estimator. The effectivity index is the ratio of the estimated and the true error, i.e.

$$\mathrm{I}_{\mathrm{eff}} := \left| \frac{\eta}{J(u^{\mathrm{fine}}) - J(u^{\mathrm{coarse}})} \right| .$$

We desire $\mathrm{I}_{\mathrm{eff}} \approx 1$, since then the error estimator can reliably predict the reduced-order-modeling error. In practice, for linear problems, the true error and the estimated error coincide up to a small tolerance caused by rounding errors, thus we can expect that $\mathrm{I}_{\mathrm{eff}} \approx 1$ for such cases.

3.1.4 Space-time dual-weighted residual method for nonlinear problems

This has not been implemented yet and we need to figure out whether the theory here is similar as for nonlinear (in)stationary DWR.

3.2 Error estimator based ROM updates

3.2.1 Incremental Proper Orthogonal Decomposition

This section aims to derive an algorithm that adapts an already existing tSVD or solely its left singular vectors without any additional knowledge about the underlying snapshot matrix. This methodology can then be used to update the POD incrementally. For this purpose, we rely on the general approach of an additive rank-b modification of the SVD, mainly developed by [8, 9] and applied to model-order reduction of fluid flows in [21]. Although this approach provides a variety of possible modifications, e.g. updating and down-dating, modification of individual values or exchanging rows and columns, we are merely interested in the updates of columns, i.e. column extension and thus restrict the proceeding on this. The following steps are oriented on [21][Section 2.2].

We start with a given rank-N truncated SVD $Y \approx USV^{\mathrm{T}}$ of a snaphot matrix $Y \in \mathbb{R}^{\mathrm{n}\times\tilde{\mathrm{q}}}$ with $\tilde{q} \leq q$. Additionally, let a bunch of snapshots $\{Y_1, \dots, Y_b\}$ be stored in the bunch matrix

$$B = \begin{bmatrix} Y_1 & \dots & Y_b \end{bmatrix} \in \mathbb{R}^{n\times b}. \tag{19}$$

We now aim to compute the expanded tSVD

$$\tilde{U}\tilde{S}\tilde{V}^T = \tilde{Y} = \begin{bmatrix} Y & B \end{bmatrix}$$

without explicitly computing Y or $\tilde{Y}$ due to performance and memory reasons. Instead, we perform an additive modification according to Kühl, Fischer and Rung [21] and obtain the rank-$\tilde{N}$ tSVD of $\tilde{Y}$ with

$$\tilde{V} = \begin{bmatrix} V & 0 \\ 0 & I \end{bmatrix} V'(:, 1:\tilde{N}) \tag{20}$$

$$\tilde{S} = S'(1:\tilde{N}, 1:\tilde{N})\tilde{N} \tag{21}$$

$$\tilde{U} = \begin{bmatrix} U & Q_{\mathrm{B}} \end{bmatrix} U'(:, 1:\tilde{N}), \tag{22}$$

where $K = U'S'{V'}^T$ is the SVD of

$$K = \begin{bmatrix} S & U^{\mathrm{T}}B \\ 0 & R_{\mathrm{B}} \end{bmatrix}. \tag{23}$$

and the QR decomposition

$$Q_{\mathrm{B}}R_{\mathrm{B}} = (I - UU^{\mathrm{T}})B. \tag{24}$$

For the POD basis update, we identify U and $\tilde{U}$ with the previous and updated version of the reduced basis matrix Z_N, respectively. We also neglect the update of the right singular values in Eq. 20, since it does not provide any additional benefit apart from extra computational effort. The singular values are considered for the rank determination but they also come within zero computational costs. In conclusion, Eqns. (21)-(23) serve as the basis for the on-the-fly or incrementally computed POD (iPOD) in this paper.

An additional technical observation: For small bunch matrix widths b, the iPOD algorithm is invoked frequently, and algebraic subspace rotations involved possibly do not preserve orthogonality. Hence, a numerically induced loss of orthogonality of the POD basis vectors can occur. In order to deal this problem an additional orthonormalization of $\begin{bmatrix} U & Q_{\mathrm{B}} \end{bmatrix}$ is recommended in this case. The following algorithm drafts the implementation of an incremental POD update.

Algorithm 2 Incremental POD update

Input: Reduced basis matrix $Z_N \in \mathbb{R}^{n \times N}$, singular value vector $\Sigma = [\sigma_1, \dots, \sigma_N] \in \mathbb{R}^N$, bunch matrix $B \in \mathbb{R}^{n \times b}$, and energy threshold $\varepsilon \in [0, 1]$.

Output: Reduced basis matrix $Z_N \in \mathbb{R}^{n \times \tilde{N}}$, singular value vector $\Sigma = [\sigma_1, \dots, \sigma_{\tilde{N}}] \in \mathbb{R}^{\tilde{N}}$

```
$M = Z_N^T B$
$P = B - Z_N M$
$[Q_P,\ R_P] = \mathrm{QR}(P)$
$Q = [Z_N\ Q_p]$
$K = \begin{bmatrix} \mathrm{diag}(\Sigma) & M \\ 0 & R_P \end{bmatrix}$
if Q not orthogonal then
    $[Q,\ R] = \mathrm{QR}(Q)$
    $K = RK$
$[U', \Sigma'] = \mathrm{SVD}(K)$
$\tilde{N} = \min\{N \in \mathbb{N} \mid \varepsilon(N) \geq \varepsilon,\quad 1 \leq N \leq d\}$
$\Sigma = \mathrm{diag}(\Sigma')(1 : \tilde{N})$
$Z_N = QU'(:, 1 : \tilde{N})$
```

3.2.2 Certified incremental ROM

In this section, we combine the space-time ROM presented in section 2 and the incremental POD of section 3.2.1 with the findings on goal-oriented error control of section 3. In our work, the concrete idea is to identify the fine and coarse solution introduced in the DWR method with the finite element and reduced basis solution, respectively. We aim to solve the reduced system without any prior knowledge or exploration of the solution manifold. Instead, we solve from the start on the ROM and –if necessary– update the reduced basis during the simulation with finite element solutions by means of the iPOD. For this purpose, we estimate the error in cost functional and initiate an update if the estimate exceeds a given threshold. Therefore, we are able to reduced the full-order solves to a minimum. This approach yields a certified reduced basis method with on-the-fly basis generation. Further, it allows an adaptive enrichment of the reduced basis to unforeseen changes in the solution behavior.

Algorithm 3 Certified incremental ROM

```
Input: Initial condition $U_0$, $Z_N = []$, $\Sigma = []$, time steps $M$, energy threshold $\varepsilon \in [0,1]$ and error tolerance $\mathrm{tol} > 0$.
Output: Reduced basis matrix $Z_N \in \mathbb{R}^{n \times N}$, singular value vector $\Sigma = [\sigma_1, \dots, \sigma_N] \in \mathbb{R}^N$
Compute adjoint solutions $Z_{Q_l}$ of 16 for $1 \leq l \leq L$
$Z_N, \Sigma = \mathrm{iPOD}(Z_N, \Sigma, [U^0], \varepsilon)$
Compute reduced system matrix $A_{N,Q_l}$ of Eq. 9
Compute reduced boundary matrix $B_{N,Q_l}$ of Eq. 9
for l = 1, 2, ..., L do
    Compute reduced RHS $F_{N,Q_l}$ of Eq. 9
    Solve reduced-order system $A_{N,Q_l} U_{N,Q_l} = F_{N,Q_l}$ of Eq. 9
    $U_{Q_l}(t_i) = Z_N U_{N,Q_l}(t_i)$ for $1 \leq i \leq r+1$
    Compute the error estimate $\eta|_{I_m}$ of Eq. 18
    if $\eta|_{Q_l} > \mathrm{tol}$ then
        Solve full-order system $A_{Q_l} U_{Q_l} = F_{Q_l}$ of Eq. 4
        $Z_N, \Sigma = \mathrm{iPOD}(Z_N, \Sigma, [U_{Q_l}(t_1), \dots, U_{Q_l}(t_{q+1})], \varepsilon)$
        $U_{N,Q_l}(t_i) = Z_N^T U_{Q_l}(t_i)$ for $1 \leq i \leq r+1$
        Update reduced system matrix $A_{N,Q_l}$ of Eq. 9
        Update reduced boundary matrix $B_{N,Q_l}$ of Eq. 9
```

Note that similar to the mere approximation error of the POD in Eq. 5 the interpretation of the error estimate is not intuitive. Therefore, here again we are looking for a relative measurement of the approximation quality. However, the full-order solution is not available for a normalization such that we resort to $J(u)|_{Q_l} \approx J(u_N)|_{Q_l} + \eta|_{Q_l}$. This results in the relative error estimator $\eta_{rel}|_{Q_l}$ defined by

$$\eta_{rel}|_{Q_l} = \left| \frac{\eta|_{Q_l}}{J(u)|_{Q_l}} \right| \approx \left| \frac{\eta|_{Q_l}}{J(u_N)|_{Q_l} + \eta|_{Q_l}} \right|. \tag{25}$$

4 Numerical tests

We start off with two numerical tests in one spatial dimension on which we will show the efficiency of our proposed method. We consider $\Omega = (0,1)$ and $I = (0,4)$. We use a zero initial condition, homogeneous Dirichlet boundary conditions and the mean value goal functional $J(u) := \int_0^1 \int_0^{\frac{1}{2}} u(t,x)\ \mathrm{d}x\ \mathrm{d}t$.

For the reduced-order model, we choose that the reduced basis has to preserve $\varepsilon = 99.99\%$ of the information. As previously stated, we resort to the relative error estimate $\eta_{rel}|_{Q_l}$ and allow errors up to $\mathrm{tol} = 1\%$. The full-order model is characterized by $4,097$ and $1,024$ DoFs in space and time, respectively. This gives us a total of $4,195,328$ space-time degrees of freedom. Further, the temporal domain is split up to 512 time slabs containing 2 temporal DoFs each.

TW: Please add a few sentences to the software: purely deal.II? How is the SVD done, also in deal.II?

4.1 Two alternating heat sources

For our first numerical test, we use a heat source which alternates between heating in the subdomain $(\frac{1}{8}, \frac{3}{8}) \subset \Omega$ and cooling in the subdomain $(\frac{6}{8}, \frac{7}{8}) \subset \Omega$ each second. For this we use the right-hand side function

$$f(t,x) := \begin{cases} 0.2 & x \in (\frac{1}{8}, \frac{3}{8}),\ t \in (0,1) \cup (2,3) \\ -0.2 & x \in (\frac{6}{8}, \frac{7}{8}),\ t \in (1,2) \cup (3,4) \\ 0 & \text{else} \end{cases}.$$

In Figure 2, we present the reduced-order space-time solution u_N which is solved utilizing the previously introduced certified incremental ROM (iROM). Additionally, we plot the true error $u_h - u_N$ between the full-order space-time solution u_h and the reduced-order space-time solution u_N.

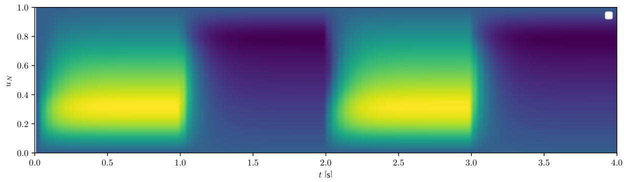


Figure 2: Temporal evolution of the reduced order solution u_N and the true error $u_h - u_N$ for two alternating heat sources.

In Figure 3, the temporal error estimates on each slab are shown. Note, we use slabs consisting of one time interval each. We indicate full-order solves in red color and reduced-order solves in blue. We observe that full-order solves are mostly needed at the beginning of the computation or in case of a sudden change to a new unknown solution behavior. For the numerical test at hand this change in solution behavior takes place around each full second, when we toggle between heating and cooling. Therefore, we only need a few full-order model solves at these time slabs, but can use the reduced-order model in the majority of the time.

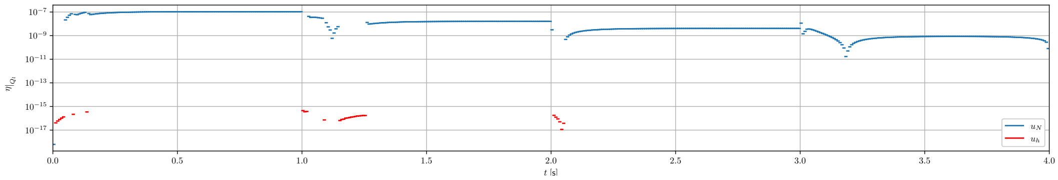


Figure 3: Temporal evolution of the slabwise error estimator for two alternating heat sources.

In Figure 4, we illustrate the time trajectory of the goal functional restricted to each time slab for the full-order space-time solution u_h and the reduced-order space-time solution u_N. The observed error in the cost functional between the full- and reduced-order solution is in the expected range of the tolerance.

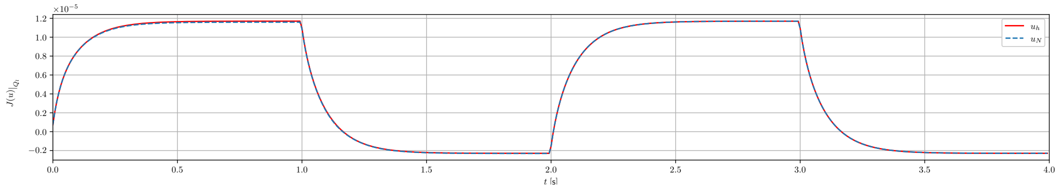


Figure 4: Temporal evolution of cost functional for two alternating heat sources.

In Table 5, we place side by side the results from the full-order model and our proposed certified incremental reduced-order model. As we have seen in Figure 3, the certified iROM reduces the number the full-order evaluations significantly. Further, the reduced system is far smaller, thus enabling an overall performance boost. Additionally, the error index validates the accuracy of the error estimator. Finally, we also tried replacing the fine dual solution in the error estimator, which requires expensive full-order solves, by a POD-ROM dual solution with a high energy content. We observe that this still yields excellent error estimates.

	FOM	Certified iROM
Evaluation $J(u)$	$2.449663 \cdot 10^{-3}$	$2.436404 \cdot 10^{-3}$
#FOM solves	512	31
#ROM solves	–	481
#DoFs FOM per slab	8,194	8,194
#DoFs ROM per slab	–	≤ 8
Execution time	2.14 s	0.79 s
Effectivity index with z_h	–	1.000000
Effectivity index with z_{N_d}	–	0.999181

Figure 5: Comparison of full-order and certified incremental reduced-order model solve for two alternating heat sources.

TW: Nice. But 2.14s vs. 0.79s is not yet completely amazing. Is it possible to compute larger problems with more significant differences in the computing time?

This is why our next step is to go to 2D where we can expect higher speedups due to larger FOM systems. However, the speed-up really depends on the chosen heating scenario and since it can fully replace FEM this speed-up is also quite good in our opinion. If we would merely increase each heating interval, we would get better speed ups. Additionally, the current iROM implementation has room for improvement which we have to investigate.

4.2 Moving heat source

For our second numerical test, we use a single moving heat source which changes its temperature after each second and moves through the spatial domain with heating interval width of 0.1 from $x = 0.1$ to 0.9 and then back to 0.1. For this we use the right hand side function

$$f(t,x) := \begin{cases} 0.2 & t \in (0,1),\ -0.05 \leq x - 0.4t - 0.1 \leq 0.05 \\ -0.5 & t \in (1,2),\ -0.05 \leq x - 0.4t - 0.1 \leq 0.05 \\ 1.0 & t \in (2,3),\ -0.05 \leq x + 0.4(t-2) - 0.9 \leq 0.05 \\ -0.75 & t \in (3,4),\ -0.05 \leq x + 0.4(t-2) - 0.9 \leq 0.05 \end{cases}.$$

In Figure 6, we present the reduced-order space-time solution u_N which is solved utilizing the previously introduced certified incremental ROM. Additionally, we plot the true error $u_h - u_N$ between the full-order space-time solution u_h and the reduced-order space-time solution u_N.

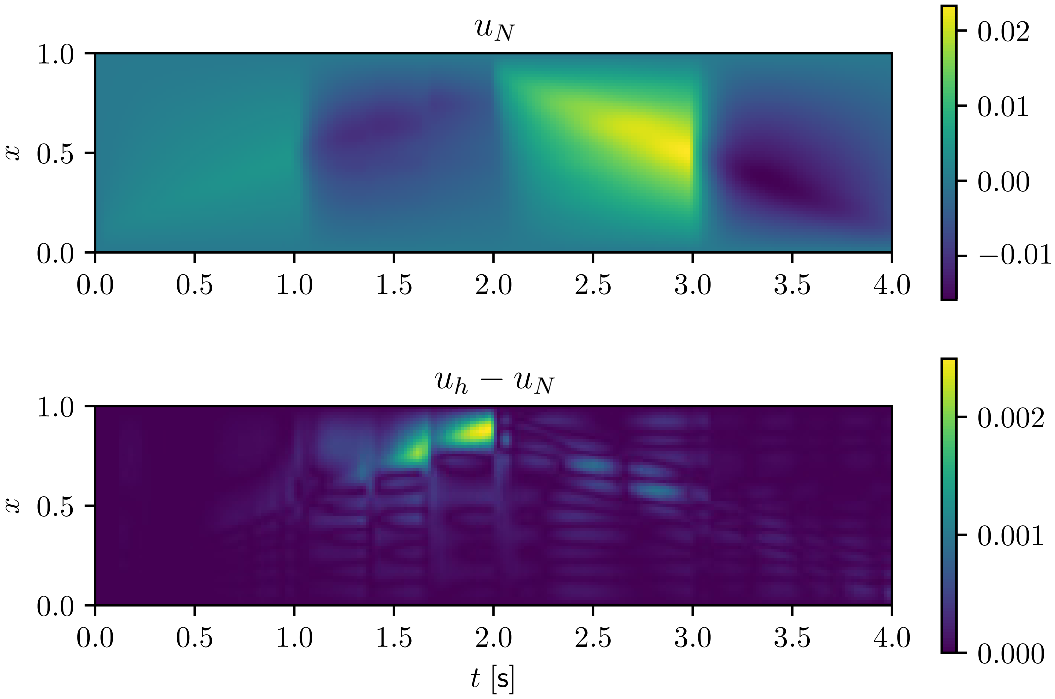


Figure 6: Temporal evolution of the reduced order solution u_N and the true error $u_h - u_N$ for the moving heat sources.

In Figure 7, the temporal error estimates on each slab are shown. Note, we use again slabs consisting of one time interval each. We indicate full-order solves in red color and reduced-order solves in blue. We observe that full-order solves are mostly needed at the beginning of the computation when the heat source heats the area in which the cost functional lives. Thereafter, we see sporadic updates of the reduced basis when the threshold is exceeded. Therefore, we only need some full-order model solves for the initial enrichment of the basis, but can use the reduced-order model in the majority of the time.

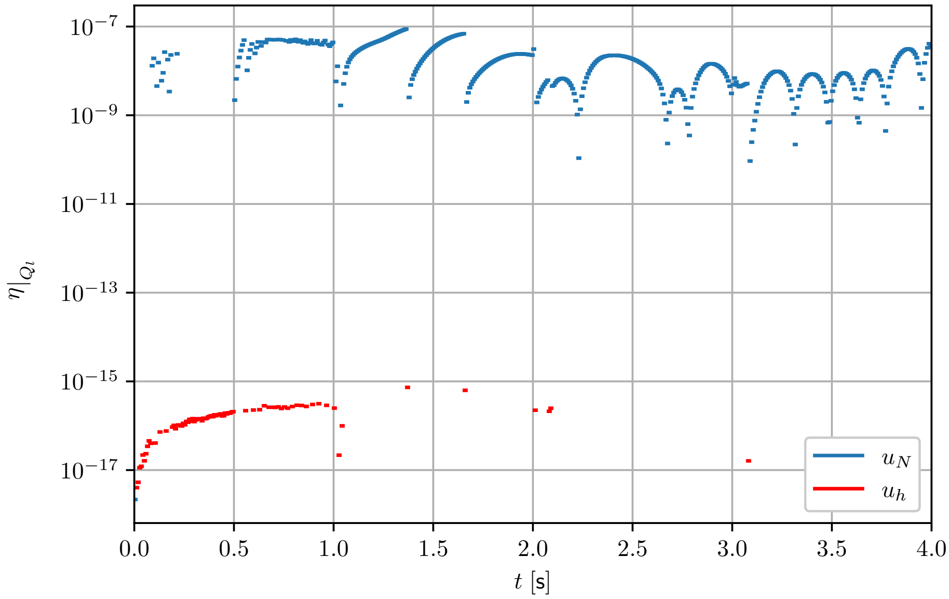


Figure 7: Temporal evolution of the time interval-wise error estimator for the moving source.

In Figure 8, we illustrate the time trajectory of the goal functional restricted to each time slab for the full-order space-time solution u_h and the reduced-order space-time solution u_N. The observed error in the cost functional between the full- and reduced-order solution is in the expected range of the tolerance.

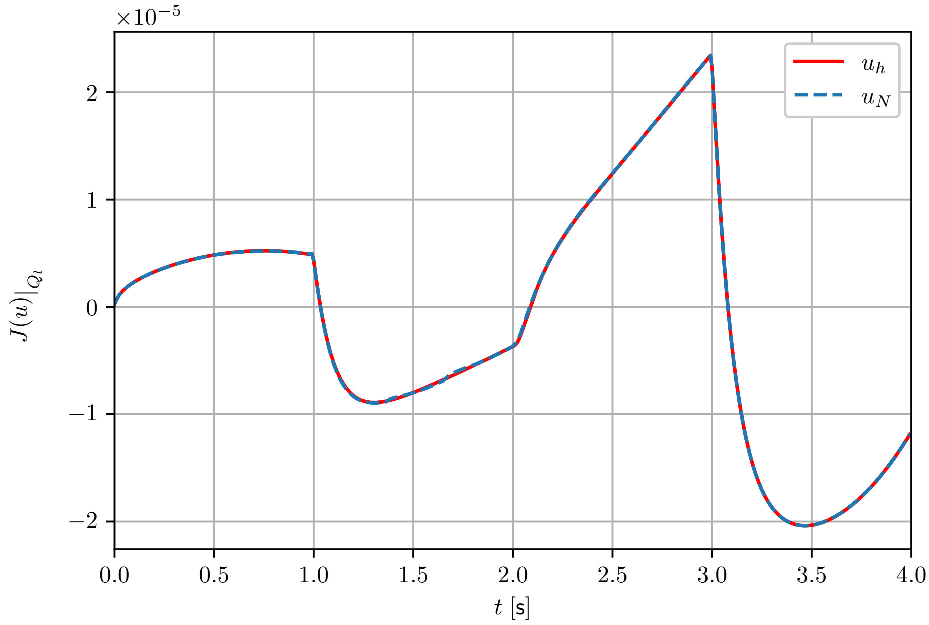


Figure 8: Temporal evolution of cost functional for the moving source.

In Table 9, we place side by side the results from the full-order model and our proposed certified incremental reduced-order model. As we have seen in Figure 7, the certified iROM reduces the number the full-order evaluations quite a bit. Further, the reduced system is a bit bigger than in the last numerical test, but still far smaller then the full-order linear equation system and thus enabling an overall performance boost. Additionally, the error index validates the accuracy of the error estimator. Finally, we also tried replacing the fine dual solution in the error estimator, which requires expensive full-order solves, by a POD-ROM dual solution with a high energy content. We observe that this still yields excellent error estimates.

	FOM	Certified iROM
Evaluation $J(u)$	$6.104924 \cdot 10^{-4}$	$6.078482 \cdot 10^{-4}$
#FOM solves	512	79
#ROM solves	–	433
#DoFs FOM per slab	8194	8194
#DoFs ROM per slab	–	≤ 20
Execution time	2.57 s	1.75 s
Effectivity index with z_h	–	1.000000
Effectivity index with z_{N_d}	–	0.991585

Figure 9: Comparison of full-order and certified incremental reduced-order model solve for moving heat source experiment.

4.3 Elastodynamics

Link to example

4.4 Next Equations - Problems

1. This paper:
   - 1D Heat equation. (done; see this report)
   - 2D Heat equation. Moving heat source in RHS. (FOM solution stored in Images/rotating_circle.mp4)
   - Nonlinear cost functional L^2-error utilizing mass matrix.
   - Wave equation? - Reducible? Investigate singular values. (2D FOM solution stored in Images/wave.mp4)
2. Future steps:
   - Porous media back at LUH.
   - Burger's equations.
   - Navier-Stokes equations.
   - Optimization
3. Possible journals:
   - CMAME, Computers & Mathematics with Applications, Journal of Scientific Computing, (JCP)

Rely on TW's deal.ii codes.

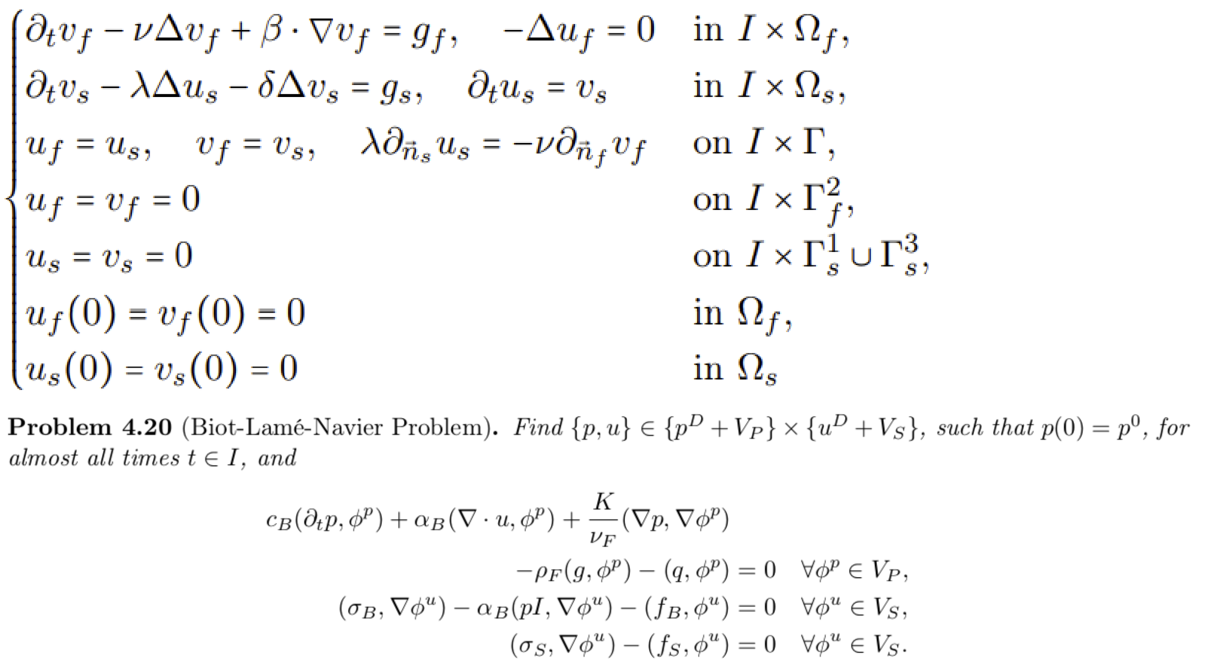
@Thomas: Which other numerical tests should we perform? Navier-Stokes? Wave equation? Porous media? Which problems make sense for our dissertations?

TW: For a possible paper with Amelie and Ludovic, we can decide together. For your dissertations, my very strong wish is also an example in porous media. Why? 1st: I want to keep a foot in the porous media community. 2nd : at least Hendrik suggested (of course I did this), to collaborate with Insa Neuweiler. So, at some point I want to keep this promise. And Wansheng Gao does beautiful work also in deal.II. Thus, we can collaborate with them also a bit, which is in the final end again fruitful for all of us within the IRTG. 3rd reason: having porous media, you have after your dissertations many, many possibilities since you have some knowledge in NSE, heat/wave equation and also porous media. This is excellent and a major research accomplishment.

AF: my preference is to employ as much as possible the same application as proposed by TW for your dissertations for our possible collaborative paper. Why?1st: the goal of IRTG is to offer an international collaboration incorporated in your doctoral project. Without being a double degree, which transfers the whole doctoral project into a collaboration, it would be nice that this collaboration is easily included as a chapter of your dissertation. 2nd: DFG pays attention that the international collaboration is really incorporated in the sense that this time at LMPS should not be a brake in your progress, but a booster. Multiplying the applications can become cumbersome. 3rd: having on a same benchmark example various developments is great because you can compare results, understand them in details,... and also expose them in conferences or papers more easily. 4th: coming from civil engineering community, porous media is a great topic so I will vote for it!

TW: super, Amelie. In case there is earlier interest in porous media, my suggestion is to address the very, very classical benchmark in porous media: Mandel's problem; see e.g. my own computations, but with all the important references cited therein: `https://www.repo.uni-hannover.de/handle/123456789/11802` Section 4.1.7. This problem is coupled with an elliptic and a parabolic part, which would fit perfectly into our setting. Recently, we have also done a nonlinear version `https://www.tandfonline.com/doi/full/10.1080/00036811.2022.2091992`

AF: I have looked at that example, i think it is great, and it should be possible to tune the boundary conditions to show some variable behaviors during different time slabs, to be discussed further.



$$\begin{cases} \partial_t v_f - \nu\Delta v_f + \beta\cdot\nabla v_f = g_f, \quad -\Delta u_f = 0 & \text{in } I\times\Omega_f, \\ \partial_t v_s - \lambda\Delta u_s - \delta\Delta v_s = g_s, \quad \partial_t u_s = v_s & \text{in } I\times\Omega_s, \\ u_f = u_s, \quad v_f = v_s, \quad \lambda\partial_{\vec{n}_s} u_s = -\nu\partial_{\vec{n}_f} v_f & \text{on } I\times\Gamma, \\ u_f = v_f = 0 & \text{on } I\times\Gamma_f^2, \\ u_s = v_s = 0 & \text{on } I\times\Gamma_s^1\cup\Gamma_s^3, \\ u_f(0) = v_f(0) = 0 & \text{in } \Omega_f, \\ u_s(0) = v_s(0) = 0 & \text{in } \Omega_s \end{cases}$$

Problem 4.20 (Biot-Lamé-Navier Problem). *Find* $\{p,u\}\in\{p^D+V_P\}\times\{u^D+V_S\}$, *such that* $p(0)=p^0$, *for almost all times* $t\in I$, *and*

$$\begin{aligned} c_B(\partial_t p,\phi^p) + \alpha_B(\nabla\cdot u,\phi^p) + \frac{K}{\nu_F}(\nabla p,\nabla\phi^p) & \\ -\rho_F(g,\phi^p) - (q,\phi^p) &= 0 \quad \forall\phi^p\in V_P, \\ (\sigma_B,\nabla\phi^u) - \alpha_B(pI,\nabla\phi^u) - (f_B,\phi^u) &= 0 \quad \forall\phi^u\in V_S, \\ (\sigma_S,\nabla\phi^u) - (f_S,\phi^u) &= 0 \quad \forall\phi^u\in V_S. \end{aligned}$$

Figure 10: Possible experiments. Heat+Wave probably next step. Biot-Lame-Navier for the beginning (with current implementation) out of scope, c.f. **??**

LC: Here are a few comments after a first read. 1- I think it is valuable to cite papers from Perrotto et al. on the HiMOD approach for slender structures, where POD and DWR are used (if I remember well, I already sent these papers, otherwise please let me know). 2- Would a parametrized version (e.g. Reynolds number as parameter in NS equation) make sense as a more complex case? 3- Also, probably we can investigate modeling error in this same DWR context (cf paper of Braack & Ern 2003), for instance when using a Stokes model (instead of NS) or a homogenized model (instead of heterogeneous for porous media) to define the POD basis, with splitting between modeling and discretization error sources on the target QoI. We can discuss all this in our coming meeting.

As a next step, we want to implement a 2+1D heat equation problem, e.g. something similar to the Hartman test case, and use a nonlinear goal functional, e.g. $L^2(I, L^2(\Omega))$-error. Moreover, Julian has just implemented an all-at-once cG(s) cG(r) wave equation solver or we could use the parallel NSE code by Julian and Philipp to do parallel incremental POD updates.

@all: Discuss next numerical tests and further steps.

References

[1] J. Baiges, R. Codina, and S. Idelsohn. Explicit reduced-order models for the stabilized finite element approximation of the incompressible navier–stokes equations. International Journal for Numerical Methods in Fluids, 72(12):1219–1243, 2013.

[2] W. Bangerth and R. Rannacher. Adaptive Finite Element Methods for Differential Equations. Birkhäuser Verlag„ 2003.

[3] D. Baroli, C. M. Cova, S. Perotto, L. Sala, and A. Veneziani. Hi-POD solution of parametrized fluid dynamics problems: preliminary results. Research report, MOX, Dipartimento di Matematica, Politecnico di Milano, July 2016.

[4] R. Becker and R. Rannacher. A feed-back approach to error control in finite element methods: basic analysis and examples. East-West J. Numer. Math., 4:237–264, 1996.

[5] R. Becker and R. Rannacher. An optimal control approach to a posteriori error estimation in finite element methods. Acta Numerica 2001, 10:1 – 102, 05 2001.

[6] P. Benner, W. Schilders, S. Grivet-Talocia, A. Quarteroni, G. Rozza, and L. Miguel Silveira. Model Order Reduction: Volume 2: Snapshot-Based Methods and Algorithms. De Gruyter, 2020.

[7] L. Bertagna and A. Veneziani. A model reduction approach for the variational estimation of vascular compliance by solving an inverse fluid–structure interaction problem. Inverse Problems, 30(5):055006, 2014.

[8] M. Brand. Incremental Singular Value Decomposition of Uncertain Data with Missing Values. In European Conference on Computer Vision, pages 707–720. Springer, 2002.

[9] M. Brand. Fast Low-Rank Modifications of the Thin Singular Value Decomposition. Linear Algebra and its Applications, 415(1):20–30, 2006.

[10] A. Caiazzo, T. Iliescu, V. John, and S. Schyschlowa. A numerical investigation of velocity–pressure reduced order models for incompressible flows. Journal of Computational Physics, 259:598–616, 2014.

[11] E. A. Christensen, M. Brøns, and J. N. Sørensen. Evaluation of proper orthogonal decomposition–based decomposition techniques applied to parameter-dependent nonturbulent flows. SIAM Journal on Scientific Computing, 21(4):1419–1434, 1999.

[12] B. Endtmayer. Multi-goal oriented a posteriori error estimates for nonlinear partial differential equations. PhD thesis, Johannes Kepler University Linz, 2021.

[13] C. Gräßle and M. Hinze. Pod reduced-order modeling for evolution equations utilizing arbitrary finite element discretizations. Advances in Computational Mathematics, 44(6):1941–1978, 2018.

[14] M. Gubisch and S. Volkwein. Proper orthogonal decomposition for linear-quadratic optimal control. Model reduction and approximation: theory and algorithms, 5:66, 2017.

[15] M. D. Gunzburger, J. S. Peterson, and J. N. Shadid. Reduced-order modeling of time-dependent pdes with multiple parameters in the boundary data. Computer methods in applied mechanics and engineering, 196(4-6):1030–1047, 2007.

[16] B. Haasdonk. Reduced basis methods for parametrized pdes–a tutorial introduction for stationary and instationary problems. Model reduction and approximation: theory and algorithms, 15:65, 2017.

[17] B. Haasdonk and M. Ohlberger. Reduced basis method for finite volume approximations of parametrized linear evolution equations. ESAIM: Mathematical Modelling and Numerical Analysis, 42(2):277–302, 2008.

[18] J. S. Hesthaven, G. Rozza, B. Stamm, et al. Certified reduced basis methods for parametrized partial differential equations, volume 590. Springer, 2016.

[19] Y. Kim, K. Wang, and Y. Choi. Efficient space–time reduced order model for linear dynamical systems in python using less than 120 lines of code. Mathematics, 9(14):1690, 2021.

[20] A. Kolmogoroff. über die beste annäherung von funktionen einer gegebenen funktionenklasse. Annals of Mathematics, pages 107–110, 1936.

[21] N. Kühl, H. Fischer, and T. Rung. An incremental singular value decomposition approach for large-scale spatially parallel & distributed but temporally serial data – applied to technical flows. In preparation.

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

[23] U. Langer and O. Steinbach, editors. Space-time methods: Application to Partial Differential Equations. volume 25 of Radon Series on Computational and Applied Mathematics, Berlin. de Gruyter, 2019.

[24] T. Lassila, A. Manzoni, A. Quarteroni, and G. Rozza. Model Order Reduction in Fluid Dynamics: Challenges and Perspectives. In Reduced Order Methods for Modeling and Computational Reduction, volume 9, pages 235–273. Springer International Publishing, 2014.

[25] M. Meyer and H. G. Matthies. Efficient model reduction in non-linear dynamics using the Karhunen-Loève expansion and dual-weighted-residual methods. Computational Mechanics, 31(1):179–191, May 2003.

[26] N.-C. Nguyen, G. Rozza, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for the time-dependent viscous burgers' equation. Calcolo, 46(3):157–185, 2009.

[27] S. Perotto, M. G. Carlino, and F. Ballarin. Model Reduction by Separation of Variables: A Comparison Between Hierarchical Model Reduction and Proper Generalized Decomposition. In S. J. Sherwin, D. Moxey, J. Peiró, P. E. Vincent, and C. Schwab, editors, Spectral and High Order Methods for Partial Differential Equations ICOSAHOM 2018, volume LNCSE - Lecture Notes in Computational Science and Engineering, pages 61–77. Springer, Aug. 2020.

[28] S. Perotto and A. Zilio. Space–time adaptive hierarchical model reduction for parabolic equations. Advanced Modeling and Simulation in Engineering Sciences, 2, 12 2015.

[29] S. S. Ravindran. A reduced-order approach for optimal control of fluids using proper orthogonal decomposition. International journal for numerical methods in fluids, 34(5):425–448, 2000.

[30] J. Roth, J. P. Thiele, U. Köcher, and T. Wick. Tensor-product space-time goal-oriented error control and adaptivity with partition-of-unity dual-weighted residuals for nonstationary flow problems, 2022.

[31] G. Rozza. Shape design by optimal flow control and reduced basis techniques. Technical report, EPFL, 2005.

[32] G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. Archives of Computational Methods in Engineering, 15(3):229–275, 2008.

[33] M. Schmich. Adaptive Finite Element Methods for Computing Nonstationary Incompressible Flows. Doctoral Thesis, Heidelberg University, 2009. ISSN: 0001-0200.

[34] M. Schmich and B. Vexler. Adaptivity with dynamic meshes for space-time finite element discretizations of parabolic equations. SIAM J. Sci. Comput., 30(1):369–393, 2008.

[35] L. Sirovich. Turbulence and the dynamics of coherent structures. i. coherent structures. Quarterly of applied mathematics, 45(3):561–571, 1987.

[36] J. P. Thiele and T. Wick. Variational partition-of-unity localizations of space-time dual weighted residual estimators for parabolic problems, 2022.

[37] K. Willcox and J. Peraire. Balanced model reduction via the proper orthogonal decomposition. AIAA journal, 40(11):2323–2330, 2002.