AN EFFICIENT LOCALIZED MODEL ORDER REDUCTION FRAMEWORK FOR THE SHAPE OPTIMIZATION OF ADDITIVELY MANUFACTURED LATTICE STRUCTURES ECCOMAS CONGRESS 2024

Philipp Diercks[1], Karen Veroy[2], Annika Robens-Radermacher[1] and Jörg F. Unger[1]

[1] Bundesanstalt für Materialforschung und -prüfung, Unter den Eichen 87, 12205 Berlin, {philipp.diercks, annika.robens-radermacher, joerg.unger}@bam.de, `www.bam.de`

[2] Centre for Analysis, Scientific Computing and Applications (CASA), Department of Mathematics and Computer Science, TU Eindhoven, P.O. Box 513, 5600 MB Eindhoven, The Netherlands, k.p.veroy@tue.nl

Key words: Multiscale methods, Domain decomposition methods, Model order reduction, parameterized PDEs, Shape optimization

Summary. A common engineering practice is to optimize the geometry of a structure by an iterative process, in which an objective function is minimized by systematically choosing the value of design variables and computing the value of the objective function many times. However, regarding multiscale problems, this direct numerical approach is not feasible for a number of reasons. The model has to take into account both the structural and unit cell (UC) scale, because the objective function to be minimized is a global quantity, while the parameters are related to the shape or material parameters of the UC. In the absence of a clear separation of scales, the direct resolution of both scales in the numerical model leads to a significant increase in computational cost, which makes it impossible to repeatedly evaluate the model during the optimization process. The main goal of this contribution is to overcome the aforementioned limitations and develop an efficient computational framework for the optimal design of lattice structures. To this end, parametric model order reduction is combined with domain decomposition methods. In offline simulations, suitable localized approximation spaces, that account for the change in geometry of the UC and its neighbours, are constructed using the concept of oversampling and random sampling. The global approximation space is constructed from the local spaces via the partition of unity method. The reduced order model is validated on the example of a stress-constrained shape optimization problem.

1 INTRODUCTION

Additive manufacturing is a manufacturing technique that allows for the production of a wide range of structures and complex geometries. The technology offers numerous advantages over conventional manufacturing, including greater design flexibility, reduced material waste, and the possibility to produce complex structures with tailored material properties. It has been used in a wide range of applications, including the construction industry [1]. The great flexibility regarding the geometry motivates the solution of topology or shape optimization problems to

determine the optimal design for a structural problem. In this contribution, we focus on shape optimization in the context of multiscale problems, i.e. our aim is to optimize the geometrical parameters of the microstructure, e.g. the unit cell of a lattice structure. However, the solution of the optimization problem requires many evaluations of the numerical model, which may be infeasible if solving the high fidelity finite element (FE) model, also called full order model (FOM) is a computationally demanding task. Reduced order models (ROMs) are powerful tools to mitigate this problem, but for multiscale or large-scale industrial applications with high-dimensional parameter spaces, the nowadays established "single-domain" approaches are limited [2], and the construction of a fast-to-evaluate and accurate ROM is a challenging task.

Therefore, we propose a framework based on localized, also called component-based (CB), parametric model order reduction (pMOR). The main idea is to precompute, in a localized manner, empirical basis functions which approximate the solution for some part of the domain without the need to solve the global FOM even once. The global approximation is obtained by a suitable coupling of the local reduced spaces spanned by the aforementioned basis functions, in which one naturally relies on domain decomposition (DD) strategies. A review of concepts in localized pMOR is presented in [3]. In particular, regarding lattice structures, this approach allows to take advantage of the repetitiveness of the lattice, such that a computation of the local basis is required only for few components, i.e. unit cells. The computation of the local basis is an essential task in localized pMOR and is often done using the concept of oversampling [4]. In this approach, the target subdomain Ω_{in}, i.e. that part of the domain for which one would like to construct basis functions, is extended and boundary conditions are prescribed on the boundary of the larger so-called oversampling domain Ω to explore possible solutions. In the literature, this oversampling problem is also expressed in terms of a *transfer operator* $\boldsymbol{T}$ that maps the values on the boundary $\partial\Omega$ to the unknown solution restricted to the target subdomain $\boldsymbol{u}|_{\Omega_{\mathrm{in}}}$. The construction of (optimal) local approximation spaces then comprises the calculation of the left singular vectors of this transfer operator [5, 6]. The direct calculation via eigenvalue problems is, however, computationally expensive and the range of the transfer operator, and thus the optimal local approximation spaces can be efficiently approximated by random sampling [7]. Here, the authors treat non-parametrized partial differential equations (PDEs) and to our knowledge the extension to the parametric setting for linear problems has not been done yet. In [6], the authors propose a spectral greedy algorithm to construct parameter-independent local approximation spaces and Taddei and Patera [8] propose a combination of transfer eigenproblems and proper orthogonal decomposition (POD). For parameterized nonlinear elliptic PDEs Smetana and Taddei [9] present a randomized local training procedure with global enrichment.

The contributions of the present work are given as follows. First, as references [6, 8] do not make use of range approximation via random sampling, a suitable training strategy to construct local approximation spaces for parameterized linear problems via random sampling is discussed. Herein, the approach given in [8], identified as a *distributed approximate POD* (see also [10]), is adopted to range approximation via random sampling. Second, a framework for the shape optimization of lattice structures is proposed. It combines the aforementioned algorithm to construct local approximation spaces with an auxiliary problem, as in [11], to facilitate geometrical parametrizations of the unit cell and the matrix version [12] of the empirical interpolation method (EIM) [13, 14] to ensure online efficiency of the final ROM. Furthermore, the global approximation space is constructed from the local spaces using the partition of unity method or generalized finite element method (GFEM) [15, 16].

The rest of the article is organized as follows. In section 2, the building blocks of the shape optimization framework are described. In particular, the range approximation of a parametric transfer operator is discussed. Section 3 comprises the numerical experiments. Based on the example of a graded concrete slab, the quality of the local spaces generated by the proposed method is analyzed, the resulting ROM is validated, and results for the solution of a stress-constrained shape optimization problem are presented. Finally, conclusions are given in section 4.

2 METHOD

The method proposed in this article is a localized pMOR framework for the shape optimization of lattice structures. First, the auxiliary problem to model geometrical parameterizations is introduced in section 2.1. Second, the construction of local approximation spaces is described in section 2.2. Herein, the parametric oversampling problem and training strategy to approximate the range of the corresponding parametric transfer operator are discussed. Third, it is outlined how a global approximation is obtained from local approximation spaces via the GFEM. Finally, the hyper-reduction (in the form of empirical interpolation) to ensure an online-efficient ROM is detailed in section 2.4.

2.1 Auxiliary Problem

The approach adopted in this work to facilitate geometrical parameterizations belongs to the class of surface-based deformations [17] and is based on transformations $\Phi_{\boldsymbol{\mu}}$ that map each material point $\boldsymbol{x}_{\mathrm{p}}$ of a parameter-independent reference or parent domain Ω^{p} to a point $\boldsymbol{x}_{\boldsymbol{\mu}}$ in the parameter-dependent current or physical domain $\Omega^{\boldsymbol{\mu}}$. Throughout the paper, the parameter value (design variables) are denoted by $\boldsymbol{\mu}$ and dependence on the parameter is indicated by a sub- or superscript $(\cdot)_{\boldsymbol{\mu}}, (\cdot)^{\boldsymbol{\mu}}$. The sub- or superscript $(\cdot)_{\mathrm{p}}, (\cdot)^{\mathrm{p}}$ indicates that the quantity does not depend on the parameter. In the context of shape optimization of lattice structures, our objective is to determine the mapping $\Phi_{\boldsymbol{\mu}}$ for a single unit cell, see fig. 1, and use it to describe the change in geometry of each unit cell throughout the structure. An auxiliary problem based on the equations of linear elasticity is solved to obtain such domain transformations, following the approach outlined in [11], and briefly repeated here for completeness.

The transformation map $\Phi_{\boldsymbol{\mu}}$ from parent to physical domain $\Phi_{\boldsymbol{\mu}} : \Omega^{\mathrm{p}} \mapsto \Omega^{\boldsymbol{\mu}}$ is given by $\boldsymbol{x}_{\boldsymbol{\mu}} = \Phi_{\boldsymbol{\mu}}(\boldsymbol{x}_{\mathrm{p}}) = \boldsymbol{x}_{\mathrm{p}} + \boldsymbol{d}(\boldsymbol{x}_{\mathrm{p}}; \boldsymbol{\mu})$, with $\boldsymbol{d}(\boldsymbol{x}_{\mathrm{p}}; \boldsymbol{\mu})$ being the transformation displacement field. The transformation displacement is determined by solving the following linear elastostatic auxiliary problem.

$$\boldsymbol{\nabla} \cdot \left(\hat{\mathbb{C}} \cdot \cdot \frac{1}{2} \left(\boldsymbol{d} \otimes \boldsymbol{\nabla} + \boldsymbol{\nabla} \otimes \boldsymbol{d} \right) \right) = \boldsymbol{0}, \quad \text{in } \Omega^{\mathrm{p}}, \tag{1}$$

$$\boldsymbol{d} = \boldsymbol{0}, \quad \text{on } \partial\Omega^{\mathrm{p}}, \tag{2}$$

$$\boldsymbol{d} = \boldsymbol{x}_{\mu} - \boldsymbol{x}_{\mathrm{p}}, \quad \text{on } \partial\Omega^{\mathrm{p}}_{\mathrm{int}}. \tag{3}$$

The stiffness tetrad $\hat{\mathbb{C}}$ of the auxiliary problem is defined as

$$\hat{\mathbb{C}} = \hat{\lambda} \boldsymbol{I} \otimes \boldsymbol{I} + 2\hat{\mu}\mathbb{I}, \quad \text{with} \quad \hat{\lambda} = \frac{\nu}{(1+\nu)(1-2\nu)} \quad \text{and} \quad \hat{\mu} = \frac{1}{2(1+\nu)}. \tag{4}$$

With $\boldsymbol{x}_{\boldsymbol{\mu}}$ known for all points on the parent interface $\partial\Omega^{\mathrm{p}}_{\mathrm{int}}$, the desired transformation (e.g. enlarging/shrinking of the voids radius) is enforced by prescribing eqs. (2) and (3). Given
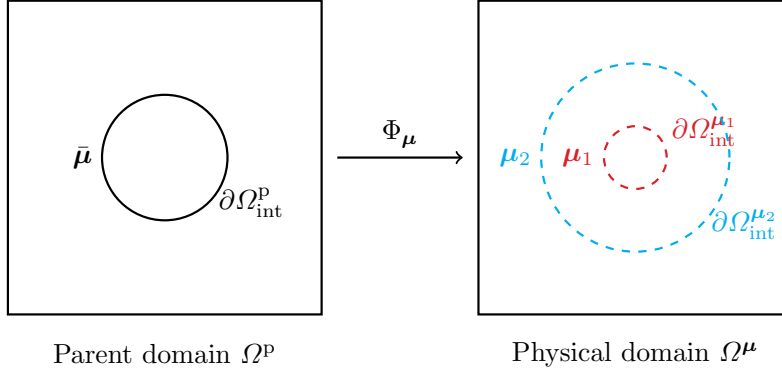
3

<center>Parent domain Ω^{p} Physical domain $\Omega^{\boldsymbol{\mu}}$</center>

Figure 1: Transformation map $\Phi_{\boldsymbol{\mu}}$ from parent to physical domain on the example of the unit square domain with a circular void placed in the center. The parameter $\boldsymbol{\mu}$ controls the radius of the void.

the transformation displacement $\boldsymbol{d}(\boldsymbol{x}_{\mathrm{p}}; \boldsymbol{\mu})$, the variational formulation is formulated over the parameter-independent parent domain Ω^{p} instead of the parameter-dependent physical domain $\Omega^{\boldsymbol{\mu}}$. This introduces the parameter dependence in the variational formulation as shown in the following subsections, but has the advantage that numerical integration can be carried out over the fixed parent domain and thus no re-meshing is required. However, for each new parameter value $\boldsymbol{\mu}$ for which the model is to be evaluated, we first need to solve the auxiliary problem. Nevertheless, this does not pose a problem, since the solution of the auxiliary problem is well amenable to acceleration via established MOR techniques and we refer to [11] for further details.

2.2 Construction of local approximation spaces

In this paper, local approximation spaces are constructed by solving an oversampling problem many times for different parameter values $\boldsymbol{\mu}$ and different (random) boundary conditions. It is therefore useful to cast this oversampling problem in the form of a parameter-dependent transfer operator $\boldsymbol{T}_{\boldsymbol{\mu}}$ that maps the boundary function $\boldsymbol{g}$ to the solution $\boldsymbol{u}(\boldsymbol{\mu})$ in the target subdomain $\Omega^{\boldsymbol{\mu}}_{\mathrm{in}}$. The approximation of the range of this transfer operator is then the local approximation space, see [7].

First, the global domain $\Omega^{\boldsymbol{\mu}}_{\mathrm{gl}}$ and a non-overlapping domain decomposition

$$\Omega^{\boldsymbol{\mu}}_{\mathrm{gl}} = \cup_{i=1}^{N_{\mathrm{cells}}} \Omega^{\boldsymbol{\mu}}_i \tag{5}$$

is introduced. In the context of lattice structures, each subdomain $\Omega^{\boldsymbol{\mu}}_i$ corresponds to a unit cell. Next, a coarse grid partition of the global domain and a fine grid partition of the unit cell is introduced as shown in fig. 2. Figure 2 also shows one exemplary oversampling domain $\Omega^{\boldsymbol{\mu}}$. The oversampling problem (given for linear elastostatics) then comprises the solution of the following boundary value problem.

$$\begin{aligned}
-\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}(\boldsymbol{u}(\boldsymbol{\mu})) &= \boldsymbol{0} && \text{in } \Omega^{\boldsymbol{\mu}} \subset \Omega^{\boldsymbol{\mu}}_{\mathrm{gl}}, \\
\boldsymbol{\sigma}(\boldsymbol{u}(\boldsymbol{\mu})) \cdot \boldsymbol{n} &= \boldsymbol{0} && \text{on } \Gamma^{\boldsymbol{\mu}}_{\mathrm{N}} := \partial\Omega^{\boldsymbol{\mu}} \cap \Sigma^{\boldsymbol{\mu}}_{\mathrm{N}}, \\
\boldsymbol{u}(\boldsymbol{\mu}) &= \boldsymbol{0} && \text{on } \Gamma^{\boldsymbol{\mu}}_{\mathrm{D}} := \partial\Omega^{\boldsymbol{\mu}} \cap \Sigma^{\boldsymbol{\mu}}_{\mathrm{D}}, \\
\boldsymbol{u}(\boldsymbol{\mu}) &= \boldsymbol{g} && \text{on } \Gamma^{\boldsymbol{\mu}}_{\mathrm{out}} := \partial\Omega^{\boldsymbol{\mu}} \setminus \partial\Omega^{\boldsymbol{\mu}}_{\mathrm{gl}}.
\end{aligned} \tag{6}$$

<center>4</center>

Here, $\boldsymbol{\sigma}$ is the CAUCHY stress, $\boldsymbol{n}$ the normal vector and $\boldsymbol{g}$ DIRICHLET boundary data to be prescribed on the boundary $\varGamma_{\text{out}}^{\boldsymbol{\mu}}$. The boundaries $\varGamma_{\text{N}}^{\boldsymbol{\mu}}$ and $\varGamma_{\text{D}}^{\boldsymbol{\mu}}$ denote the part of the boundary of the oversampling domain that coincides with the global NEUMANN boundary $\varSigma_{\text{N}}^{\boldsymbol{\mu}}$ or DIRICHLET boundary $\varSigma_{\text{D}}^{\boldsymbol{\mu}}$, respectively. Note that the topology is dependent on the target subdomain $\varOmega_{\text{in}}^{\boldsymbol{\mu}}$ and the size of the oversampled region. Considering linear elastic isotropic materials

$$\boldsymbol{\sigma}(\boldsymbol{u}(\boldsymbol{\mu})) = \mathbb{C} \cdot\!\cdot\, \boldsymbol{\varepsilon}(\boldsymbol{u}(\boldsymbol{\mu})) \,, \tag{7}$$

with the linear strain tensor as

$$\boldsymbol{\varepsilon}(\boldsymbol{w}) := \frac{1}{2}\left(\boldsymbol{w} \otimes \boldsymbol{\nabla} + \boldsymbol{\nabla} \otimes \boldsymbol{w}\right) \,, \tag{8}$$

and the stiffness tetrad given in component form as

$$C_{ijkl} = \lambda \delta_{ij}\delta_{kl} + \mu \left(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}\right) \,, \tag{9}$$

the weak form reads

$$\int_{\varOmega^{\boldsymbol{\mu}}} \frac{\partial \delta\boldsymbol{u}}{\partial \boldsymbol{x}^{\boldsymbol{\mu}}} \cdot\!\cdot\, \mathbb{C} \cdot\!\cdot\, \frac{\partial \boldsymbol{u}(\boldsymbol{\mu})}{\partial \boldsymbol{x}^{\boldsymbol{\mu}}} \, \mathrm{d}\boldsymbol{x}^{\boldsymbol{\mu}} = \boldsymbol{0} \,. \tag{10}$$

Herein, $\delta\boldsymbol{u}$ denotes the test function. The weak form is implicitly dependent on the parameter $\boldsymbol{\mu}$ due to the integration carried out over the physical domain $\varOmega^{\boldsymbol{\mu}}$. By introducing the deformation gradient of the geometrical transformation[1] as introduced in section 2.1

$$\boldsymbol{F_\mu} := \frac{\mathrm{d}\boldsymbol{x}^{\boldsymbol{\mu}}}{\mathrm{d}\boldsymbol{x}^{\text{p}}} \quad \text{and} \quad \mathrm{d}\boldsymbol{x}^{\boldsymbol{\mu}} = \det(\boldsymbol{F_\mu})\,\mathrm{d}\boldsymbol{x}^{\text{p}} \,, \tag{11}$$

the integration can be carried out over the fixed parent domain

$$\int_{\varOmega^{\text{p}}} \left(\frac{\partial \delta\boldsymbol{u}}{\partial \boldsymbol{x}^{\text{p}}} \cdot \boldsymbol{F_\mu}^{-1}\right) \cdot\!\cdot\, \mathbb{C} \cdot\!\cdot\, \left(\frac{\partial \boldsymbol{u}(\boldsymbol{\mu})}{\partial \boldsymbol{x}^{\text{p}}} \cdot \boldsymbol{F_\mu}^{-1}\right) \det(\boldsymbol{F_\mu})\,\mathrm{d}\boldsymbol{x}^{\text{p}} = \boldsymbol{0} \,. \tag{12}$$

The solution of the oversampling problem eq. (6) is then given as the solution of eq. (12) and subsequent restriction of the solution to the target subdomain, denoted as $\boldsymbol{u}(\boldsymbol{\mu})\big|_{\varOmega_{\text{in}}^{\text{p}}}$. The transfer operator $\boldsymbol{T_\mu} : \mathcal{S} \to \mathcal{R}$ maps, for a given parameter value $\boldsymbol{\mu}$, the boundary data $\boldsymbol{g}$ to the solution on the target subdomain $\boldsymbol{u}(\boldsymbol{\mu})\big|_{\varOmega_{\text{in}}^{\text{p}}} = \boldsymbol{T_\mu}(\boldsymbol{g})$. Here, $\mathcal{S}$ and $\mathcal{R}$ are appropriate source and range spaces, respectively. Throughout the paper, we assume a suitable discretization and numerical solution of eq. (12) using the FE method. For details concerning the discretization of the transfer operator (for a fixed parameter value), we refer to [7].

Finally, we note that the images of $\boldsymbol{T_\mu}$ are computed, such that they do not contain any translations or rotations (rigid body modes are removed via orthogonal projection), i.e. they comprise purely *deformational modes*. Moreover, (global) NEUMANN or DIRICHLET boundary conditions prescribed on $\varSigma_{\text{N}}^{\boldsymbol{\mu}}$ or $\varSigma_{\text{D}}^{\boldsymbol{\mu}}$, respectively, have not been considered yet. Therefore, an

---

[1]We note here that the auxiliary problem described in section 2.1 needs to be extended to the oversampling domain, which consists of a union of unit cells. This is straightforward in the current setup, but has the limitation that eq. (2) needs to be satisfied for each unit cell, i.e. the interfaces between them.

additional NEUMANN problem[2] has to be solved to enrich the local spaces.

$$
\begin{aligned}
-\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}(\boldsymbol{u}(\boldsymbol{\mu})) &= \boldsymbol{0} && \text{in } \Omega^{\boldsymbol{\mu}} \subset \Omega_{\mathrm{gl}}^{\boldsymbol{\mu}}, \\
\boldsymbol{\sigma}(\boldsymbol{u}(\boldsymbol{\mu})) \cdot \boldsymbol{n} &= \hat{\boldsymbol{t}} && \text{on } \Gamma_{\mathrm{N}}^{\boldsymbol{\mu}} := \partial \Omega^{\boldsymbol{\mu}} \cap \Sigma_{\mathrm{N}}^{\boldsymbol{\mu}}, \\
\boldsymbol{u}(\boldsymbol{\mu}) &= \boldsymbol{0} && \text{on } \Gamma_{\mathrm{D}}^{\boldsymbol{\mu}} := \partial \Omega^{\boldsymbol{\mu}} \cap \Sigma_{\mathrm{D}}^{\boldsymbol{\mu}}, \\
\boldsymbol{u}(\boldsymbol{\mu}) &= \boldsymbol{0} && \text{on } \Gamma_{\mathrm{out}}^{\boldsymbol{\mu}} := \partial \Omega^{\boldsymbol{\mu}} \setminus \partial \Omega_{\mathrm{gl}}^{\boldsymbol{\mu}}.
\end{aligned}
\tag{13}
$$

Equation (13) is solved on the parent domain employing the pull back as above. Also, for construction of a sufficiently rich local basis it has to be solved for many parameter values $\boldsymbol{\mu}$, but a sampling of the boundary values on $\Gamma_{\mathrm{out}}^{\boldsymbol{\mu}}$ is not required.
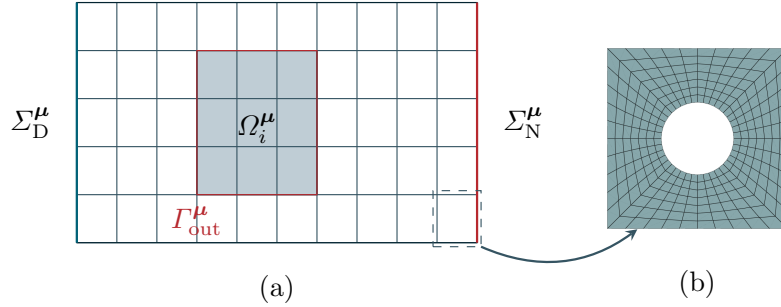

Figure 2: Exemplary coarse grid discretization of the global domain $\Omega_{\mathrm{gl}}^{\boldsymbol{\mu}}$, with target subdomain $\Omega_i^{\boldsymbol{\mu}}$ within oversampling domain $\Omega^{\boldsymbol{\mu}}$ in grey (a) and fine grid discretization of the unit cell (b).

2.2.1 Randomized range finder and proper orthogonal decomposition

Assuming a proper FE discretization of $\boldsymbol{T}_{\boldsymbol{\mu}}$, the algorithm to approximate the range of $\boldsymbol{T}_{\boldsymbol{\mu}}$ for any $\boldsymbol{\mu}$ is discussed next. Herein, a strategy for the exploration of the parameter space and sampling of the random boundary conditions in the oversampling are the key challenges. In the present work, the strategy by Taddei & Patera [8] is adopted in which based on the connection between the POD of images of the transfer operator (for certain boundary conditions) and the singular values of the transfer operator they devise a training strategy which is recognized as a distributed approximate POD (see [10]). For each $\boldsymbol{\mu}_j$ in a predefined training set, they compute the singular values of $\boldsymbol{T}_{\boldsymbol{\mu}_j}$ by solution of an eigenvalue problem and, subsequently, compute the POD over the set of all eigenvectors. As the authors state, the resolution of the eigenvalue problems may be prohibitive in some cases, and therefore we employ range approximation by random sampling.

The procedure for the computation of parameter-independent deformational modes is summarized in algorithm 1. The target tolerance ε^* is used to control the quality of the approximation space (by bounding the mean ℓ^2-error), and to compute the local tolerances ε_α that steer the range approximation of each transfer operator and the tolerance ε_ρ for the final POD. First, an empty snapshot set $\boldsymbol{S}$ and a training set $\boldsymbol{S}_{\mathrm{train}}$ are initialized. The training set $\boldsymbol{S}_{\mathrm{train}}$ has to be sufficiently rich, such that it is a good representation of the parameter space, and the reader

---

[2]The case of inhomogeneous DIRICHLET boundary conditions can be treated in the same way.

is referred to [18]. Next, for each parameter value $\boldsymbol{\mu}_j \in \boldsymbol{S}_{\text{train}}$, the range of $\boldsymbol{T}_{\boldsymbol{\mu}_j}$ which is now fixed with respect to the parameter value is approximated using Algorithm 1 ADAPTIVERAN-DOMIZEDRANGEAPPROXIMATION from [7] (line 6). If $\partial \Omega^{\text{p}} \cap \Sigma_{\text{N}}^{\text{p}} \neq \emptyset$, the additional NEUMANN problem eq. (13) is solved and the solution appended to the current range $\boldsymbol{R}$ (lines 7-9). The set of snapshots is then extended by the current range $\boldsymbol{R}$ (line 11). Note that the parameter value $\boldsymbol{\mu}$ and boundary conditions $\boldsymbol{g}$ are not varied at the same time. Finally, the POD is computed over the snapshot set $\boldsymbol{S}$ which yields the parameter-independent space X^n. The dimension of the local space X^n, i.e. the number of local basis functions is denoted by n.

Algorithm 1 Randomized range finder (RRF) combined with POD.

1: **function** HAPODRANGEAPPROXIMATION($\boldsymbol{T}$, ε^*, n_t, $\varepsilon_{\text{algofail}}$, N_{train})
    Input: Operator $\boldsymbol{T}_{\boldsymbol{\mu}}$, target tolerance ε^*, number of testvectors n_t, maximum failure probability $\varepsilon_{\text{algofail}}$, number of parameter samples N_{train}
    Output: space X^n
2:     Compute ε_ρ and ε_α from ε^* according to Theorem 10 of [10]
3:     $\boldsymbol{S} \leftarrow \emptyset$ ▷ initialize snapshot set
4:     $\boldsymbol{S}_{\text{train}} \leftarrow \{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_{N_{\text{train}}}\}$ ▷ initialize training set
5:     **for** $\boldsymbol{\mu}_j$ in $\boldsymbol{S}_{\text{train}}$ **do**
6:         $\boldsymbol{R} \leftarrow$ ADAPTIVERANDOMIZEDRANGEAPPROXIMATION($\boldsymbol{T}_{\boldsymbol{\mu}_j}$, ε_α, n_t, $\varepsilon_{\text{algofail}}$)
7:         **if** $\partial \Omega^{\text{p}} \cap \Sigma_{\text{N}}^{\text{p}} \neq \emptyset$ **then**
8:             $\boldsymbol{u}_{\text{NEUMANN}} \leftarrow$ Solve eq. (13) for $\boldsymbol{\mu}_j$
9:             $\boldsymbol{R} \leftarrow \boldsymbol{R} \cup \boldsymbol{u}_{\text{Neumann}}$
10:        **end if**
11:       $\boldsymbol{S} \leftarrow \boldsymbol{S} \cup \boldsymbol{R}$
12:     **end for**
13:     $\boldsymbol{B} \leftarrow \text{POD}(\boldsymbol{S}, \varepsilon_\rho)$
14:     **return** $X^n = \text{span}(\boldsymbol{B})$
15: **end function**

2.3   Construction of a global approximation

Given local approximation spaces X_i^n for each target subdomain Ω_i^{p}, the construction of a global approximation via the GFEM is discussed. The difficulty is that the basis functions of the local spaces X_i^n and X_j^n of two neighboring subdomains Ω_i^{p} and Ω_j^{p} are not conforming on the shared interface $\Gamma_{ij}^{\text{p}} = \partial \Omega_i^{\text{p}} \cap \partial \Omega_j^{\text{p}}$. However, given a suitably[3] defined partition of unity $\{\varphi_i\}$, the local basis functions can be included in the global FE space by multiplication with the standard FE shape functions φ_i, such that a conforming approximation is obtained. The GFEM function denoted as $\boldsymbol{\psi}(\boldsymbol{x}) = \varphi(\boldsymbol{x})\boldsymbol{\xi}(\boldsymbol{x})$, $\boldsymbol{x} \in \omega_\alpha$ for a patch ω_α is shown in fig. 3 to illustrate the procedure of constructing the GFEM functions.

2.4   Global solution procedure

With the global GFEM space, each coarse grid cell (i.e. unit cell of the lattice) can be viewed as a high order FE. For instance, assuming a coarse grid partition as shown in fig. 2, the coarse

---

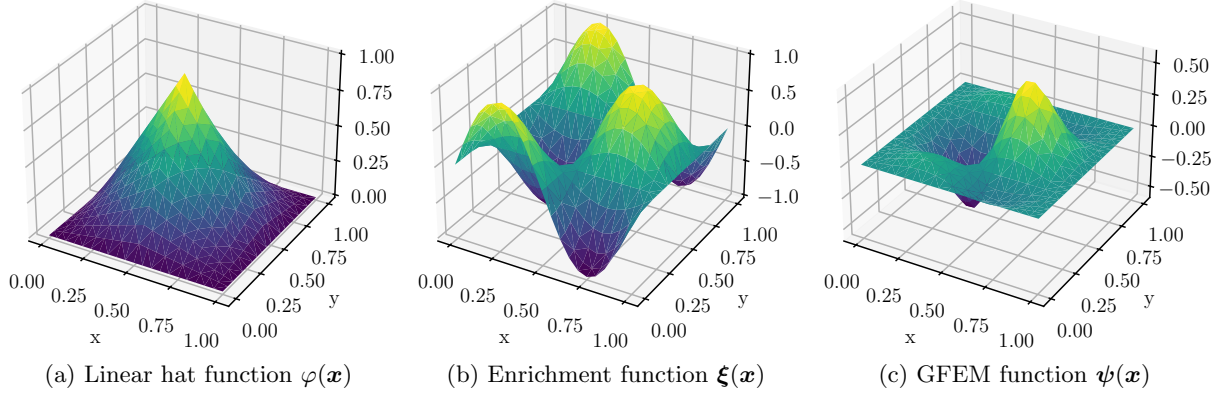[3]According to Definition 1 of [15].

7

(a) Linear hat function $\varphi(\boldsymbol{x})$ (b) Enrichment function $\boldsymbol{\xi}(\boldsymbol{x})$ (c) GFEM function $\boldsymbol{\psi}(\boldsymbol{x})$

Figure 3: Construction of GFEM functions.

grid cell is a quadrilateral cell with special ansatz functions $\boldsymbol{\psi}_i$. The components of the FE matrix of the $i-$th coarse grid cell are given by

$$
\begin{aligned}
a_{kl}^i &= \int_{\Omega_i^{\mathrm{p}}} \left(\frac{\partial \boldsymbol{\psi}_k}{\partial \boldsymbol{x}^{\mathrm{p}}} \cdot \boldsymbol{F}_{\boldsymbol{\mu}}^{-1} \right) \cdot\cdot \, \mathbb{C} \, \cdot\cdot \left(\frac{\partial \boldsymbol{\psi}_l}{\partial \boldsymbol{x}^{\mathrm{p}}} \cdot \boldsymbol{F}_{\boldsymbol{\mu}}^{-1} \right) \det(\boldsymbol{F}_{\boldsymbol{\mu}}) \, \mathrm{d}\boldsymbol{x}^{\mathrm{p}} \, , \\
&= (V^{\mathrm{T}})_{km} K_{mn}(\boldsymbol{\mu}) V_{nl} \, .
\end{aligned}
\tag{14}
$$

Here, $\boldsymbol{V}$ is the matrix that holds the FE vectors of the GFEM functions $\boldsymbol{\psi}_i$ as column vectors and $\boldsymbol{K}(\boldsymbol{\mu})$ is the stiffness matrix of the subdomain Ω_i^{p} (i.e. unit cell). Since $\boldsymbol{K}$ is dependent on the parameter $\boldsymbol{\mu}$, integration over the full fine scale mesh of the unit cell is required whenever the model is evaluated for a new parameter value which will drastically reduce the computational efficiency of our method.

To overcome this problem, the stiffness matrix $\boldsymbol{K}(\boldsymbol{\mu})$ is approximated by Empirical Interpolation (see [13, 14] and in particular [12] for the matrix version termed MDEIM employed here), such that

$$
\boldsymbol{K}(\boldsymbol{\mu}) \approx \boldsymbol{K}_Q(\boldsymbol{\mu}) = \sum_{q=1}^{Q} \theta_q(\boldsymbol{\mu}) \boldsymbol{K}_q \, .
\tag{15}
$$

Herein, $\boldsymbol{K}_q$ are parameter-independent matrices that can be precomputed[4]. The $\boldsymbol{\theta}(\boldsymbol{\mu}) \in \mathbb{R}^Q$ are called interpolation coefficients and can be determined via the interpolation equation

$$
\boldsymbol{\Phi}_{\mathcal{I}} \boldsymbol{\theta}(\boldsymbol{\mu}) = \boldsymbol{k}_{\mathcal{I}}(\boldsymbol{\mu}) \, .
\tag{16}
$$

Considering the matrix version of DEIM, the interpolation is constructed for matrices that are vectorized by stacking their columns on top of each other which is denoted by $\boldsymbol{k} = \mathrm{vec}(\boldsymbol{K})$. The suffix $(\cdot)_{\mathcal{I}}$ indicates the restricted evaluation, reduced integration respectively. The basis $\boldsymbol{\Phi} = [\mathrm{vec}(\boldsymbol{K}_1), \dots, \mathrm{vec}(\boldsymbol{K}_Q)]$ and the interpolation indices $\mathcal{I}$ are determined by applying the DEIM algorithm as proposed in [14] to a set of snapshots $\{\mathrm{vec}(\boldsymbol{K}(\boldsymbol{\mu}_1)), \dots, \mathrm{vec}(\boldsymbol{K}(\boldsymbol{\mu}_m))\}$, $m > 0$.

---

[4]Due to the Galerkin projection onto $\boldsymbol{V}$ (eq. (14)) the matrices to be precomputed and stored are also sufficiently small.

By reversing the $\text{vec}(\cdot)$ operation, the approximation $\boldsymbol{K}_Q(\boldsymbol{\mu})$ can be formed using eq. (15) once the interpolation coefficients are determined by solution of eq. (16). For more details on the overall procedure the reader is referred to [12]. Finally, with a fast method to compute $\boldsymbol{K}_Q(\boldsymbol{\mu})$, the calculation of the FE matrices a_{kl}^i of each coarse grid cell are independent of the underlying fine grid discretization of the unit cell. The assembly of the global system matrix follows the usual assembly routine for FE methods.

3 NUMERICAL EXPERIMENTS

3.1 Graded concrete slab

The numerical example investigated here is motivated by *graded concrete* in which voids are created by inserting hollow bodies into the concrete formwork with the goal of mass reduction, see https://gradientenbeton.eu/. The problem is simplified and we consider a two dimensional beam under plane stress conditions as shown in fig. 4. The coarse grid partition of half of the beam body $\Omega_{\mathrm{gl}}^{\mathrm{p}} = [0, 10a] \times [0, a]$ is given by 10 quadrilateral cells of size a, where a is the unit length of the unit cell. Each unit cell, i.e. subdomain Ω_i^{p}, is taken as a square of size a with a circular void with radius R and is partitioned in 384 isoparametric quadrilateral FEs of polynomial degree two, see fig. 2 (b). With regard to the geometrical parametrization, we consider the variation of the radius R in each subdomain. For the global problem this results in a parameter space $\mathcal{P}_{\mathrm{gl}} = [R_-, R_+]^{10}$, such that a parameter value $\boldsymbol{\mu} \in \mathcal{P}_{\mathrm{gl}}$ has parameter components $\mu_i \in [R_-, R_+], i = 1, \ldots, 10$, with ranges chosen as $R_- = 0.1a, R_+ = 0.3a$. For the parent domain we set $\bar{\mu}_i = 0.2a, i = 1, \ldots, 10$ as the reference value. The point load is distributed as a constant traction over the length of one unit cell (on the far left).

Full order model An auxiliary problem is defined on the global domain $\Omega_{\mathrm{gl}}^{\mathrm{p}}$ to account for the variation in the geometry. The full order model is given by a standard FE model using the fine grid discretization of $\Omega_{\mathrm{gl}}^{\mathrm{p}}$ as described above.

Reduced order model The ROM is constructed using the procedure for range approximation described in section 2.2. Using the coarse grid as above, we choose to solve $k = 1, \ldots, 11$ oversampling problems, such that for each of the target subdomains $\Omega_{\mathrm{in}}^{\mathrm{p},k}$ and $\Omega_{\mathrm{in}}^{\mathrm{p},k+1}$ an overlap of size $a \times a$ is created. The pointwise overlap condition is required for the construction of the GFEM functions. The size of the target subdomain $\Omega_{\mathrm{in}}^{\mathrm{p},k}$ is $2a \times a$ for $k = 2, \ldots, 10$ and $a \times a$ for $k = 1, 11$. For each problem the oversampling layer has size $a \times a$, i.e. the target subdomain is extended by a unit cell on each side unless the boundary of the target subdomain intersects with the boundary of the global domain ($\partial\Omega_{\mathrm{in}}^{\mathrm{p},k} \cap \partial\Omega_{\mathrm{gl}}^{\mathrm{p}} \neq \emptyset$).

The input parameters to algorithm 1 are $\varepsilon^* = 0.001$, $n_t = \dim(\mathcal{S})$ and $\varepsilon_{\mathrm{algofail}} = 1 \cdot 10^{-14}$ for $k = 1, \ldots, 11$, where $\dim(\mathcal{S})$ denotes the dimension of the source space of the transfer operator and corresponds to the number of degrees of freedom on the boundary $\Gamma_{\mathrm{out}}^{\mathrm{p}}$ in the FE model. The number of training samples is chosen as $N_{\mathrm{train}} = 50$ ($k = 1, 11$), $N_{\mathrm{train}} = 100$ ($k = 2, 10$) and $N_{\mathrm{train}} = 200$ elsewhere, as the local parameter spaces have different dimensions. The singular values of the POD at the end of algorithm 1 for each oversampling problem show a rapid decay (fig. 5). As expected, a less rapid decay is observed for the larger oversampling domains. Furthermore, for $k = 1, 2, 3$ in comparison to the oversampling problems of the same size on the

right ($k = 9, 10, 11$), a difference in the total number of snapshots generated is observed which is due to the additional NEUMANN problems.
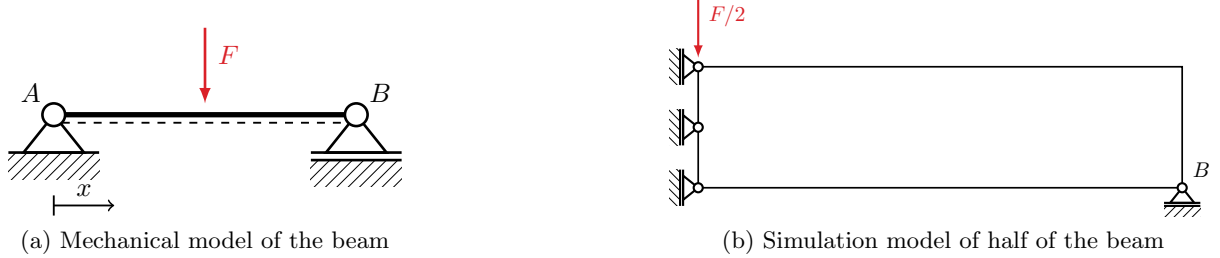

(a) Mechanical model of the beam

(b) Simulation model of half of the beam

Figure 4: Mechanical model of the beam and representation of the simulation model with loads and kinematical constraints.

3.2 Reduced order model validation

For the validation of the ROM, the relative error with respect to the FOM is computed over a parameter set of size 200. The error in the displacements is measured in the energy norm (over the domain $\Omega^{\mathrm{p}}_{\mathrm{gl}}$) and defined as

$$e_u(\boldsymbol{\mu}) := \frac{\|\boldsymbol{u}_{\mathrm{fom}} - \boldsymbol{u}_{\mathrm{rom}}\|}{\|\boldsymbol{u}_{\mathrm{fom}}\|} . \tag{17}$$

Apart from the displacement field as the primary unknown, we are interested in the first principal component of the CAUCHY stress as this will be used to define a stress constraint for the optimization problem. The error in the stress is defined as

$$e_\sigma(\boldsymbol{\mu}) = \max \frac{|\mathbf{s}_{\mathrm{fom}} - \mathbf{s}_{\mathrm{rom}}|}{\|\mathbf{s}_{\mathrm{fom}}\|_2} . \tag{18}$$

Here, $\|\cdot\|_2$ denotes the Euclidean norm and $\mathbf{s} \in \mathbb{R}^{N_g}$ denotes the vector holding the first principal stress values at the N_g integration points of the mesh. For ease of notation, the parameter-dependence is omitted for $\boldsymbol{u}$ and $\mathbf{s}$. The minimum, average and maximum of e_u and e_σ over the validation set are shown in fig. 6. With regard to the error in the displacement, the average value is much smaller than the maximum over the validation set. In both cases, for local basis size ca. > 75 the error decreases only very slowly with increasing number of basis functions. This is due to the rather moderate size of the training set, as e.g. $N_{\mathrm{train}} = 200$ is used for a local parameter space $\mathcal{P}^k = [R_-, R_+]^4$ for $k = 3, \ldots, 9$. However, the error in the stress is less sensitive to variations in the parameter $\boldsymbol{\mu}$ and a pointwise maximum relative error below 1% is acceptable for the current application.

3.3 Shape optimization

The formulation of the shape optimization problem seeks to minimize an objective function, $J(\boldsymbol{\mu}, \boldsymbol{u}(\boldsymbol{\mu}))$, with constraints, $g_q(\boldsymbol{\mu}, \boldsymbol{u}(\boldsymbol{\mu})) \leq 0$, used to limit the first principal stress at each
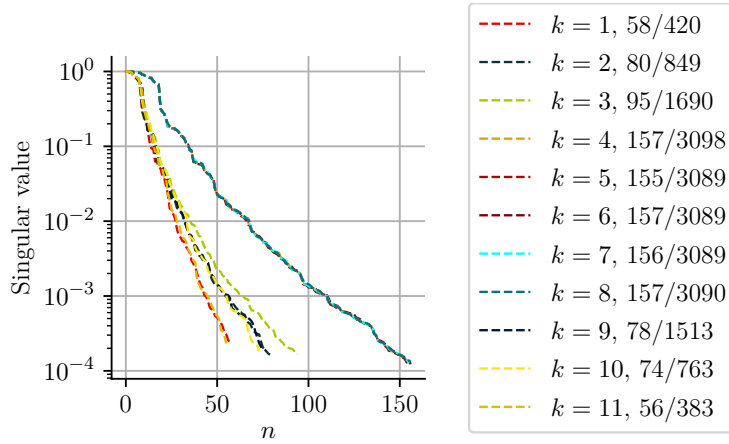
Figure 5: Decay of singular values for each of the 11 oversampling problems. The total number of basis functions and the total number of snapshots are given in the legend.
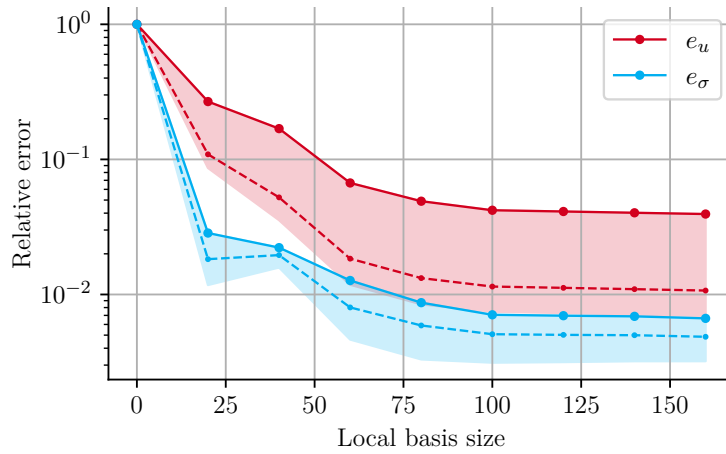

Figure 6: Minimum, average (dashed) and maximum relative error over the validation set of size 200.

Table 1: Result of the optimization. For the ROM a local basis size of 100 is used. $\boldsymbol{\mu}^*$ is used as a placeholder for the optimum found by the respective model.

Model	$J(\boldsymbol{\mu}^*)$	$m(\boldsymbol{\mu}^*)$	Iterations	Function calls
FOM	0.941	7.855	4	45
ROM	0.946	7.897	13	155

integration point, q, with $q = 1, \ldots, N_g$.

$$\min_{\boldsymbol{\mu} \in \mathcal{P}_{\mathrm{gl}}} J(\boldsymbol{\mu}, \boldsymbol{u}(\boldsymbol{\mu})),$$
$$\text{s.t.} \quad g_q(\boldsymbol{\mu}, \boldsymbol{u}(\boldsymbol{\mu})) \leq 0, \quad q = 1, \ldots, N_q, \tag{19}$$
$$\text{with} \quad \boldsymbol{K}(\boldsymbol{\mu})\boldsymbol{u}(\boldsymbol{\mu}) = \boldsymbol{f}.$$

The objective function is defined as a weighted sum of mass $m(\boldsymbol{\mu})$ and compliance $C(\boldsymbol{u}(\boldsymbol{\mu}))$.

$$J(\boldsymbol{\mu}, \boldsymbol{u}(\boldsymbol{\mu})) = (1 - \omega)\frac{m(\boldsymbol{\mu})}{m(\bar{\boldsymbol{\mu}})} + \omega\frac{C(\boldsymbol{u}(\boldsymbol{\mu}))}{C(\boldsymbol{u}(\bar{\boldsymbol{\mu}}))}, \quad 0 \leq \omega \leq 1. \tag{20}$$

The compliance is included, since we do not actively constrain the second principal component and, therefore, designs leading to high compressive stresses would not be detected. The reference values $m(\bar{\boldsymbol{\mu}})$ and $C(\boldsymbol{u}(\bar{\boldsymbol{\mu}}))$ are used to normalize. The definition of the stress constraint follows a risk factor approach [19] and is written as

$$g_q(\boldsymbol{\mu}, \boldsymbol{u}(\boldsymbol{\mu})) = \frac{\mathbf{s}_q}{\sigma_+} - 1, \quad q = 1, \ldots, N_q, \tag{21}$$

with the components of the vector of first principal stress $\mathbf{s}_q$ and an upper bound σ_+ on the admissible tensile stress. Note that $\mathbf{s}$ depends on $\boldsymbol{u}$ and thus also $\boldsymbol{\mu}$ although not explicitly written. The load F is chosen so that the parameter value with $\mu_i = R_+, i = 1, \ldots, 10$ yields $g_q > 0$ (infeasible design), such that this trivial minimum of $m(\boldsymbol{\mu})$ is excluded from the set of possible solutions. The shape optimization problem eq. (19) is then solved using the FOM and the ROM. Note that the computation of the ROM output $J_n(\boldsymbol{\mu}, \boldsymbol{u}_n(\boldsymbol{\mu}))$, with reduced displacement solution $\boldsymbol{u}_n(\boldsymbol{\mu})$, is not dependent on the dimension of the FOM. The results are obtained using sequential least squares programming [20] (SLSQP) and finite difference approximation of the gradients, and are given in table 1. The absolute error in the optimum is $\sqrt{\sum_{i=1}^{10}(\mu_i^* - \mu_{n,i}^*)^2} \approx 2.44 \cdot 10^{-2}$, with the optimum, $\boldsymbol{\mu}^*$, found by the FOM and the optimum, $\boldsymbol{\mu}_n^*$, found by the ROM, respectively. While the ROM is able to converge to a solution close to the FOM solution, over three times more iterations and function evaluations are necessary.

4 CONCLUSIONS

In this contribution a hyper-reduced localized model order reduction framework for the shape optimization of lattice structures is presented. The data compression strategy, adopted from [8], is shown to work well with range approximation by random sampling. Furthermore, the ROM

solution of the considered shape optimization problem is in good agreement with the FOM solution and this is regarded as a first proof of concept. A thorough comparison of the computational efficiency including runtime measurements is planned for future work, as this is out of the scope of the current investigation. Other directions of future research include the exploration of offline training strategies in which the variation in the parameter value $\boldsymbol{\mu}$ and the boundary data $\boldsymbol{g}$ are addressed simultaneously and suitable sampling distributions of the boundary data $\boldsymbol{g}$.

REFERENCES

[1] P. Wu, J. Wang, and X. Wang. "A critical review of the use of 3-D printing in the construction industry". In: *Automation in Construction* 68 (Aug. 2016), pp. 21–31. DOI: 10.1016/j.autcon.2016.04.005.

[2] J. Ballani et al. "A component-based hybrid reduced basis/finite element method for solid mechanics with local nonlinearities". In: *Comput. Methods Appl. Mech. Eng.* 329 (Feb. 2018). DOI: 10.1016/j.cma.2017.09.014.

[3] A. Buhr et al. "6 Localized model reduction for parameterized problems". In: *Volume 2: Snapshot-Based Methods and Algorithms.* Ed. by P. Benner et al. Berlin, Boston: De Gruyter, 2021, pp. 245–306. DOI: doi:10.1515/9783110671490-006.

[4] T. Y. Hou and X.-H. Wu. "A Multiscale Finite Element Method for Elliptic Problems in Composite Materials and Porous Media". In: *Journal of Computational Physics* 134.1 (June 1997), pp. 169–189. DOI: 10.1006/jcph.1997.5682.

[5] I. Babuska and R. Lipton. "Optimal Local Approximation Spaces for Generalized Finite Element Methods with Application to Multiscale Problems". In: *Multiscale Modeling & Simulation* 9.1 (Jan. 2011), pp. 373–406. DOI: 10.1137/100791051.

[6] K. Smetana and A. T. Patera. "Optimal Local Approximation Spaces for Component-Based Static Condensation Procedures". In: *SIAM J. Sci. Comput.* 38 (5 Jan. 2016). DOI: 10.1137/15m1009603.

[7] A. Buhr and K. Smetana. "Randomized Local Model Order Reduction". In: *SIAM J. Sci. Comput.* 40 (4 Jan. 2018). DOI: 10.1137/17m1138480.

[8] T. Taddei and A. T. Patera. "A Localization Strategy for Data Assimilation; Application to State Estimation and Parameter Estimation". In: *SIAM J. Sci. Comput.* 40 (2 Jan. 2018). DOI: 10.1137/17m1116830.

[9] K. Smetana and T. Taddei. "Localized Model Reduction for Nonlinear Elliptic Partial Differential Equations: Localized Training, Partition of Unity, and Adaptive Enrichment". In: *SIAM J. Sci. Comput.* 45 (3 June 2023). DOI: 10.1137/22m148402x.

[10] C. Himpe, T. Leibner, and S. Rave. "Hierarchical Approximate Proper Orthogonal Decomposition". In: *SIAM J. Sci. Comput.* 40 (5 Jan. 2018). DOI: 10.1137/16m1085413.

[11] T. Guo et al. "Learning constitutive models from microstructural simulations via a non-intrusive reduced basis method: Extension to geometrical parameterizations". In: *Comput. Methods Appl. Mech. Eng.* 401 (Nov. 2022). DOI: 10.1016/j.cma.2022.115636.

[12] F. Negri, A. Manzoni, and D. Amsallem. "Efficient model reduction of parametrized systems by matrix discrete empirical interpolation". In: *Journal of Computational Physics* 303 (Dec. 2015), pp. 431–454. DOI: 10.1016/j.jcp.2015.09.046.

[13] M. Barrault et al. "An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations". In: *Comptes Rendus. Mathématique* 339.9 (Oct. 2004), pp. 667–672. DOI: 10.1016/j.crma.2004.08.006.

[14] S. Chaturantabut and D. C. Sorensen. "Nonlinear Model Reduction via Discrete Empirical Interpolation". In: *SIAM J. Sci. Comput.* 32 (5 Jan. 2010). DOI: 10.1137/090766498.

[15] I. Babuška and J. M. Melenk. "The partition of unity method". In: *International Journal for Numerical Methods in Engineering* 40.4 (1997), pp. 727–758. DOI: 10.1002/(SICI)1097-0207(19970228)40:4<727::AID-NME86>3.0.CO;2-N.

[16] I. Babuška, U. Banerjee, and J. E. Osborn. "Generalized Finite Element Methods — Main Ideas, Results and Perspective". In: *International Journal of Computational Methods* 01.01 (June 2004), pp. 67–103. DOI: 10.1142/s0219876204000083.

[17] M. Botsch et al. *Polygon Mesh Processing.* A K Peters/CRC Press, Oct. 2010. DOI: 10.1201/b10688.

[18] B. Haasdonk. "Chapter 2: Reduced Basis Methods for Parametrized PDEs—A Tutorial Introduction for Stationary and Instationary Problems". In: *Model Reduction and Approximation.* Society for Industrial and Applied Mathematics, July 2017, pp. 65–136. DOI: 10.1137/1.9781611974829.ch2.

[19] T. Pastore et al. "Topology optimization of stress-constrained structural elements using risk-factor approach". In: *Computers & Structures* 224 (Nov. 2019), p. 106104. DOI: 10.1016/j.compstruc.2019.106104.

[20] D. Kraft. *A software package for sequential quadratic programming.* Tech. rep. DFVLR-FB 88-28. Köln, Germany: DLR German Aerospace Center – Institute for Flight Mechanics, 1988.