RESEARCH ARTICLE

WILEY

A finite element reduced-order model based on adaptive mesh refinement and artificial neural networks

Joan Baiges[1] | Ramon Codina[1,2] | Inocencio Castañar[1] | Ernesto Castillo[3]

[1]Universitat Politècnica de Catalunya, Barcelona, Spain

[2]Centre Internacional de Mètodes Numèrics a l'Enginyeria (CIMNE), Barcelona, Spain

[3]Departamento de Ingeniería Mecánica, Universidad de Santiago de Chile, Santiago, Chile

Correspondence
Joan Baiges, Universitat Politècnica de Catalunya, Jordi Girona 1-3, Edifici C1, 08034 Barcelona, Spain.
Email: joan.baiges@upc.edu

Summary

In this work, a reduced-order model based on adaptive finite element meshes and a correction term obtained by using an artificial neural network (FAN-ROM) is presented. The idea is to run a high-fidelity simulation by using an adaptively refined finite element mesh and compare the results obtained with those of a coarse mesh finite element model. From this comparison, a correction forcing term can be computed for each training configuration. A model for the correction term is built by using an artificial neural network, and the final reduced-order model is obtained by putting together the coarse mesh finite element model, plus the artificial neural network model for the correction forcing term.

The methodology is applied to nonlinear solid mechanics problems, transient quasi-incompressible flows, and a fluid-structure interaction problem. The results of the numerical examples show that the FAN-ROM is capable of improving the simulation results obtained in coarse finite element meshes at a reduced computational cost.

KEYWORDS

adaptivity, artificial neural network, finite element methods, reduced-order model

1 | INTRODUCTION

Numerical simulation is nowadays being routinely used in design and optimization in engineering and other scientific fields. However, many times these optimization and design processes are limited by the amount of available computational power: in the absence of other tools and analysis, numerical simulation is often insufficient in itself to address complex physics.[1] The amount of data that comes from a nonlinear evolution and coupled problems could be difficult to manage and, obviously, highly expensive to solve, even with the use of good mesh generators, discretization schemes, and solution algorithms.[2] For accurate simulations, typical finite element codes (for instance) may require several thousands of degrees of freedom, which many times results in the need for using costly supercomputing facilities.[3-5] In order to overcome this problem, the scientific community has been putting a lot of effort in the development of the so-called reduced-order models (ROMs) in recent years, which has been applied to practically all the areas of computational mechanics. The ROMs aim at reproducing, at a low computational cost, the features and accuracy of full-order models (FOMs), with models with only a few degrees of freedom, but which have an accuracy similar to the FOM.

The idea to reduce the dimensions of the problem has been a field of intense research in the last years. Examples of dimension reduction approaches include separated representations in the framework of proper generalized

decompositions,[6,7] sparse grid techniques,[8,9] reduced basis, and model-order reduction based on the proper orthogonal decomposition method (POD)[10-15] or in the centroidal Voronoi tessellation method,[2,16] among others. The advantages and disadvantages of each approach depend on the specific problem.

The combined usage of high-fidelity models and ROMs is known as multifidelity models. In problems were multiple simulations are required, such as optimization and uncertainty quantification, using the expensive high-fidelity models for some of the functional evaluations and less expensive reduced or low-fidelity models can help in alleviate the total computational cost of the process.[17] Several families of multifidelity models exist, which include surrogate models,[18-20] hierarchical multifidelity approaches.[21-23] These can be based on several approaches, which include response surfaces[24-26] and Kriging methods.[27-29] The method proposed here can be understood, and used as a multifidelity approach, although it will not be applied as such in this work.

The approach presented in this work departs from the ideas presented in the work of Baiges et al.[30] In that work, a reduced-order subscales model for POD was presented, where the subscales were defined as a linear function of the solution of the ROM. The coefficients of this linear function were obtained by comparing the solution of the FOM with the solution of the ROM for the same initial conditions. The main idea was to use a high-fidelity simulation and compare the result with that of a POD-based ROM. The difference between both yielded a correction term, which could, firstly, be modeled through a least squares approach, and secondly, be added to the ROM system of equations as a correction term. This correction greatly enhanced the performance of the ROM model, allowing to simulate complex flow behavior and to improve the results when using subintegration through hyperreduction.[31,32]

Similar ideas have been exploited in recent works. In the work of Rahman et al,[33] a dynamic closure modeling approach has been proposed to stabilize the projection-based ROMs in turbulent flows, proving to be a promising technique for different flow conditions. The key idea of that method consists in using an eddy viscosity closure approach to stabilize the resulting surrogate model considering the analogy between large eddy simulation and truncated modal projection. In the same line of closure problem, in the work of Xie et al,[34] a data-driven filtered ROM method is proposed for the numerical approximation of CFD problems. In that work, the closure of the filtered ROM problem is enforced in two steps using a quadratic ansatz to model the interaction between the resolved and unresolved modes. In the same line, in the work of Mohebujjaman et al,[35] the authors propose a data-driven correction ROM method to ensure the closure of the ROM, enforcing that the ROM problem satisfies the same physical laws that the full-order problem.

A well-known weakness of ROM methods is that they can be neither robust with respect to parameter changes nor cost-effective for handling the nonlinear dependence of complex dynamical systems.[36] On this regard, machine learning frameworks have shown to be a promising tool to enhance the robustness of this numerical technique. In the work of San and Maulik,[37] the effect of discarded modes is taken into account using a machine learning architecture based on a single hidden layer feedforward artificial neural network (ANN). This approach allows the authors to achieve robust stabilization with respect to parameter changes in thermally coupled flows. In that work, the two-dimensional differentially heated cavity flow is used to demonstrate the capabilities of the proposed method in stabilizing the temporal mode evolution that characterize the physical problem. In the same framework, in the work of Wang et al,[38] the authors propose a nonintrusive reduced-basis method for parametrized unsteady flows, where a set of reduced basis functions are extracted from a collection of high-fidelity snapshots via POD, using additionally a feedforward neural network to recover the coefficient of the reduced-order basis functions in a combustion problem. In turbulent flows, similar procedures have been proposed based on multikernel neural networks[39] and the extreme learning machine concept,[40,41] which provide a significant reduction in computational time and effectively retain the dynamic of the FOM during the forward simulation period beyond the training data set.

In this work, these ideas are extended to the setting of **F**inite element ROMs and **A**daptive mesh refinement through artificial **N**eural networks (FAN-ROM): instead of solving a large-scale finite element simulation in order to train a POD model, the efforts are going to be focused in two finite element models, which depart from a coarse finite element mesh. The first model, the high-fidelity model, is going to be adaptively refined using an error estimator. This model will be computationally expensive (although still less costly than a uniformly refined mesh model), and it will be the reference FOM model. The second model is going to be the finite element model in the original coarse mesh. In order to enhance the performance of the low cost model, a correction term is going to be trained by comparing the results of the FOM and ROM models in a set of training configurations, and using an ANN. This correction term will be added to the ROM finite element equations, which will now yield a result close to the FOM model.

Several advantages with respect to other methodologies can be observed in this approach: firstly, although any highly accurate methodology could be used for the high-fidelity model, through the usage of hierarchical adaptive mesh refinement, the cost of the high-fidelity model is kept as low as possible. These high-fidelity simulations can also be run in

parallel, which helps to alleviate the training computational cost. Secondly, there is no need to store snapshots for the training of the ROM because the ANN can be continuously trained. In addition, the ANN is trained on the coarse model degrees of freedom, so the ANN training computational cost is relatively low. Finally, there is no need to build two different approaches for the ROM and the FOM, in both cases a finite element model is used, with a minor modification in the case of the ROM.

This paper is organized as follows. In Section 2, the proposed ROM methodology is presented for a generic problem. Firstly, a least squares approach is described, and secondly, the final FAN-ROM method by using ANNs is explained. In Section 3, the methodology is tested in different problems: a nonlinear large strain solid mechanics simulation, a nearly incompressible flow problem, and a fluid-structure interaction model. Finally, some conclusions close this paper in Section 4.

2 | PROPOSED METHODOLOGY

In order to illustrate the proposed method, let us suppose that we want to solve a simple linear, stationary problem. This would be the case, for instance, of the nontransient heat transfer problem, which yields the Poisson equation. Moreover, let us suppose that we want to solve it in a given domain, but in a large number of different configurations, which depend on the boundary conditions, the external source terms, or the material properties. If an accurate solution is required in each of these configurations, a simple possible approach is to run a simulation for each set of boundary and source term conditions by using a fine mesh finite element model. However, this methodology would be expensive since each of the executions requires of significant computational effort.

The proposed strategy is based on executing the fine mesh simulations only in a small set of configurations, and then extract the appropriate information from these high-fidelity solutions in order to train a ROM. This ROM will be used to accurately approximate the solution in the rest of the configurations. The idea is similar to what is done in POD. However, in the present methodology, the ROM is simply a coarse finite element model, which will be enhanced by using an ANN.

In order to train the ROM, the method consists in building two models of the same physical problem in the so-called training configurations. The first model is an adaptively refined finite element mesh, which takes as starting point a coarse mesh. An error estimator is used in order to decide which areas of the finite element mesh need to be recursively refined (see the works of Badia and Baiges,[42] Baiges and Bayona,[43] and Baiges et al[44] for the strategy employed by our group). The resulting model is supposed to be very accurate but also expensive to execute. The second model is the ROM, which consists simply in the original coarse finite element model, without any further refinement, enhanced by means of an ANN, as it will be explained. Consequently, it is very cheap to execute.

Let us depart from the nonenhanced coarse and the fine models. For the considered physical problem, each finite element discretization, coarse and adaptively refined, yields a bilinear form, and an algebraic system can be built from it. Let us represent the resulting variational problem as

$$B\left(u_h^c, v_h^c\right) = L\left(v_h^c\right) \quad \forall v_h^c \in V_h^c, \tag{1}$$

for the coarse mesh, where $u_h^c \in V_h^c$ is the coarse mesh solution and $v_h^c \in V_h^c$ are the corresponding finite element test functions, V_h^c being the coarse finite element space. $B(\cdot, \cdot)$ is the variational problem bilinear form, and $L(\cdot)$ is its corresponding right hand side. We also denote by m the number of finite element nodes of the coarse mesh. Equivalently, for the refined mesh, the corresponding problem is

$$B\left(u_h^f, v_h^f\right) = L\left(v_h^f\right) \quad \forall v_h^f \in V_h^f, \tag{2}$$

where now $u_h^f \in V_h^f$ is the fine adaptive mesh solution and $v_h^f \in V_h^f$ are the corresponding finite element test functions, V_h^f being the adaptively refined finite element space. We also denote by M the number of nodes of the fine finite element mesh. It is clear that the solution to both problems is different, and that the fine mesh will yield a more accurate result. The objective now is to be able to modify the coarse mesh system so that an improved solution is obtained.

Let us start by defining a projection operator onto the coarse mesh finite element space V_h^c, $P_h^c(\cdot)$. Note that, when using hierarchical adaptive mesh refinement, this projection operation can be computed easily since the coarse nodes of both meshes coincide. The next step is to define a model enhancement function. For each configuration i (given by the

boundary conditions and external source terms), we denote u_h^c as the solution to (1), and u_h^f as the solution to (2). Then, the model enhancement function is defined as

$$C_i \left(P_h^c \left(u_h^f \right), v_h^c \right) := B_i \left(P_h^c \left(u_h^f \right), v_h^c \right) - B_i \left(u_h^c, v_h^c \right)$$

$$= B_i \left(P_h^c \left(u_h^f \right), v_h^c \right) - L_i \left(v_h^c \right).$$

Now, the solution to

$$B_i \left(u_h, v_h^c \right) - C_i \left(u_h, v_h^c \right) = L_i \left(v_h^c \right) \quad \forall v_h^c \in V_h^c$$

is precisely $u_h = P_h^c(u_h^f)$, that is, the projection of the fine mesh solution onto the coarse mesh space. The conclusion is that if we are capable of building a model for $C_i(u_h, v_h^c)$, we will be able to improve the solution of the coarse mesh model and obtain a cheap, accurate ROM.

2.1 | Least squares approach

The first approach to build C_i is to use a least squares method. For a given configuration i, the coarse mesh algebraic form of (1) is written as

$$A_i U_i^c = F_i,$$

where A_i is the algebraic matrix form of the bilinear form B for configuration i (determined by a set of boundary conditions), U_i^c is the coarse, uncorrected vector of unknowns, and F_i is the algebraic counterpart of L for configuration i (given by a set of external forces). The objective is to add a vector D_i such that the following equation holds:

$$A_i U_i^{cf} + D_i = F_i,$$

where D_i is the algebraic counterpart of the model function C_i, evaluated at configuration i. U_i^{cf} is the algebraic counterpart of $P_h^c(u_{h,i}^f)$, that is, the coarse mesh projected, fine mesh solution at the given configuration. In the least squares methodology, the model for D_i will be linearly dependent with the algebraic solution U_i^{cf}

$$D_i \approx A_D U_i^{cf} - F_D.$$

The corrected ROM is now

$$(A_i + A_D)U = F_i + F_D. \tag{3}$$

The question now is how to build A_D and F_D. One of the requirements for matrix A_D is that the graph of the matrix must not be denser than the graph of matrices A_i. Due to this, matrix A_D is forced to have the same connectivity pattern as matrices A_i, that is, the correcting term for a given matrix node will depend only on the values of the unknown in the first layer of surrounding nodes of the coarse finite element mesh, that is, nodes which share an element. Let us suppose that there exist N training configurations. Then, matrices A_D and F_D can be computed as

$$A_D, F_D = \arg \min_{a \in \mathbb{R}^{m \times m}, b \in \mathbb{R}^m} \sum_{i=1}^{N} \left\| D_i - \left(a U_i^{cf} - b \right) \right\|^2,$$

which yields a classical least squares problem. Note that the correction for a node j (corresponding to the coefficients in row j of A_D and the jth coefficient of F_D) can be computed independently from the coefficients for any other node.

2.2 | Artificial neural network approach

The least squares approach is convenient for linear, stationary problems where the error of a given system does not propagate in successive simulations. However, when applied to nonlinear transient problems, it can lead to an unstable behavior of the ROM because there is no upper or lower limit to the correction it can introduce. A similar behavior was also observed in the work of Baiges et al,[30] where a model filtering was required to stabilize the least squares correction model. This is the reason why an ANN to approximate correction vectors D_i is proposed. Despite ANN models are able to represent much more complex behaviors than a simple linear least squares approach, they present the inconvenience of

being nonlinear, and also more expensive to compute. This means that it is not possible to build an implicit model such as (3), and this issue makes the approach less suitable for linear stationary problems. On the contrary, in nonlinear or transient problems (where each time step can be associated to a configuration), an explicit model based on an ANN can be built. In this case, the correction vector for a given configuration is approximated as

$$D_i \approx F_D\left(U_i^{cf}\right),$$

where $F_D(U_i^{cf})$ represents a trained neural network model, which takes as an input U_i^{cf} and as an output the D_i vector approximation.

Computational effort is one major concern for this model, and this is the reason why building a very complex neural network with $\mathcal{O}(m)$ inputs and $\mathcal{O}(m)$ outputs is avoided, which would have a training and evaluation cost of $\mathcal{O}(m^2)$. Instead, $\mathcal{O}(m)$ neural networks are trained, one associated to each degree of freedom of the finite element coarse mesh, and each of these neural networks takes as input the values of the unknown in the degrees of freedom in the first layer of surrounding nodes of the coarse finite element mesh. The cost of training the ANN is of $\mathcal{O}(m \times n_{\text{neigh}})$, where n_{neigh} is the mean number of first layer neighbors for each node of the finite element mesh. This means that the computational cost is of the correct order, although the training of artificial networks is in general considerably expensive. On the other hand, the training can easily be parallelized, and it is performed in the off line phase, together with the execution of the high-fidelity model.

The resulting coarse mesh ROM is as follows:

$$A_i U_i = F_i + F_D(U_{i-1}),$$

where now i represents not only a configuration counter but also a nonlinear iteration or time step counter, and the correction for the model is evaluated in the previous nonlinear iteration or time step, denoted by $i - 1$. This approach assumes that close to nonlinear convergence or when the time step is small enough, then

$$F_D(U_i) \approx F_D(U_{i-1}),$$

and then the approximation is appropriate. For the neural network model, classical neural networks with sigmoid functions have been adopted in this work, although other possibilities may be explored. The used implementation is the one in the open source library FANN (see the work of Nissen[45]).

3 | NUMERICAL EXAMPLES

In this section, some numerical examples are presented, which illustrate the behavior of the proposed methodology. The first example deals with a large strain, dynamic, solid mechanics problem. Secondly, the method is applied to a nearly incompressible finite element flow problem. Finally, both ROMs are put together in a fluid-structure interaction setting.

In all of the numerical examples, the same type of ANN was used: for each ANN corresponding to a node and degree of freedom of the coarse mesh model, the input data were the nodal values at the surrounding nodes. The ANN was composed of three layers, with seven hidden cells, and a desired relative error of 0.001 was enforced for the ANN model, with a maximum of 150 evaluations in the training process. Sigmoid functions were used for the ANN models.

3.1 | Large strain, dynamic, solid mechanics problem

In this numerical example, we use a nonlinear, dynamic, solid mechanics simulation and we test the FAN-ROM approach on it. A Saint-Venant Kirchhoff material is considered, which in spite of being one of the simplest nonlinear materials, it allows to illustrate the performance of the methodology in a large strain setting. The equations that govern this problem are

$$\rho \partial_t \boldsymbol{u} = \nabla \cdot \boldsymbol{\sigma} + \rho \boldsymbol{b},$$
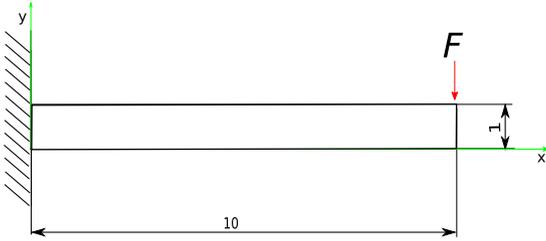$$\rho J = \rho_0,$$

FIGURE 1 Geometry and load for the dynamic, large strains beam example [Colour figure can be viewed at wileyonlinelibrary.com]
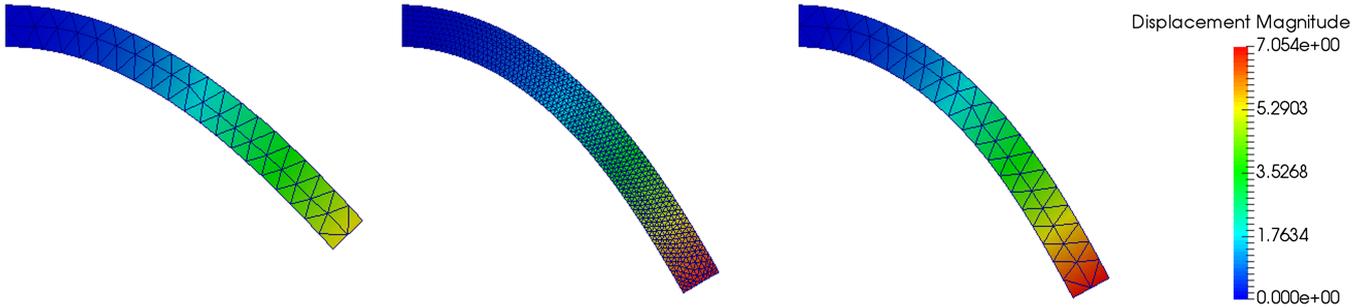

FIGURE 2 Static training of the coarse mesh model. From left to right, displacement field for the coarse, fine, and coarse trained models

where

$$F = \frac{\partial x}{\partial X}, \quad J = \det(F),$$

where ρ is the solid density (ρ_0 the initial solid density), u is the velocity field, σ is the Cauchy stress tensor, b is the vector of external forces, F is the deformation gradient tensor, and J its determinant. The constitutive equation for the Saint-Venant Kirchoff material, which relates displacements and stresses, is as follows:

$$S = \lambda \text{tr}(E)I + 2\mu E,$$

where S is the second Piola-Kirchhoff stress tensor, which is related to the Cauchy stress tensor through the following expression:

$$S = JF^{-1} \cdot \sigma \cdot F^{-T},$$

and E is the Green-Lagrange deformation tensor. For the solution of this problem, a large strain finite element formulation together with a Newton-Raphson linearization scheme is used.

The geometry of the considered beam and the load applied to it are depicted in Figure 1. The material parameters are taken as Young modulus $E = 1000$, and Poisson ratio $\nu = 0.21$.

The first test that we perform on this case is the training of a coarse mesh finite element model by using a static case. For this, we solve 25 static simulation steps in which the vertical load is progressively increased from 0 to a value of 2.5. The solution for each of these steps is going to be used in order to train a coarse mesh model by using the FAN-ROM approach, the characteristics of the ANN-models being the ones specified at the beginning of this section. The time for training the ANN models was approximately 8% of the time required to run the fine mesh model, whereas the overhead of running the coarse model including the ANN correction with respect to the coarse model alone was of 90%. Despite the increased computational cost for the coarse model, the usage of the ANN model is convenient as we will see in the following.

Figures 2 and 3 show the comparison of the solution for the static case between the coarse mesh (78 elements) and a refined mesh (in this case, two levels of uniform refinement are used, resulting in 1248 elements). Clearly, the coarse mesh is too dissipative, which results in total displacements much smaller in the coarse mesh (4.2 vertical tip displacement) than in the fine one (5.6 vertical tip displacement). However, the same coarse mesh with the reduced-order correction performs much more similarly to the fine mesh (5.5 vertical tip displacement), and dissipation practically disappears, as shown in Figure 3.

The next test that we perform is to use the coarse mesh statically-trained model in a dynamic setting and to evaluate its performance. For this, a constant vertical load of modulus 1 is applied, and we compare the tip displacement of the
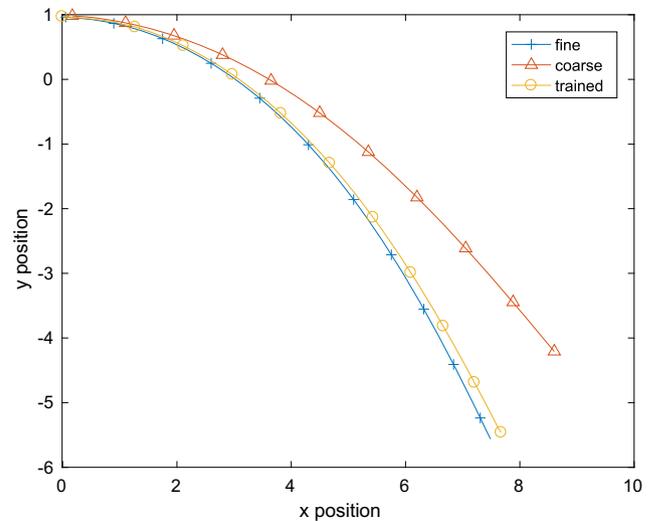
FIGURE 3 Displacement field in the upper surface of the beam for the coarse, fine, and trained models [Colour figure can be viewed at wileyonlinelibrary.com]
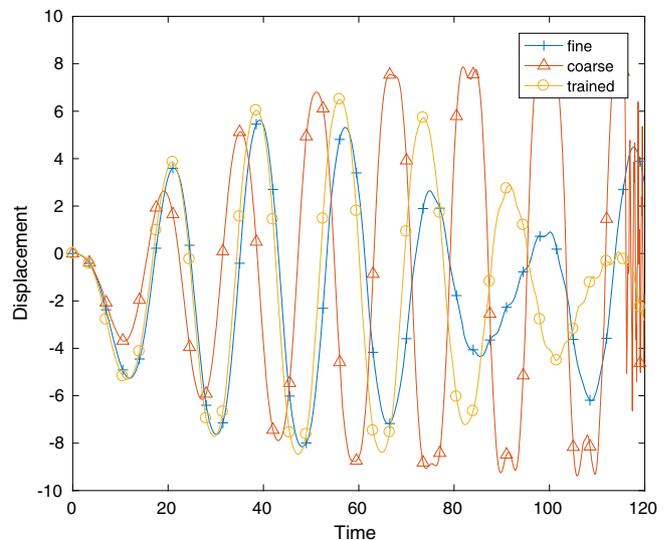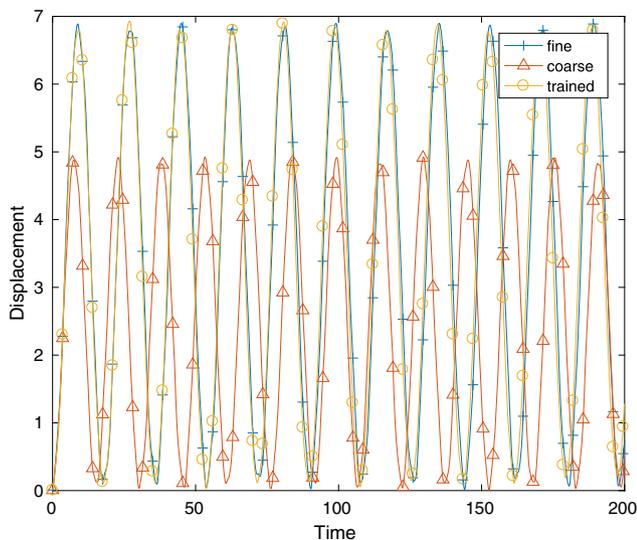

FIGURE 4 Application to a dynamic case. Left: static load. Right: close to resonant dynamic load [Colour figure can be viewed at wileyonlinelibrary.com]

coarse mesh model, the coarse mesh model statically trained with the FAN-ROM methodology, and the fine mesh model (Figure 4, left). The coarse model shows an important error both in amplitude (5.0 maximum tip displacement) and frequency. The FAN-ROM model, on the contrary, performs very similarly to the fine model (both with approximately 6.9 maximum tip displacement), with almost no error in the oscillation frequency, and only a slight error in the oscillation amplitude. Regarding computational efficiency, the speedup of the FAN-ROM model with respect to the fine mesh model is of 11.7, which makes the proposed strategy very convenient for this problem. Let us emphasize again that the training of the FAN-ROM model was done in the static setting, and then used in this dynamic configuration.

In the next example, we solve the same dynamic problem with a dynamic load instead of a constant one. The period of the dynamic load is approximately 0.8 times the resonance period of the structure according to the fine mesh results. Figure 4, right, shows the comparison of the fine, coarse, and (statically trained) FAN-ROM results. The coarse mesh model rapidly separates from the fine mesh results, and after some simulation steps, the solution explodes. The FAN-ROM, on the contrary, with virtually the same computational cost, is capable of capturing the general behavior of the structure, although a phase difference accumulates through time integration.

3.2 | Fluid mechanics problem

In this numerical example, we solve a flow problem in a close to incompressible regime. The governing equations are

$$\rho \partial_t \boldsymbol{u} + \rho \boldsymbol{u} \cdot \nabla \boldsymbol{u} - \mu \Delta \boldsymbol{u} + \nabla p = \rho \boldsymbol{f}$$
$$\nabla \cdot \boldsymbol{u} + \epsilon \partial_t p = 0,$$

where ρ is the fluid density, μ is the viscosity, p is the pressure, $\boldsymbol{f}$ is the vector of body forces, and ϵ is an artificial compressibility parameter. The incompressible limit corresponds to $\epsilon = 0$. As it is well known, the incompressible flow limit gives rise to a saddle point problem (in the stationary and linear case). Although, in our group, we have extensively worked on stabilized formulations for this type of flows (see, for instance, the review in the work of Codina et al[46]), in this work, the proposed ROM turns out to behave in an unstable manner when used together with a purely incompressible formulation, even if stabilization is used. This is the reason why in this numerical example, a simpler problem where small values of ϵ are used is solved, which is known as an artificial compressibility approach. For the solution of this problem, the stabilized finite element formulation presented in the works of Codina[47,48] is used.

In this numerical example, we study the flow around a cylinder at Re = 100. The computational domain consists of a 16×8 rectangle with a unit-diameter cylinder centered at $(4, 4)$. The horizontal inflow velocity is set to 1 at $x = 0$. Slip boundary conditions that allow the flow to move in the direction parallel to the walls are set at $y = 0$ and $y = 8$, and velocity is set to 0 at the cylinder surface. The viscosity has been set to $\mu = 0.01$, and the density at $\rho = 1$, which yields a Reynolds number Re = 100 based on the diameter of the cylinder and the inflow velocity. A second-order backward differences scheme has been used for the time integration with time step $\delta t = 0.01$. The problem setting is depicted in Figure 5. In this example, the coarse mesh consists of a 7200 linear element mesh, whereas the fine mesh has approximately 33 000 adaptively refined elements. A comparison of the coarse and fine meshes is depicted in Figure 6. As in the previous example, the information obtained by using the adaptively refined mesh is used as an input to train the FAN-ROM model on the coarse mesh, the characteristics of the ANN models being the ones specified at the beginning of this section. The time for training the ANN models was approximately 23% of the time required to run the fine mesh model, whereas the overhead of running the coarse model including the ANN correction with respect to the coarse model alone was of 15%.
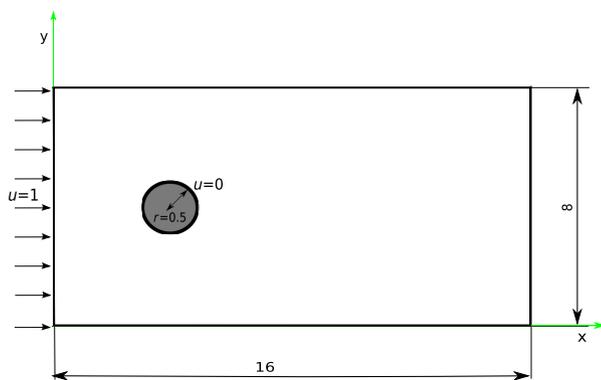

FIGURE 5   Flow around the cylinder geometry and boundary conditions [Colour figure can be viewed at wileyonlinelibrary.com]

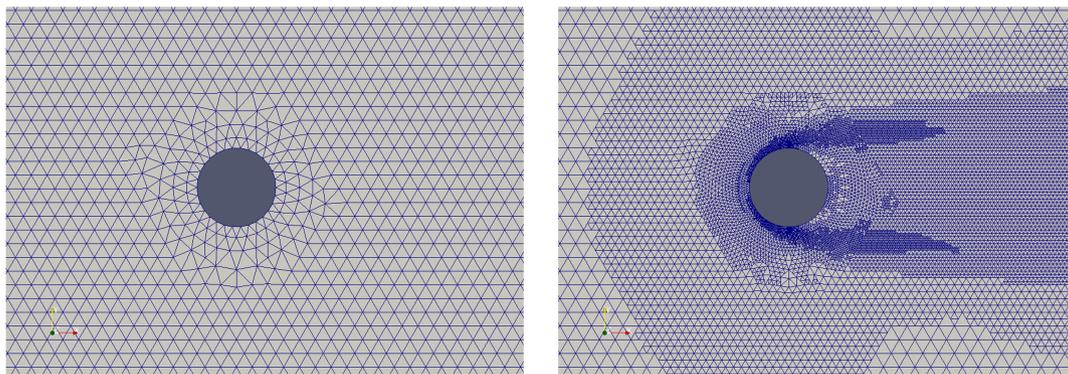FIGURE 6   Original and adaptively refined meshes for the flow past a cylinder case [Colour figure can be viewed at wileyonlinelibrary.com]
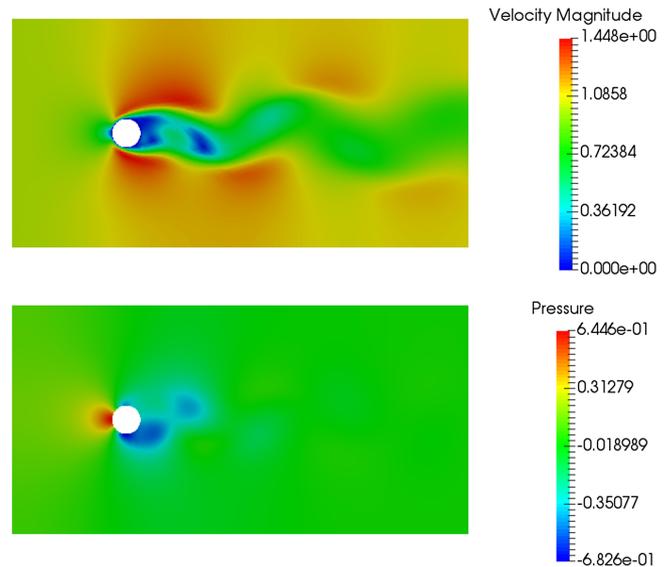
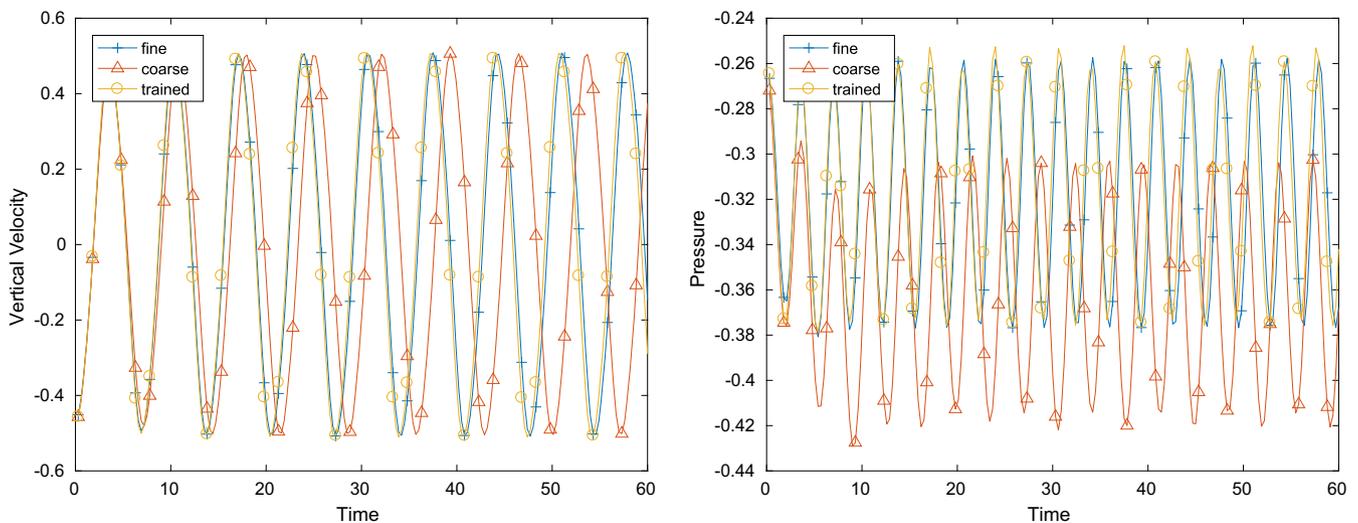**FIGURE 7** Velocity and pressure fields for the solution of the flow past a cylinder case

FIGURE 8 Comparison of velocity and pressure time histories at point (6,4) for the fine, coarse, and FAN-ROM models [Colour figure can be viewed at wileyonlinelibrary.com]

Regarding the adaptive refinement process, the cost of dynamically adapting the fine mesh model was 27% of the total computational cost of running the fine mesh model.

Figure 7 shows the instantaneous velocity and pressure fields; note that the problem is transient and nonstationary.

Figure 8 shows a comparison of the vertical velocity component and the pressure value at a point in the wake of the cylinder, for the coarse, adaptively refined, and coarse trained meshes. While the coarse mesh significantly departs from the fine model result, specially regarding the oscillation frequency and the absolute value of the pressure, the coarse trained model is able to perfectly recover the values obtained with the adaptively refined model.

Figure 9 shows a comparison of the drag and lift time histories, for the coarse, adaptively refined, and coarse trained meshes. Again, the trained model is much more accurate than the coarse model without any training, although in this case the correction still yields some error in the drag computation (mean drag value if 0.755 for the fine model, 0.821 for the coarse model, and 0.773 for the trained model). Regarding computational efficiency, the speedup of the FAN-ROM model with respect to the fine mesh model is of 9.74.

Despite the nice results obtained in this simple Re = 100 example, we must emphasize that when applied to the same problem with Re = 10, 000, we were not able to obtain reasonable results for the trained coarse model. Our guess is that this is possibly due to the impossibility of the coarse mesh of representing the fine mesh solution with a minimum of
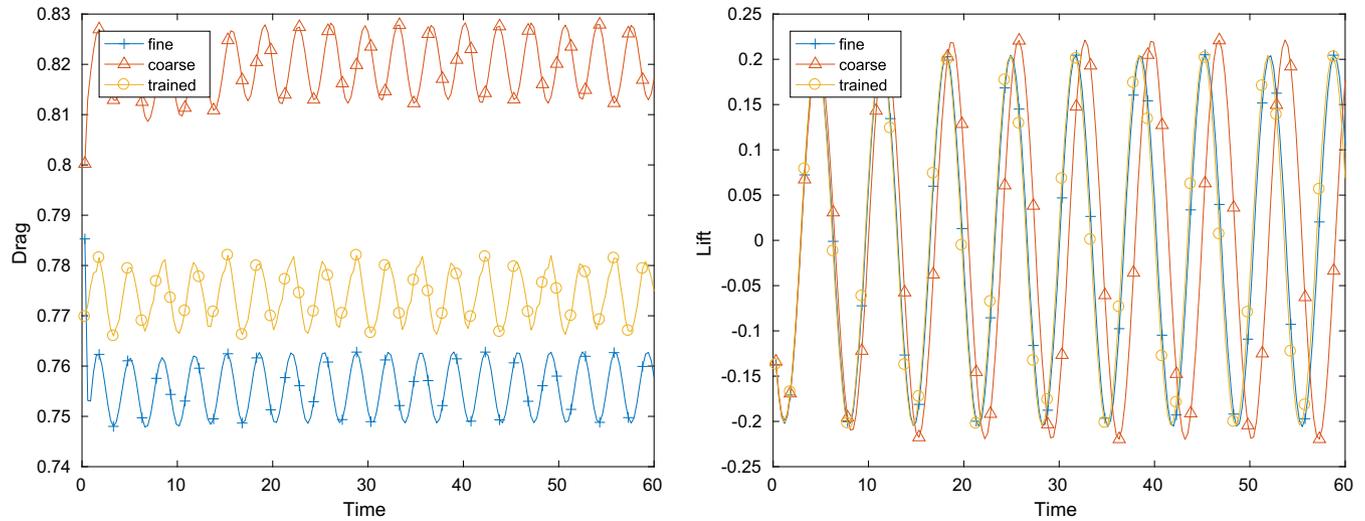
FIGURE 9 Comparison of drag and lift time histories for the fine, coarse, and FAN-ROM models [Colour figure can be viewed at wileyonlinelibrary.com]
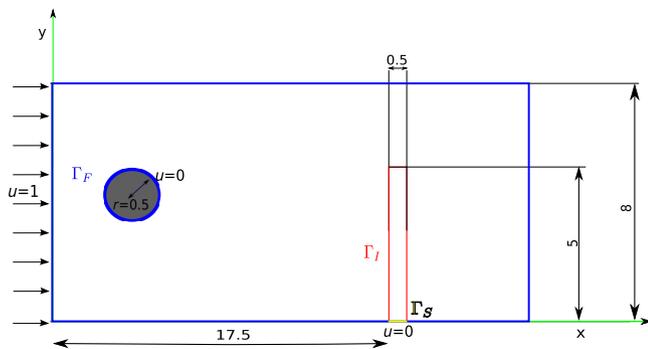

FIGURE 10 Geometry and boundary conditions for the fluid structure interaction (FSI) problem

accuracy. This means that possibly finer, more expensive to train, coarse models would be required for complex problems. This issue poses a limitation for the proposed methodology, which needs to be addressed in the future.

3.3 | Fluid structure interaction problem

A simple but illustrative fluid-structure interaction example is presented in this section. The example consists in putting together the ROMs of Sections 3.1 and 3.2, *without any further training*, and solving a fluid-structure interaction problem: an incompressible fluid flows past a cylinder and when the wake of the cylinder impacts in a vertical beam, it causes it to oscillate. This is a quite complex problem to solve because the coupling between fluid and structure is bidirectional. The considered equations for each of the problems are the ones in Section 3.1 for the structural part and the equations in Section 3.2 for the fluid mechanics part, supplemented with an Arbitrary-Lagrangian-Eulerian frame of reference for the fluid domain (see the work of Donea et al[49]). The material properties and boundary conditions are the same as in the previous examples. Two additional equations at the fluid-solid interface that enforce the continuity of tractions and velocities need to be applied

$$\boldsymbol{u}_S = \boldsymbol{u}_F, \quad \text{on } \Gamma_I,$$
$$\boldsymbol{n} \cdot \boldsymbol{\sigma}_S = \boldsymbol{n} \cdot \boldsymbol{\sigma}_F, \quad \text{on } \Gamma_I,$$

where $\boldsymbol{n}$ is the external normal to each domain, and the S and F subindexes represent solutions in the solid and fluid domain, respectively, Γ_I being the interface surface between both subdomains. In our partitioned coupling approach, the first condition is enforced strongly on the fluid domain at each iteration, whereas the second condition is enforced weakly on the solid domain, this is classically known as a Dirichlet-Neumann approach. Several iterations are required at each time step in order to reach the converged solution, in each of the iterations, the position of each of the domains is updated.
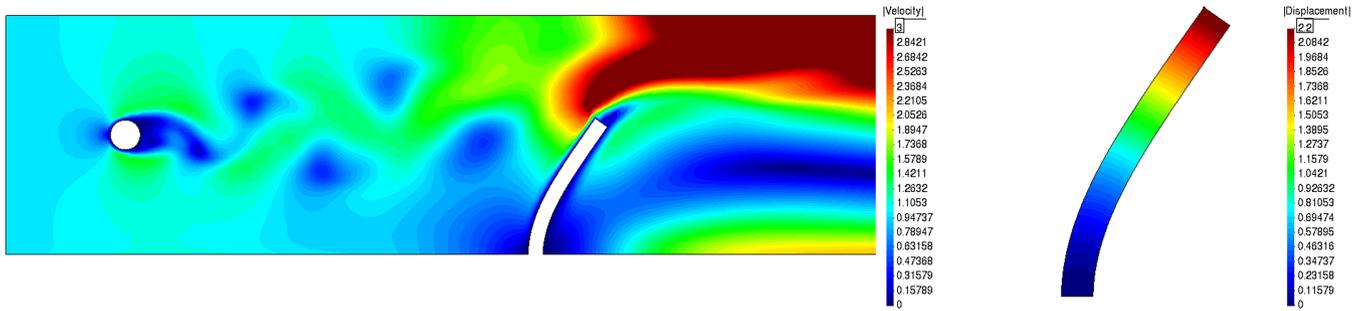
FIGURE 11 Instantaneous velocity and displacement fields for the fluid and solid domains
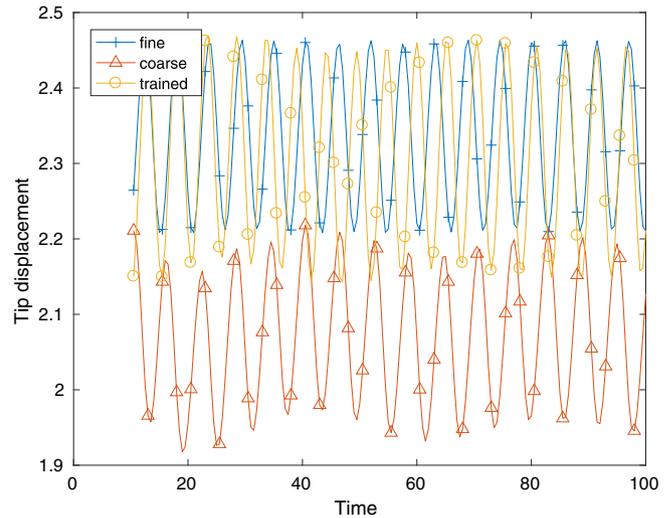

FIGURE 12 Comparison of the time history of the beam tip displacement for the coarse, fine, and FAN-ROM reduced-order model [Colour figure can be viewed at wileyonlinelibrary.com]

The solution of each problem is done by using a finite element formulation, in the case of the fluid mechanics problem, stabilization is added to the Galerkin finite element method as explained in Section 3.2.

The geometry and boundary conditions for this problem are shown in Figure 10. Figure 11 shows the instantaneous velocity and displacement profiles in the fluid and solid domains. The comparison between the coarse, fine, and FAN-ROM results is shown in Figure 12. The time history of the displacement at the upper tip of the vertical beam is plotted for each model, after the dynamic stationary state is reached. The coarse model is quite diffusive and underestimates the amplitude and mean value of the displacements (with tip displacement values ranging from 1.92 to 2.3). The FAN-ROM model, on the contrary, is capable of better reproducing the dynamic behavior (with tip displacement values ranging from 2.15 to 2.47, compared to values ranging from 2.2 to 2.47 for the high-fidelity model), although it slightly overestimates the amplitude and frequency of the oscillations.

The conclusion of this numerical example is that the FAN-ROM is capable of adapting to configurations different from those for which it was trained.

4 | CONCLUSIONS

In this work, a reduced-order model based on adaptive finite element meshes and a correction term obtained by using an artificial neural network (FAN-ROM) has been presented. The idea is to run a high-fidelity simulation by using an adaptively refined finite element mesh and compare the results obtained with those of a coarse mesh finite element model. From this comparison, a correction forcing term can be computed for each training configuration. A model for the correction term is built by using an ANN, and the final ROM is obtained by putting together the coarse mesh finite element model, plus the ANN model for the correction forcing term.

The methodology has been applied firstly to a nonlinear solid mechanics problem, where we have departed from the training in a static configuration and then compared the results in a nontrained dynamic setting. The results in this case

have been accurate, and an enhancement of the behavior of the ROM has been obtained, while maintaining a significant speedup with respect to the high-fidelity model.

The second problem to which the methodology has been applied is a transient quasi-incompressible flow. Due to the saddle point nature of the pure incompressible flow problem and its associated stability issues, a small penalty term has been added to the incompressibility constraint. This term has been necessary for the stability of the ROM. A simple flow past a cylinder problem has been tested, and good results have been obtained for Re = 100, but unstable behavior was observed when trying to apply the reduced-order approach to a more complex Re = 10,000 setting.

Finally, the two previous ROMs have been put together in a fluid-structure interaction problem at Re = 100, without any further training. Good results have been obtained in this problem, showing the applicability of the ANN trained ROM in configurations that are different from the training regime.

The results of the numerical examples show that the FAN-ROM is capable of improving the simulation results obtained in coarse finite element meshes at a reduced computational cost. The model is capable of adapting to configurations that differ from the training set, and as a consequence, it can be useful in design processes in engineering, where iterative optimization is usually employed. However, the proposed FAN-ROM model has failed to reproduce fluid flow problems when the Reynolds number is high. The stability of the model in such settings and other complex problems will be a matter of future work.

ORCID

Joan Baiges https://orcid.org/0000-0002-3940-5887

REFERENCES

1. Lucia DJ, Beran PS, Silva WA. Reduced-order modeling: new approaches for computational physics. *Prog Aerosp Sci*. 2004;40(1-2):51-117.
2. Burkardt J, Gunzburger M, Lee H. POD and CVT-based reduced-order modeling of Navier-Stokes flows. *Comput Methods Appl Mech Eng*. 2006;196(1-3):337-355.
3. Akhtar I, Borggaard J, Hay A. Shape sensitivity analysis in flow models using a finite-difference approach. *Math Probl Eng*. 2010;2010:1-23.
4. Bergmann M, Cordier L, Brancher J. Drag minimization of the cylinder wake by trust-region proper orthogonal decomposition. In: *Active Flow Control*. Berlin, Germany: Springer-Verlag Berlin Heidelberg; 2007;95:309-324.
5. Rozza G, Lassila T, Manzoni A. Reduced basis approximation for shape optimization in thermal flows with a parametrized polynomial geometric map. In: Hesthaven JS, Rønquist EM, eds. *Spectral and High Order Methods for Partial Differential Equations*. Berlin, Germany: Springer-Verlag Berlin Heidelberg; 2011:307-315.
6. Chinesta F, Ammar A, Cueto E. Recent advances and new challenges in the use of the proper generalized decomposition for solving multidimensional models. *Arch Comput Methods Eng*. 2010;17(4):327-350.
7. Chinesta F, Ladeveze P, Cueto E. A short review on model order reduction based on proper generalized decomposition. *Arch Comput Methods Eng*. 2011;18(4):395-404.
8. Chen P, Schwab C. Adaptive sparse grid model order reduction for fast Bayesian estimation and inversion. In: Garcke J, Pflüger D, eds. *Sparse Grids and Applications, Stuttgart 2014*. Cham, Switzerland: Springer International Publishing; 2016:1-27.
9. Bungartz H-J, Griebel M. Sparse grids. *Acta Numerica*. 2004;13:147-269.
10. Baiges J, Codina R, Idelsohn S. A domain decomposition strategy for reduced order models. Application to the incompressible Navier-Stokes equations. *Comput Methods Appl Mech Eng*. 2013;267:23-42.
11. Baiges J, Codina R, Idelsohn S. Explicit reduced-order models for the stabilized finite element approximation of the incompressible Navier-Stokes equations. *Int J Numer Methods Fluids*. 2013;72(12):1219-1243.
12. Kerschen G, Golinval J-C, Vakakis AF, Bergman LA. The method of proper orthogonal decomposition for dynamical characterization and order reduction of mechanical systems: an overview. *Nonlinear Dynamics*. 2005;41(1-3):147-169.

13. Feng L. Review of model order reduction methods for numerical simulation of nonlinear circuits. *Appl Math Comput*. 2005;167(1): 576-591.

14. Reyes R, Codina R, Baiges J, Idelsohn S. Reduced order models for thermally coupled low Mach flows. *Adv Model Simul Eng Sci*. 2018;5:1-20.

15. Tello A, Codina R, Baiges J. Fluid structure interaction by means of variational multiscale reduced order models. *Int J Numer Methods Eng*. 2019.

16. Du Q, Faber V, Gunzburger M. Centroidal Voronoi tessellations: applications and algorithms. *SIAM Review*. 1999;41(4):637-676.

17. Fernández-Godino MG, Park C, Kim N-H, Haftka RT. Review of multi-fidelity models. 2016. arXiv preprint arXiv: 1609.07196.

18. Forrester AI, Sóbester A, Keane AJ. Multi-fidelity optimization via surrogate modelling. *Proc Royal Soc A Math Phys Eng Sci*. 2007;463(2088):3251-3269.

19. Ng LW-T, Eldred M. Multifidelity uncertainty quantification using non-intrusive polynomial chaos and stochastic collocation. Paper presented at: 53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference 20th AIAA/ASME/AHS Adaptive Structures Conference 14th AIAA; 2012; Honolulu, HI.

20. Viana FA, Steffen V, Butkewitsch S, de Freitas Leal M. Optimization of aircraft structural components by using nature-inspired algorithms and multi-fidelity approximations. *J Glob Optim*. 2009;45(3):427-449.

21. Goh J, Bingham D, Holloway JP, Grosskopf MJ, Kuranz CC, Rutter E. Prediction and computer model calibration using outputs from multifidelity simulators. *Technometrics*. 2013;55(4):501-512.

22. Huang D, Allen TT, Notz WI, Miller RA. Sequential kriging optimization using multiple-fidelity evaluations. *Struct Multidiscip Optim*. 2006;32(5):369-382.

23. Le Gratiet L. Bayesian analysis of hierarchical multifidelity codes. *SIAM/ASA J Uncertain Quantification*. 2013;1(1):244-269.

24. Chen R, Xu J, Zhang S, Chen C-H, Lee LH. An effective learning procedure for multi-fidelity simulation optimization with ordinal transformation. Paper presented at: IEEE International Conference on Automation Science and Engineering (CASE); 2015; Gothenburg, Sweden.

25. Jessen TL. *Effect of Microstructural Variability on the Mechanical Behavior of Ceramic Fiber Reinforced Ceramic Matrix Composites* [PhD thesis]. New Brunswick, NJ: Rutgers University; 1996.

26. Sun G, Li G, Stone M, Li Q. A two-stage multi-fidelity optimization procedure for honeycomb-type cellular materials. *Comput Mater Sci*. 2010;49(3):500-511.

27. Kennedy MC, O'Hagan A. Bayesian calibration of computer models. *J Royal Stat Soc Ser B Stat Methodol*. 2001;63(3):425-464.

28. Kleijnen JP. Kriging metamodeling in simulation: a review. *Eur J Oper Res*. 2009;192(3):707-716.

29. March A, Willcox K, Wang Q. Gradient-based multifidelity optimisation for aircraft design using Bayesian model calibration. *Aeronaut J*. 2011;115(1174):729-738.

30. Baiges J, Codina R, Idelsohn S. Reduced-order subscales for POD models. *Comput Methods Appl Mech Eng*. 2015;291:173-196.

31. Ryckelynck D. A priori hyperreduction method: an adaptive approach. *J Comput Phys*. 2005;202(1):346-366.

32. Ryckelynck D. Hyper-reduction of mechanical models involving internal variables. *Int J Numer Methods Eng*. 2009;77(1):75-89.

33. Rahman SM, Ahmed SE, San O. A dynamic closure modeling framework for model order reduction of geophysical flows. *Phys Fluids*. 2019;31(4).

34. Xie X, Mohebujjaman M, Rebholz L, Iliescu T. Data-driven filtered reduced order modeling of fluid flows. *SIAM J Sci Comput*. 2018;40(3):B834-B857.

35. Mohebujjaman M, Rebholz L, Iliescu T. Physically constrained data-driven correction for reduced-order modeling of fluid flows. *Int J Numer Methods Fluids*. 2019;89(3):103-122.

36. San O, Maulik R. Neural network closures for nonlinear model order reduction. *Adv Comput Math*. 2018;44(6):1717-1750.

37. San O, Maulik R. Machine learning closures for model order reduction of thermal fluids. *Appl Math Model*. 2018;60:681-710.

38. Wang Q, Hesthaven JS, Ray D. Non-intrusive reduced order modeling of unsteady flows using artificial neural networks with application to a combustion problem. *J Comput Phys*. 2019;384:289-307.

39. Kou J, Zhang W. Multi-kernel neural networks for nonlinear unsteady aerodynamic reduced-order modeling. *Aerosp Sci Technol*. 2017;67:309-326.

40. Huang G-B, Zhu Q-Y, Siew C-K. Extreme learning machine: theory and applications. *Neurocomputing*. 2006;70(1-3):489-501.

41. San O, Maulik R. Extreme learning machine for reduced order modeling of turbulent geophysical flows. *Phys Rev E*. 2018;97(4):042322.

42. Badia S, Baiges J. Adaptive finite element simulation of incompressible flows by hybrid continuous-discontinuous Galerkin formulations. *SIAM J Sci Comput*. 2013;35(1):A491-A516.

43. Baiges J, Bayona C. Refficientlib: an efficient load-rebalanced adaptive mesh refinement algorithm for high-performance computational physics meshes. *SIAM J Sci Comput*. 2017;39(2):65-95.

44. Baiges J, Codina R, Pont A, Castillo E. An adaptive fixed-mesh ALE method for free surface flows. *Comput Methods Appl Mech Eng*. 2017;313:159-188.

45. Nissen S. *Implementation of a Fast Artificial Neural Network Library (FANN)*. Report. Copenhagen, Denmark: Department of Computer Science, University of Copenhagen (DIKU); 2003.

46. Codina R, Badia S, Baiges J, Principe J. Variational multiscale methods in computational fluid dynamics. In: *Encyclopedia of Computational Mechanics*. 2nd ed. Hoboken, NJ: Wiley; 2018:1-28.

47. Codina R. A stabilized finite element method for generalized stationary incompressible flows. *Comput Methods Appl Mech Eng*. 2001;190(20-21):2681-2706.

48. Codina R. Stabilized finite element approximation of transient incompressible flows using orthogonal subscales. *Comput Methods Appl Mech Eng*. 2002;191(39-40):4295-4321.

49. Donea J, Huerta A, Ponthot J, Rodriguez-Ferran A. Arbitrary Lagrangian-Eulerian methods. In: *Encyclopedia of Computational Mechanics*. Vol 1. Hoboken, NJ: Wiley; 2004.