Towards non-intrusive reduced order 3D free surface flow modelling

D. Xiao[a,*], F. Fang[a,*], C.C. Pain[a], I.M. Navon[b]

[a]*Applied Modelling and Computation Group,*
Department of Earth Science and Engineering,Imperial College London,
Prince Consort Road, London, SW7 2BP, UK.URL: http://amcg.ese.imperial.ac.uk
[b]*Department of Scientific Computing, Florida State University,Tallahassee, FL, 32306-4120, USA*

1. Abstract

In this article, we describe a novel non-intrusive reduction model for three-dimensional (3D) free surface flows. However, in this work we limit the vertical resolution to be a single element. So, although it does resolve some non-hydrostatic effects, it does not examine the application of reduced modelling to full 3D free surface flows, but it is an important step towards 3D modelling. A newly developed non-intrusive reduced order model (NIROM) [1] has been used in this work. Rather than taking the standard POD approach using the Galerkin projection, a Smolyak sparse grid interpolation method is employed to generate the NIROM. A set of interpolation functions is constructed to calculate the POD coefficients, where the POD coefficients at previous time steps are the inputs of the interpolation function. Therefore, this model is non-intrusive and does not require modifications to the code of the full system and is easy to implement.

By using this new NIROM, we have developed a robust and efficient reduced order model for free surface flows within a 3D unstructured mesh finite element ocean model. What distinguishes the reduced order model developed here from other existing reduced order ocean models is (1) the inclusion of 3D dynamics with a free surface (the 3D computational domain and meshes are changed with the movement of the free surface); (2) the incorporation of wetting-drying; and (3) the first implementation of non-intrusive reduced order method in ocean modelling. Most importantly, the change of the computational domain with the free surface movement is taken into account in reduced order modelling. The accuracy and predictive capability of the new non-intrusive free surface flow ROM have been evaluated in Balzano and Okushiri tsunami test cases. This is the first step towards 3D reduced order modelling in realistic ocean cases. Results obtained show that the accuracy of free surface problems relative to the high fidelity model is maintained in ROM whilst the CPU time is reduced by several orders of magnitude.

*Corresponding author
Email addresses: dh.xiao@imperial.ac.uk (D. Xiao), f.fang@imperial.ac.uk (F. Fang)

Key words: Non-intrusive Model Reduction, free surface flows, Proper Orthogonal Decomposition, Smolyak sparse grid

2. Introduction

The numerical simulation of ocean modelling is important to a wide range of applications such as atmosphere, sea ice, climate prediction, biospheric management and especially natural disasters (for example, flood and tsunami). The natural disasters often cause big losses and tragic consequences. In order to reduce the losses, a real-time, early-warning and rapid assessment model is required. In comparison to 2D modelling, 3D ocean modelling provides better understanding and much more information about local flow structures, vertical inertia, water level changes, unsteady dynamic loads on structure interacting with fluids, flow structures close to islands and dikes etc. However, the majority of existing 3D ocean models suffer from an intensive computational cost and cannot respond rapidly for tsunami forecasting. In this case, model reduction technology has been presented to mitigate the expensive CPU computational cost since the model reduction technology offers the potential to simulate complex systems with substantially increased computation efficiency.

Among existing model reduction techniques, the proper orthogonal decomposition (POD) method has proven to be an efficient means of deriving the reduced basis functions for high-dimensional nonlinear flow systems. The POD method and its variants have been successfully applied to a number of research fields, for example, signal analysis and pattern recognition [2], statistics [3], geophysical fluid dynamics and meteorology [4], ocean modelling [5, 6, 1, 7], large-scale dynamical systems [8], ecosystem modelling [9], data assimilation of wave modelling [10, 11], ground-water flow [12], air pollution modelling [13], shape optimisation [14], aerospace design [15, 16], lithium-ion batteries convective Boussinesq flows [17], mesh optimisation model [18] and also shallow water equations. This includes the work of Stefanescu *et al.* [19, 20], Daescu and Navon [21, 22], Chen *et al.* [23, 24], Du *et al.* [25] as well as Fang *et al.* [26, 27].

However, the standard reduced order modelling is usually generated through POD and Galerkin projection method, which means it suffers from instability and nonlinearity efficiency problems. Various methods for improving numerical instability have been developed such as regularisation method [28], Petrov−Galerkin [5, 26], method of introducing a diffusion term [29, 30] and Fourier expansion [31]. For nonlinear efficiency problems, a number of methods have been proposed including empirical interpolation method (EIM) [32] and discrete empirical interpolation method (DEIM) [33], residual DEIM (RDEIM) [34], Gauss−Newton with approximated tensors (GNAT) method [35], least squares Petrov−Galerkin projection method [29], and quadratic expansion method [36, 27].

However, those methods are still dependent on the full model source codes. In many contexts, the source codes governed by partial differential equations need to be modified and maintained. Developing and maintaining these modifications are cumbersome [37]. To circumvent these shortcomings, non-intrusive approaches have been introduced into ROMs. Chen presented a black-box stencil interpolation non-intrusive method (BSIM) based on machine learning methods [37]. D. Wirtz *et al.* proposed the

kernel methods where the learning methods are based on both support vector machines and a vectorial kernel greedy algorithm [38, 39]. Audouze *et al.* proposed a non-intrusive reduced order modeling method for nonlinear parametrized time-dependent PDEs using the radial basis function approach and POD [40, 41]. Klie used a three-layer radial basis function neural network combined with POD/DEIM to predict the production of petroleum reservoirs [42]. Walton *et al.* developed a NIROM for unsteady fluid flows using the radial basis function (RBF) interpolation and POD [43]. Noori [44] and Noack [45] applied a neural network to construct the ROM. Xiao *et al.* presented a non-intrusive reduced order modelling method for Navier-Stokes equations based on POD and the RBF interpolation [7] and applied it successfully into fluid-structure interaction problems [46, 47]. The CPU computational times are reduced by several orders of magnitude by using this POD-RBF method. Xiao *et al.* also introduced the Smolyak sparse grid interpolation method into model reduction to construct the NIROM [1].

POD ROM approaches have been applied to ocean problems [48, 27, 49, 50]. Dao *et al* introduced ROM into tsunami forecasting [49], and Zokagoa and Soulaimani [50] used POD/ROM for Monte-Carlo-type applications. In their work, the POD-based reduced-order models were constructed for the shallow water equations. In shallow water modelling, however there are some errors in results when involving ocean problems like radical topography changes, short waves and local flows around the buildings or mountains. The work of Fang *et al* [48, 27], Du *et al* [36], and Xiao *et al* [5] introduced POD ROM for 2D/3D Navier-Stokes unstructured mesh finite element fluid modelling. However 3D free surface flow examples were not included in their work due to the difficulty in implementation of intrusive POD-ROMs. The implementation difficulty was caused by the change of both the computational domain and 3D unstructured meshes with free surface movement. However, NIROM is capable of handling this issue easily.

This paper, for the first time, constructs a NIROM for free surface flows in the framework of an unstructured mesh finite element ocean model. This is achieved by using the Smolyak sparse grid interpolation method. The Smolyak sparse grid method is a widely used interpolation method and is used to overcome the curse of dimensionality. It was also used for uncertainty quantification for electromagnetic devices [51] where the Smolyak sparse grid was used to calculate statistically varying material and geometric parameters which were the inputs of the ROM. Xiao *et al.* first used Smolyak sparse grids to construct ROM [1] and it has been shown to be a promising non-intrusive method for representing complex physical system using a set of hypersurface interpolating functions. The NIROM can be treated as a black box, which uses a set of hypersurfaces constructed based on the Smolyak sparse grid collocation method to replace the traditional reduced order model. The errors in the NIROMs come from: the POD function truncation error (the ability of the basis functions to represent the solution), the error associated with having just a certain number of solution snapshots (rather than the solution at all time steps) and the error from the calculation of the NIROM solution (for more details, please see [52]) using, for example, sparse grids or Radial Basis Functions.

In this work, the newly presented NIROM method based on Smolyak sparse grids is applied to complex ocean free surface flows. The capability of newly developed

NIROM for 3D free surface flows are numerically tested and illustrated in Balzano and Okushiri tsunami test cases. The main novelty of this work is the inclusion of 3D flow dynamics with a free surface and the wetting-drying front. The solutions from the full fidelity ocean model are recorded as a sequence of snapshots, and from these snapshots appropriate basis functions are generated that optimally represent the flow dynamics. The Smolyak sparse grid interpolation method is then used to form a hyper-surface that represents the ROM. Once the hyper-surface has been constructed, the POD coefficient at current time step can be obtained by providing the POD coefficients at previous time steps to this hyper-surface. Numerical comparisons between the high fidelity model and this NIROM are made to investigate the accuracy of this novel NIROM for free surface flows.

The structure of the paper is as follows. Section 3 presents the governing equations of free surface flows. Section 4 presents the derivation of the POD model reduction and re-formulation of the governing equations using the Smolyak sparse grid method. Section 5 illustrates the methodology derived above via two numerical examples. This is based on two test problems where the Balzano test case and Okushiri tsunami test case are numerically simulated. Finally in section 6 conclusions are presented and the novelty of the manuscript is fully summarized and illuminated.

3. Three dimensional governing equations for free surface flows

3.1. 3D Navier-Stokes equations

The three dimensional incompressible Navier-Stokes equations with Boussinesq approximation and the conservative equation of mass are used in this work:

$$\nabla \cdot \vec{u} = 0, \tag{1}$$

$$\frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \nabla \vec{u} = -\nabla p + \nabla \cdot \tau. \tag{2}$$

where the terms $\vec{u} \equiv (u_x, u_y, u_z)^T$ are the velocity vector, p the perturbation pressure ($p := p/\rho_0$, ρ_0 is the constant reference density). The stress tensor τ represents the viscous forces:

$$\tau_{ij} = 2\mu_{ij} S_{ij}, \quad S_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) - \frac{1}{3}\sum_{k=1}^{3}\frac{\partial u_k}{\partial x_k}, \quad i, j = \{x, y, z\}, \tag{3}$$

where μ denotes the kinematic viscosity. The no-normal flow boundary condition is applied on the bottom and sides of the computational domain:

$$\vec{u} \cdot \vec{n} = 0, \tag{4}$$

where $\vec{n}$ denotes the unit normal vector on boundary surface.

3.2. Combining kinematic free surface condition

The kinematic free surface boundary condition is expressed as follows:

$$\frac{\partial \eta}{\partial t} = -\left.\vec{u}_H\right|_{z=\eta} \cdot \nabla_H \eta + \left. u_z\right|_{z=\eta} \quad \text{on} \quad \partial\Omega_s, \tag{5}$$

where η is the free surface elevation, $\partial\Omega_s \subset \partial\Omega$ is the free surface boundary, $\nabla_H \equiv (\partial/\partial x, \partial/\partial y)^T$, and $\vec{u}_H$ is the horizontal component of $\vec{u}$. Using the fact that the normal vector $\vec{n}$ at the free surface is $\frac{(-\frac{\partial\eta}{\partial x}, -\frac{\partial\eta}{\partial y}, 1)^T}{\|(-\frac{\partial\eta}{\partial x}, -\frac{\partial\eta}{\partial y}, 1)^T\|}$, equation (5) can be reformulated to

$$\frac{\partial \eta}{\partial t} = \frac{\vec{u} \cdot \vec{n}}{\vec{n} \cdot \vec{k}}, \tag{6}$$

where $\vec{k} = (0, 0, 1)$ is the vertical standard basis vector. Note that in spherical geometries $\vec{k}$ is replaced with $\vec{r} = (\sin\theta\cos\phi, \sin\theta\sin\phi, \cos\theta)$ where ϕ and θ are the azimuthal and co-latitudinal angles respectively.

Taking into account $p = \rho_0 g\eta$ on the free surface $\partial\Omega_s$, gives the combining kinematic free surface boundary condition:

$$\vec{n} \cdot \vec{k} \frac{1}{\rho_0 g} \frac{\partial p}{\partial t} = \vec{n} \cdot \vec{u}. \tag{7}$$

In a wetting and drying scheme, a threshold value d_0 is introduced to define the wet and dry areas/interface. In order to prevent a non-physical flow, a thin layer is kept equal to the threshold value d_0 in dry areas. In wetting and drying, different boundary conditions are applied on the free surface. A no normal flow boundary condition is applied on dry areas while a kinematic free surface boundary condition is used on wet areas.

4. POD/Smolyak non-intrusive reduced order formulation

In this section, the method of constructing the NIROM for free surface flow problems is described. The essence of this method lies in how to construct a set of interpolation functions or hyper surfaces that represent the reduced free surface problem system using the Smolyak sparse grid method. Firstly, the solutions from the high fidelity ocean model are recorded as a number of snapshots where the details of 3D free surface dynamics (wetting-drying front, free surface heights, waves etc) are included. Secondly, from these snapshots a number of basis functions, that optimally represent the free surface flow dynamics, are then generated. Thirdly, the Smolyak sparse grid interpolation method is used to form a set of hyper-surfaces that represent the reduced system. Once the hyper-surfaces have been constructed, the solution of the NIROM, at the current time level, can be obtained from reduced solution at the previous time level using the hyper-surface functions.

4.1. The Proper Orthogonal Decomposition method

In this section, the POD theory is briefly described. The objective of the POD method presented here is to extract a set of P optimal basis functions from the snapshots recorded solutions of velocity and pressure (free surface) at a number of different time levels. In this work the snapshots are obtained by solving the discretised form of equations (2), which considers the free surface boundary condition. Four separate matrices $\mathbf{U}^x, \mathbf{U}^y, \mathbf{U}^z$ and $\mathbf{U}^p$ representing velocity from different coordinates directions

and pressure (free surface) are formed from the snapshots. Each matrix will be treated in an identical way, so for the sake of simplicity of presentation, a general matrix $\mathbf{U}$ is used for representing the four matrices. The dimension of the matrix $\mathbf{U}$ is $F \times S$, where F and S denote the number of nodes on the finite element mesh and the number of snapshots respectively. The mean of the snapshots is defined as:

$$\overline{\mathbf{U}}_i = \frac{1}{S} \sum_{j=1}^{S} \mathbf{U}_{j,i}, \quad i \in \{1, 2, \ldots, F\}. \tag{8}$$

Taking the mean from the matrix $\mathbf{U}$ yields a new matrix $\tilde{\mathbf{U}}_{j,i}$, which is used for performing Singular Value Decomposition (SVD):

$$\tilde{\mathbf{U}}_{j,i} = \mathbf{U}_{j,i} - \overline{\mathbf{U}}_i, \quad i \in \{1, 2, \ldots, F\}, \quad j \in \{1, 2, \ldots, S\}. \tag{9}$$

Computing the SVD of the matrix $\tilde{\mathbf{U}}_{j,i}$ has the form,

$$\tilde{\mathbf{U}} = U \Sigma V^T, \tag{10}$$

where matrix U has a size of $F \times F$ and it is constructed by the eigenvectors $\tilde{\mathbf{U}}\tilde{\mathbf{U}}^T$. Matrix V has a size of $S \times S$ and it is constructed by the eigenvectors $\tilde{\mathbf{U}}^T\tilde{\mathbf{U}}$. They are unitary matrices and the matrix Σ is a diagonal matrix of size $F \times S$. The non zero values of Σ are the singular values of matrix $\tilde{\mathbf{U}}$ and are listed in decreasing order. The singular values provide a criteria (truncation point) for choosing the number of optimal basis functions P. A formulation is given to calculate the energy captured from the full system:

$$E = \frac{\sum_{i=1}^{P} \lambda_i}{\sum_{i=1}^{S} \lambda_i}, \tag{11}$$

where E represents the energy of the snapshots captured by the first P POD basis functions. If the singular values decay fast, most of the 'energy' in the original dynamic system can be captured only by a small number of leading POD basis functions provided we satisfy the Kolmogorov n-width condition.

The POD basis functions can be defined as the column vectors of the matrix U [53] :

$$\Phi_j = U_{:,j}, \qquad \text{for } j \in \{1, 2 \ldots S\}. \tag{12}$$

$$\tag{13}$$

These functions are optimal in the sense that no other rank P set of basis functions can be closer to the matrix $\tilde{\mathbf{U}}$ in the Frobenius norm. That is, if the first P basis functions are used, the resulting matrix is the closest possible to the matrix $\tilde{\mathbf{U}}$ in the relevant norm. In addition, the POD basis functions are orthonormal since the matrix U is unitary. After obtaining the POD basis functions, the solution of velocity $\mathbf{u}$ and pressure (free surface) $\mathbf{p}$ can be represented by the expansion:

$$\mathbf{u} = \overline{\mathbf{u}} + \sum_{j}^{P} \alpha_{\mathbf{u},j} \Phi_{\mathbf{u},j}, \quad \mathbf{p} = \overline{\mathbf{p}} + \sum_{j}^{P} \alpha_{\mathbf{p},j} \Phi_{\mathbf{p},j}, \tag{14}$$

where α denote the expansion coefficients.

4.2. The Smolyak sparse grid interpolation algorithm

In this work, the Smolyak sparse grid interpolation method is used to construct a set of hyper-surfaces representing the reduced fluid system. In this section, the sparse grid interpolation presented by Smolyak [54] is described. The Smolyak sparse grid interpolation algorithm is an efficient method that is used to approximate a high dimensional function. The advantage of using Smolyak sparse grid is that it selects only a small number of nodes from the full tensor-product grid, thus resulting in computational efficiency. It uses a parameter, approximation level l to control number of Smolyak sparse nodes R.

For one dimensional problems, a function f can be approximated by the formulae,

$$(U^l)(f) = \sum_{i=1}^{O_l} f(\xi_i^l).(\omega_i^l(\xi)), \tag{15}$$

where O_l is the number of nodes at this dimension, superscript l is the approximation level, ω is a weighting coefficient and $f(\xi_i)$ denotes the value of the function f at location ξ_i.

For d-dimensional problem, a function f can be approximated using a full tensor product, that is, has a form of,

$$(U^{l_1} \otimes \cdots \otimes U^{l_d})(f) = \sum_{i_1=1}^{O_{l_1}} \cdots \sum_{i_d=1}^{O_{l_d}} f(\xi_{i_1}^{l_1}, ..., \xi_{i_d}^{l_d}).(\omega_{i_1}^{l_1} \otimes \cdots \otimes \omega_{i_d}^{l_d}), \tag{16}$$

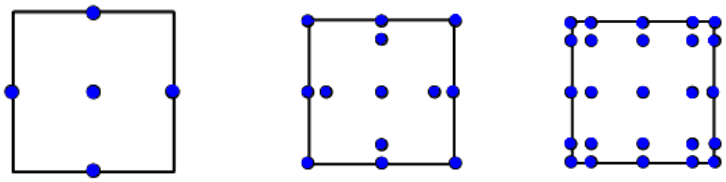
where $O_{l_1}, O_{l_2}...O_{l_d}$ are number of nodes used in dimension $(1, 2...d)$ respectively, $f(\xi_{i_1}^{l_1}, ..., \xi_{i_d}^{l_d})$ represents the function value at a point $(\xi_{i_1}^{l_1}, ..., \xi_{i_d}^{l_d})$ on the full tensor product grid. The number of O_{l_d} can be determined by the Clenshaw-Curtis quadrature rule, and $O_{l_d} = 2^{l_d-1} + 1$ [55]. However, the full tensor product interpolation suffers from the problem of 'curse of dimensionality', that is, the number of nodes $O_{l_1} \times ... \times O_{l_d}$ increases exponentially with the number of dimensions d, thus resulting in an intensive computational cost. The Smolyak sparse grid interpolation algorithm is a method to deal with the issue of 'curse of dimensionality'. The key idea of this method is that it selects the important nodes rather than all the nodes on tensor product grid. The interpolant has the following expression:

$$\hat{f}(d+l, d) = \sum_{max\{d,l+1\} \le |\mathbf{I}| \le d+l} (-1)^{d+l-|\mathbf{I}|} \cdot \begin{pmatrix} d-1 \\ d+l-|\mathbf{I}| \end{pmatrix} (U^{l_1} \otimes \cdots \otimes U^{l_d}), \tag{17}$$

where $|\mathbf{I}| = I_1 + \cdots + I_d$, I is a point index on each dimension, and for each $I_i, i \in \{1, 2, \ldots d\}$, it has a maximum value of number of nodes in this dimension i, that is, $1 \le I_i \le O_{l_i}$. The Smolyak interpolation uses the following formulation to choose nodes [56],

$$d \leqslant I_1 + I_2 + \cdots + I_d \geqslant d + l. \tag{18}$$

The number of the Smolyak sparse grid points R is determined by the approximation level l and the dimension size d (for 2D examples, see Figure 1) and $R \simeq \frac{2^d}{l!} d^l$ [57].



(a) Smolyak grid, level=1 (b) Smolyak grid, level=2 (c) 2-D tensor product

Figure 1: The figure shows the 2-D smolyak sparse grid with level 1 (left), 2-D smolyak sparse grid with level 2 (middle) and full tensor product grid (right).

The Smolyak formulation generates sparse grid points upon which the function f is evaluated on the Smolyak sparse points, thus increasing the computational efficiency in comparison with the tensor product evaluations.

4.3. Constructing a NIROM for free surface flows using Smolyak sparse grid

This section describes the method for constructing a NIROM for free surface flows using POD and Smolyak sparse grid interpolation method described in sections 4.1 and 4.2. The flow chart of constructing and solving the NIROM is graphically presented in figure 2. The process can be essentially divided into the steps below:

(1) Form a number of POD basis functions for velocity and pressure (free surface) which are used to construct the reduced order spaces;

(2) Construct the NIROM where the Smolyak sparse grid interpolation method is used to generate a set of hyper-surfaces;

(3) Solve the NIROM at each time step and project the POD coefficients onto the full space, that is, the velocity, pressure and free surface height at each time step;

(4) Update 3D unstructured elements as the free surface moves at each time step.

The key of the NIROM lies in the second step, that is, constructing a set of Smolyak interpolation functions (hyper-surfaces) ($\hat{f}_j$, $j \in \{1, 2, \dots P\}$), which has the form of

$$\alpha_j^{n+1} = \hat{f}_j(\alpha^n_{\mathbf{u},1}, \alpha^n_{\mathbf{u},2}, \dots, \alpha^n_{\mathbf{u},P}, \alpha^n_{\mathbf{p},1}, \alpha^n_{\mathbf{p},2}, \dots, \alpha^n_{\mathbf{p},P}), \quad j \in \{1, 2, \dots P\}, \tag{19}$$

where P is the number of POD bases. The input variables of the Smolyak interpolation function $\hat{f}_j$ is complete set of POD coefficients $\alpha^n = (\alpha^n_{\mathbf{u},1}, \alpha^n_{\mathbf{u},2}, \dots, \alpha^n_{\mathbf{u},P}, \alpha^n_{\mathbf{p},1}, \alpha^n_{\mathbf{p},2}, \dots, \alpha^n_{\mathbf{p},P})$ at the previous times step n. The output of the Smolyak interpolation function $\hat{f}_j$ is the j^{th} POD coefficient α^{n+1} at time step $n+1$. A detailed algorithm describing the steps of constructing the NIROM for free surface flows is outlined in algorithm 1, where, the interpolation function values need to be determined only at the Smolyak sparse grid nodes rather than on the full tensor product grid, thus resulting in an impressive computational economy. The online algorithm 2 presents the process of obtaining solutions using NIROM. After obtaining the POD coefficients, the solutions can be obtained by projecting back the POD coefficients on the full space. Then, the last step is to update the free surface values at all finite element nodes and 3D mesh locations, this is achieved by keeping the coordinates of x and y of each node in mesh unchanged and replacing the z-direction with the new free surface value at each node.

Algorithm 1: POD-Smolyak NIROM algorithm for free surface flows

(1) Generate the snapshots for velocity and pressure (free surface) over the time period $[1 - N_t]$ by running the full model;

(2) Obtain the POD bases for velocity Φ_v and pressure (free surface) Φ_p using the POD method;

(3) Generate a set of Smolyak sparse nodes $\alpha^{r,0} = (\alpha_1^{r,0}, \alpha_2^{r,0}, \ldots, \alpha_P^{r,0})$ (where $r \in \{1, 2, \ldots, R\}$, R is the number of sparse points to be chosen) at the full tensor product grid:
$[A_{min}, A_{max}] = [\alpha_{1,min}, \alpha_{1,max}] \cdots \otimes [\alpha_{j,min}, \alpha_{j,max}] \cdots \otimes [\alpha_{P,min}, \alpha_{P,max}]$, where $\alpha_{j,min}$ and $\alpha_{j,max}$ are the minimum and maximum values of the j^{th} POD coefficient;

(4) Obtain the function values $\alpha_j^{r,1} = f_j(\alpha^{r,0})$ associated with the Smolyak sparse nodes through running the full model one time step:

for $n = 1$ *to* R **do**

(i) Determine the initial condition $\psi^{r,0}$ for the full model by projecting $\alpha^{r,0}$ onto the full space, where ψ denotes any variable in the full model, for example, the velocity components u_x, u_y and u_z, and the pressure (free surface) **p**;

(ii) Determine the full solution $\psi^{r,1}$ by running the full model one time level;

(iii) Calculate the the function value $\alpha_j^{r,1}$ at sparse point r by projecting $\psi^{r,1}$ onto the reduced order space;

end for

(5) Give a set of $\alpha_j^{r,1}$, and then construct the interpolation function $\hat{f}_j$, $j \in \{1, 2, \ldots, P\}$ using (17);

(6) Initialize velocity $\alpha_u^{r,0}$ and pressure (free surface) $\alpha_p^{r,0}$, and give them to the interpolation function $\hat{f}_j$, $j \in \{1, 2, \ldots, P\}$ to obtain solutions for current time step using online algorithm (2).

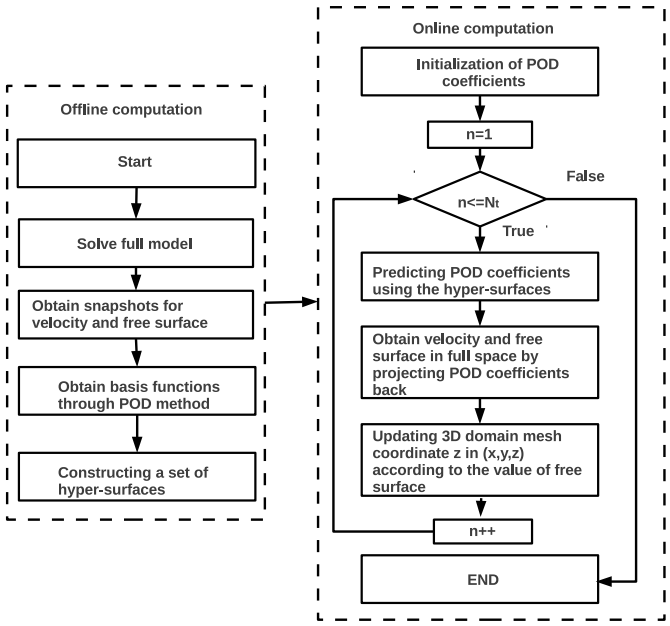


Figure 2: The flow chart of the NIROM.

Algorithm 2: Online algorithm of NIROM for free surface flows

(1) Initialize velocity ${\alpha_u}^{r,0}$ and pressure (free surface) ${\alpha_p}^{r,0}$;

(2) Calculate solutions at current time step using following loop: ;

for $n = 1$ *to* N_t **do**

for $j = 1$ *to* P **do**

Calculate the solution(POD coefficients for velocity $\alpha^n_{u,r,j}$ and pressure (free surface) $\alpha^n_{p,r,j}$) at current time step by

$$\alpha^n_j = \hat{f}_j(\alpha^{n-1}_{\mathbf{u},1}, \alpha^{n-1}_{\mathbf{u},2}, \ldots, \alpha^{n-1}_{\mathbf{u},P}, \alpha^{n-1}_{\mathbf{p},1}, \alpha^{n-1}_{\mathbf{p},2}, \ldots, \alpha^{n-1}_{\mathbf{p},P})$$

end for

(i) **Calculation of velocity components and pressure (free surface)** (u^n_x, u^n_y, u^n_z and p^n_x) by projecting α^n_j onto the full space,
$u^n_x = \overline{u}^x + \Phi^x \alpha^{x,n}$, $u^n_y = \overline{u}^y + \Phi^y \alpha^{y,n}$, $u^n_z = \overline{u}^z + \Phi^z \alpha^{z,n}$, $\mathbf{p}^n = \overline{\mathbf{p}}^{\mathbf{p}} + \Phi^{\mathbf{p}} \alpha^{\mathbf{p},n}$.

(ii) **Updating of the free surface values at all nodes and 3D mesh locations** (keeping the coordinates of x and y unchanged, replace the z-direction with the new free surface value at each node).

end for

5. Numerical Examples

The capability of the new non-intrusive reduced order model for 3D free surface flows is numerically illustrated in this section. This illustration is based on two numerical test problems: a Balzano test case and a Okushiri tsunami test case [58]. A pressure/free-surface kinematic boundary condition is enforced in the wetting zones and a no-normal flow and positive water level boundary conditions are applied to the drying zones. The free surface movement is represented by vertical mesh shifting. Evaluation of accuracy of the NIROM for 3D free surface flows was carried out through comparison of POD solutions with those obtained from the high fidelity model. The high fidelity model solutions were obtained through the use of an unstructured mesh finite element method ocean model (Fluidity, developed by the Applied Modelling and Computation Group at Imperial College London [59]).

From these full model simulations the snapshots of the solution variables were taken. Snapshots are recorded at certain time levels, for example, every five time levels or every ten time levels. The larger the number of snapshots, the higher the accuracy of the NIROM. In realistic applications, the use of too larger a number of snapshots may result in a computationally unafordable method. This has motivated the optimal selections of the time levels used as the snapshots, in for example Kunisch and Volkwei[60, 61]. The optimal time levels are chosen in such a way that the error between the high fidelity model and NIROM is minimised. Through these snapshots, the reduced order models were then formed and used to re-solve the problems.

5.1. Case 1: Balzano test cases

The first example used for validation of the new NIROM was the Balzano test case (proposed by Balzano in 1998 [62] for benchmarking different wetting and drying methods). S.W. Funke *et al*. extended the benchmarks to a 3D problem to test a wetting and drying algorithm using Fluidity [58]. In this work, a slope with a linear ascending test case was chosen to show the capability of the NIROM developed here for free surface flows. The geometry of the problem was first constructed with a 2D domain consisting of a slope with size of 13.8 km and a depth of zero meter at one end and five meters at the other end. In order to obtain a 3D domain, this 2D domain was extruded to a width of 1km (see figure 3).

A sinusoidal water level changes with a magnitude of two meters and 12 hours is applied to the five meters end (deep end of the computational domain) to trigger the flows. No normal flow boundary conditions are applied at both sides, the bottom and the shallow end of the slope. A Manning−Strickler drag with n= 0.02 $sm^{\frac{1}{3}}$ is applied at the bottom. The gravity is 9.81 ms^{-2}.

The problem was simulated for a period of 50000 seconds, and a time step size of $\Delta t = 500\ s$ was used. From the full simulation by running Fluidity, with an unstructured finite element mesh of 180 nodes and 354 elements, 100 snapshots were obtained at equally spaced time intervals for each of the u_x, u_y, u_z and p solution variables during the simulation period. A $P_1 - P_1$ finite element pair was used. The NIROM was constructed from the 100 snapshots (taking a snapshot every time step) and then used to test the problem during the simulation period.

Figure 3: Balzano case: The computational domain and mesh used in Balzano case.

Figure 4 shows the singular values in decreasing order. It can be seen that the singular eigenvalue curve decreases drastically between the first two leading POD basis functions, *i.e.* satisfying Kolmogorov condition [63]. In this case, 98% of 'energy' in the original flow dynamic system is captured with use of only three POD basis functions with 100 snapshots . In this work, two and six POD basis functions were chosen to generate the reduced order model using the Smolyak sparse grid method described above.

Figure 5 shows the solutions of pressure from the full model and NIROM using 2 and 6 POD basis functions at time instances 10.2 s and 25 s. A good agreement is achieved between the high fidelity full solutions and reduced order results. To further estimate the accuracy of NIROM, the pressure solutions at a particular location ($x = 296.8\,m, y = 686.25\,m, z = 0$) within the domain (black point in figure 3) are plotted in figure 7. Again, it can be seen that the results of NIROM with both 2 and 6 POD basis functions are in agreement with those from the full model.

To evaluate the accuracy of NIROM solutions, figure 6 shows the error of pressure solutions between the full model and NIROM with 2 and 6 POD basis functions at time instances 10.2 and 25 seconds. It is shown that the error of pressure solutions from NIROM using 6 POD basis functions is smaller than that using 2 POD basis functions. The error of pressure solutions at all nodes is further analysed by RMSE and correlation coefficient. The RMSE and correlation coefficient of pressure solutions are given in figures 8 and 9 respectively, which shows the accuracy of NIROM is improved by increasing the number of POD basis functions. The RMSE line of NIROM using 6 POD basis functions in figure 8 (a) looks like a straight line since the error is small. In order to see it clearly, it has been zoomed in, as shown in figure 8 (b). It can be seen in figure 9, the correlation coefficient line of NIROM with 6 POD basis functions is more closer to 1 than that with 2 POD basis functions. The correlation coefficient is a statistical number of the strength of a relationship between two variables. If it is close to 1, it means that the two variables are strongly correlated.

To further demonstrate the predictive capability of NIROMs, the simulation period is extended from 50000 seconds to 70000 seconds. In figure 10, the pressure solutions at a particular point ($x = 2217.9\,m, y = 475.14\,m, z = 0$), obtained from both the high fidelity model and NIROM, are given during the period $[0, 70000\,s]$. It is shown that the NIROM, built-up on the full solution during the training period $[0, 50000\,s]$ is able to provide promising results during the predictive period $[50000\,s, 70000\,s]$. More recently, we have further extended the NIROMs proposed in this work to parameterized physical problems [52]. In that work, we used another hyper-surface to represent the varying parameter space. The NIROMs are then constructed at the Smolyak sparse grid points in the parameter space. The predictive capability has been assessed by varying the boundary conditions and initial conditions, see [52].

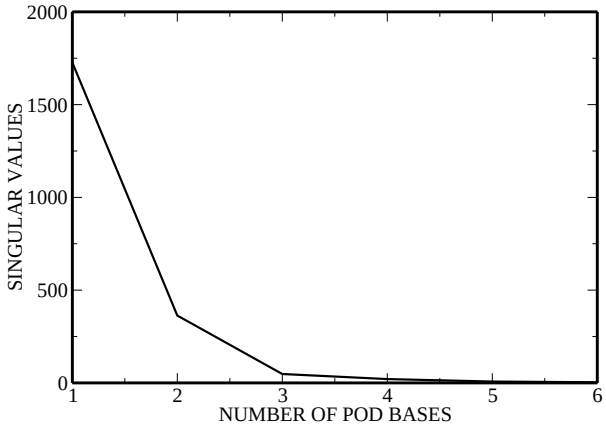
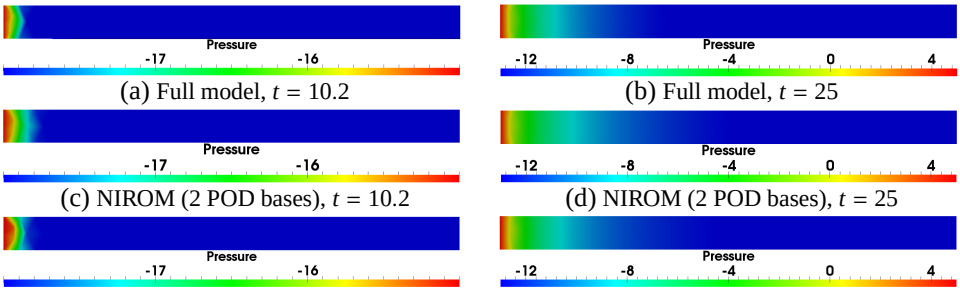


Figure 4: Balzano case: The graphs shows the singular values in order of decreasing magnitude.

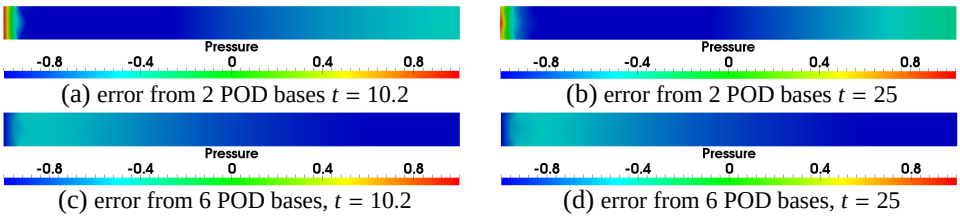


(a) Full model, $t = 10.2$ (b) Full model, $t = 25$

(c) NIROM (2 POD bases), $t = 10.2$ (d) NIROM (2 POD bases), $t = 25$

(e) NIROM (6 POD bases), $t = 10.2$ (f) NIROM (6 POD bases), $t = 25$

Figure 5: Balzano case: The solutions of pressure from the full model and NIROM at time instances 10.2 (left panel) and 25 (right panel). Top panel: the full model; middle panel: NIROM using 2 POD bases; and bottom panel: NIROM using 6 POD bases.



(a) error from 2 POD bases $t = 10.2$ (b) error from 2 POD bases $t = 25$

(c) error from 6 POD bases, $t = 10.2$ (d) error from 6 POD bases, $t = 25$

Figure 6: Balzano case: The difference of pressure solutions between the full model and NIROM, using 2 and 6 POD bases at time instances 10.2 *s* (left panel) and 25 *s* (right panel).

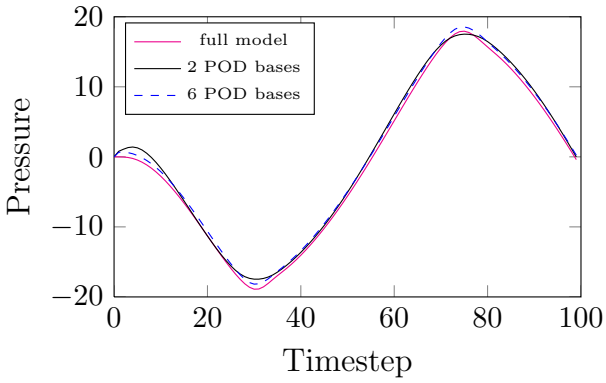
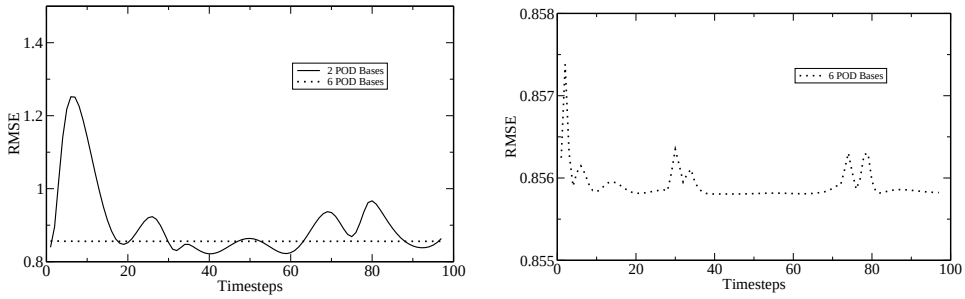


Figure 7: Balzano case: The pressure solutions from the full model and NIROM at location: (x = 296.8 m, y = 686.25 m, z = 0).



(a) RMSE, unit(m) (b) enlargement of 6 POD bases, unit(m)

Figure 8: Balzano case: The RMSE errors of pressure solutions between the full high fidelity and non-intrusive reduced order models.(b) is an enlargement of (a).

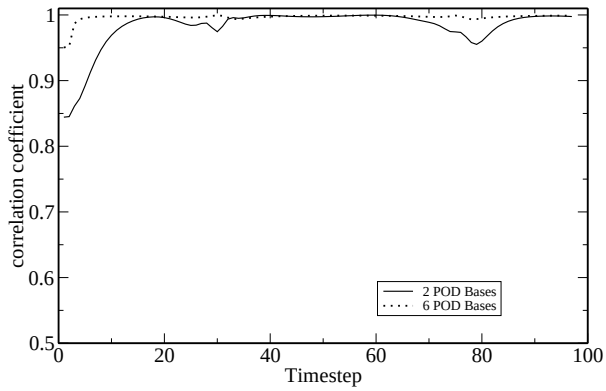


Figure 9: Balzano case: The correlation coefficient of pressure solutions between the full and non-intrusive reduced order models.

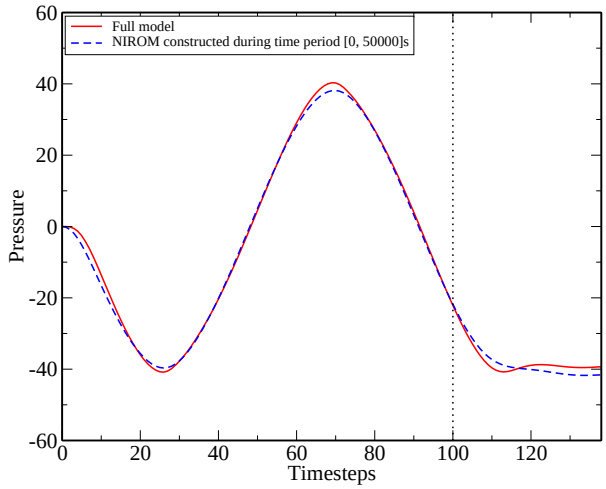


Figure 10: Balzano case: The comparison of pressure solutions between the full model and solutions predicted by NIROM model constructed during time period [0, 70000 s] at location ($x = 2217.9\,m, y = 475.14\,m, z = 0$) where the training period is [0, 50000 s] (time steps from 0 to 100) and the predictive period is [50000, 70000 s] (time steps from 100 to 140).

5.2. Case 2: Okushiri tsunami test case

The second case is a Okushiri tsunami test case. In 1993, the Okushiri tsunami struck Okushiri Island and generated huge run-up heights of almost 30 meters and currents of order of approximate 10-18 meters per second in Okushiri, Japan, which was a natural disaster. A 1/400 laboratory model of this area was constructed at Central Research Institute for Electric Power Industry in Abiko, Japan [64]. The laboratory data resembles closely the realistic bathymetry. S.W. Funke et al. used this laboratory model as a benchmark to set up a model using Fluidity [58]. The computational domain is $5.448\,m \times 3.402\,m$ in horizontal and the free surface is extruded to the bathymetry and coastal topography in vertical (see figure 12). A water height representing a tsunami wave is imposed to the left boundary and no normal flow boundary conditions are enforced to the bottom and other sides resembling the solid boundaries. The tsunami input wave boundary conditions were determined from a surface elevation profile, see figure 11. The threshold value of wetting and drying (d_0) is set to be $0.5\,mm$ in dry area to prevent non-physical flows in numerical simulation. The isotropic kinematic viscosity is set to be $0.0025\ m^2 s^{-1}$. The acceleration of gravity magnitude is $9.81\,ms^{-2}$. A $P_1 - P_1$ finite element pair is used to solve the equations. In this work, the model which is set up by Fluidity is used to evaluate the predictive capability of the NIROM.

The tsunami problem was simulated using Fluidity for a period of 26 seconds, and a time step size of $\Delta t = 0.2\,s$ was used. From the full model simulation, with a unstructured finite element mesh of 6830 nodes and 20058 elements, 100 snapshots were obtained at equal time intervals for each of the u_x, u_y and p solution variables between the simulation period. The NIROM was constructed from the 100 snapshots (taking a snapshot every time step) within an time interval $[0, 20]\,s$, a part of the full modelling run. In this test case, the main tasks were the evaluations of (1) the accuracy of NIROM during the time period $[0, 20]\,s$; and (2) the predictive capability of NIROM during the time period $[20, 26]\,s$.

Figure 14 shows the front/interface of wetting and drying. It can be seen that the shape of the computational domain is changing as the free surface keeps moving up and down. Figure 15 shows the solutions of pressure from the high fidelity model and NIROM using 18 POD basis functions at time instances $t = 10.2$ and $t = 15.2$. The difference between the high fidelity model and NIROM using 18 POD basis functions is also given in this figure. To further evaluate the performance of NIROM, the absolute error between the high fidelity model and NIROM using 6, 12 and 18 POD basis functions is given in figure 16. Again, it is shown that the error of the NIROM decreases as the number of POD basis functions used increases. Figure 17 shows the solutions of full model and the NIROM model using different number of POD basis functions at the point ($x = 0.6595\,m, y = 1.63\,m$) in the domain (point id 688 in figure 12). It can be seen that the NIROM using more POD basis functions gets closer to the solution of the full model.

The more POD basis functions are chosen, the more energy of the system will be captured. The ratio of energy captured can be quantified by equation (11). This can also be evaluated by figure 13 which shows the singular values of tsunami case in decreasing order of magnitude. The 6 POD basis functions capture 92.8% of the energy and 12 POD basis functions capture almost 98% of the energy.

In order to assess the prediction capabilities, the NIROM was built during the time period $[0, 20\,s]$ and it was run further to 26 seconds. Figure 18 shows solutions of pressure from the high fidelity model and NIROM at time instances $t = 26\,s$. The comparison of pressure solutions at two particular points ($x = 3.5696\,m, y = 1.6994\,m$, point id 760 in figure 12) and ($x = 4.9306\,m, y = 1.9685\,m$, point id 2510 in figure 12) are presented in figure 19. It can be seen that the results of NIROM are promising at the point ($x = 4.9306\,m, y = 1.9685\,m$) during the predictive time period $[20\,s, 26\,s]$ although the error is slightly larger at ($x = 3.5696\,m, y = 1.6994\,m$). Figure 20 shows the velocity and pressure solutions at the point ($x = 1.6892\,m, y = 2.1783\,m$, point id 596 in figure 12). Again, the solutions from both the high fidelity and NIROM solutions are in good agreement. The error in the predictive capability has been further analysised using the RMSE and correlation coefficient which consider all nodal values on the computational mesh. The correlation coefficient of solutions between the high fidelity full model and NIROM is computed for each time step, and is defined for given expected values χ^n_{full} and χ^n_{nirom} and standard deviations $\sigma_{\chi^n_{full}}$ and $\sigma_{\chi^n_{nirom}}$,

$$corr(\chi^n_{full}, \chi^n_{nirom})^n = \frac{cov(\chi^n_{full}, \chi^n_{nirom})}{\sigma_{\chi^n_{full}} \sigma_{\chi^n_{nirom}}} = \frac{E(\chi^n_{full} - \sigma_{\chi^n_{full}})(\chi^n_{nirom} - \sigma_{\chi^n_{nirom}})}{\sigma_{\chi^n_{full}} \sigma_{\chi^n_{nirom}}}. \tag{20}$$

where E denotes mathematical expectation, cov denotes covariance, σ denotes standard deviation. The measured error is given by the root mean square error (RMSE) which is calculated for each time step n by,

$$RMS\,E^n = \sqrt{\frac{\sum_{i=1}^{N} (\chi^n_{full,i} - \chi^n_{nirom,i})^2}{N}}. \tag{21}$$

In this expression $\chi^n_{full,i}$ and $\chi^n_{nirom,i}$ denote the full and NIROM solutions at the node i, respectively, and N represents number of nodes on the full mesh.

The figure 21 shows the RMSE and correlation coefficient values between the high fidelity full model and predicted NIROM. As shown in the figure, the error is acceptable and the correlation coefficient is above 90% during the predictive period.

Table 1 shows the online CPU cost required for simulating the high fidelity full model and NIROM for each time step. It is worth noting that the online CPU time (seconds) required for running the NIROM during one time step is only 0.004, while the full model for tsunami case and Balzano are 30.84992 and 0.7800 respectively. The simulations were performed on 12 cores workstation of an Intel(R) Xeon(R) X5680 CPU processor with 3.3GHz and 48GB RAM. The two cases were run in serial, which means only one core was used when running the test cases. The time used for the full model roughly equals to the time of assembling and solving the discretised matrices in equation (2). The CPU cost of the full model is dependent on the resolution of mesh, which means the computation time increases when finer mesh is used.

The offline cost required includes the time for forming the POD basis functions and the hypersurfaces. The time for the hypersurfaces can be ignored. The computational cost for forming the basis functions is related to the number of nodes, POD basis functions and snapshots. Table 2 lists the offline CPU cost required for forming the basis functions using different numbers of POD basis functions.

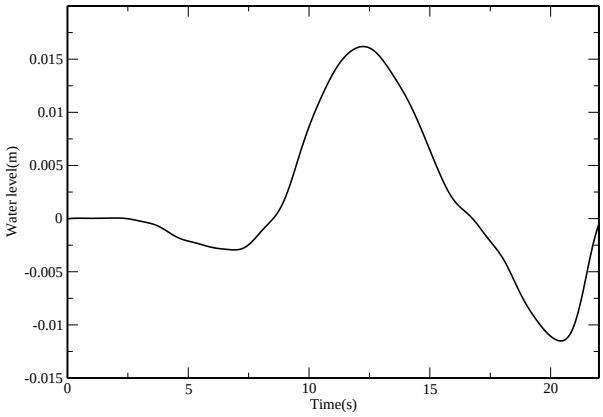


Figure 11: Okushiri tsunami case: Water level profile resembling the tsunami input wave.

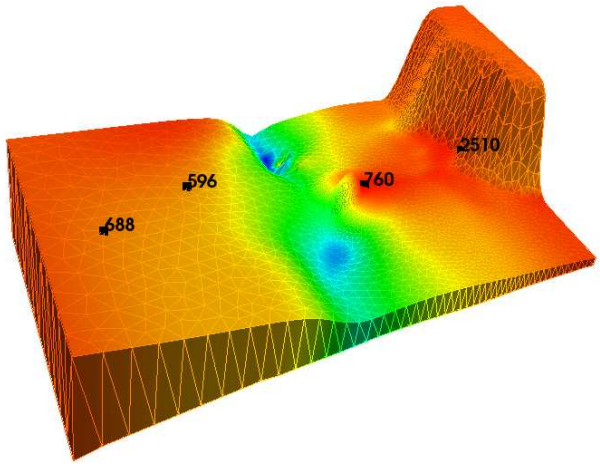


Figure 12: Okushiri tsunami case: The computational domain and unstructured meshes used.

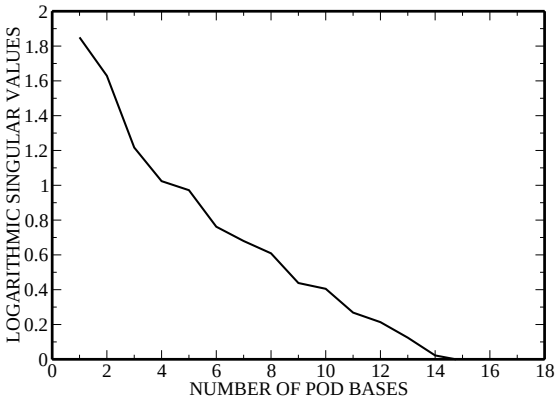
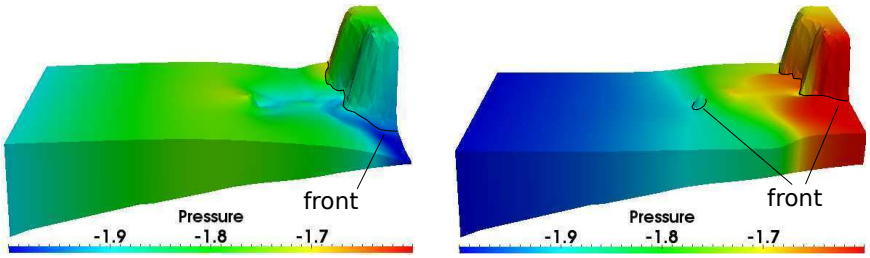


Figure 13: Okushiri tsunami case: The graphs shows the singular values in order of decreasing magnitude.

Table 1: Comparison of the online CPU time (seconds) required for running the full model and NIROM during one time step.

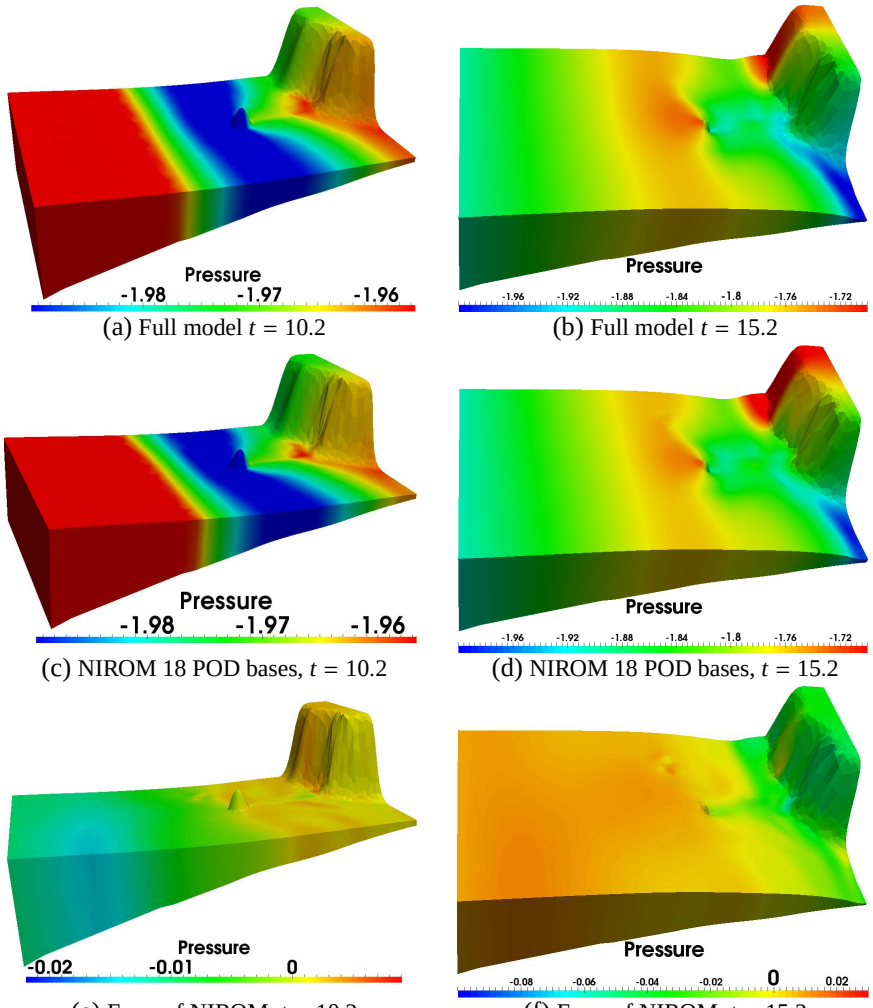
Cases	Model	assembling and solving	projection	interpolation	nonlinear iteration times	total
Okushiri tsunami case	Full model	7.71248	0	0	4	30.84992
	NIROM	0	0.003	0.001	0	0.0040
Balzano case	Full model	0.0520	0	0	15	0.7800
	NIROM	0	0.003	0.001	0	0.0040



(a) Full model, $t = 15.60s$ (b) Full model, $t = 18.75s$

(c) NIROM, $t = 15.60s$ (d) NIROM, $t = 18.75s$

Figure 14: Okushiri tsunami case: Wetting and drying front (dark line) at time instances 15.60 (left panel) and 18.75 (right panel) seconds.



(a) Full model $t = 10.2$ (b) Full model $t = 15.2$

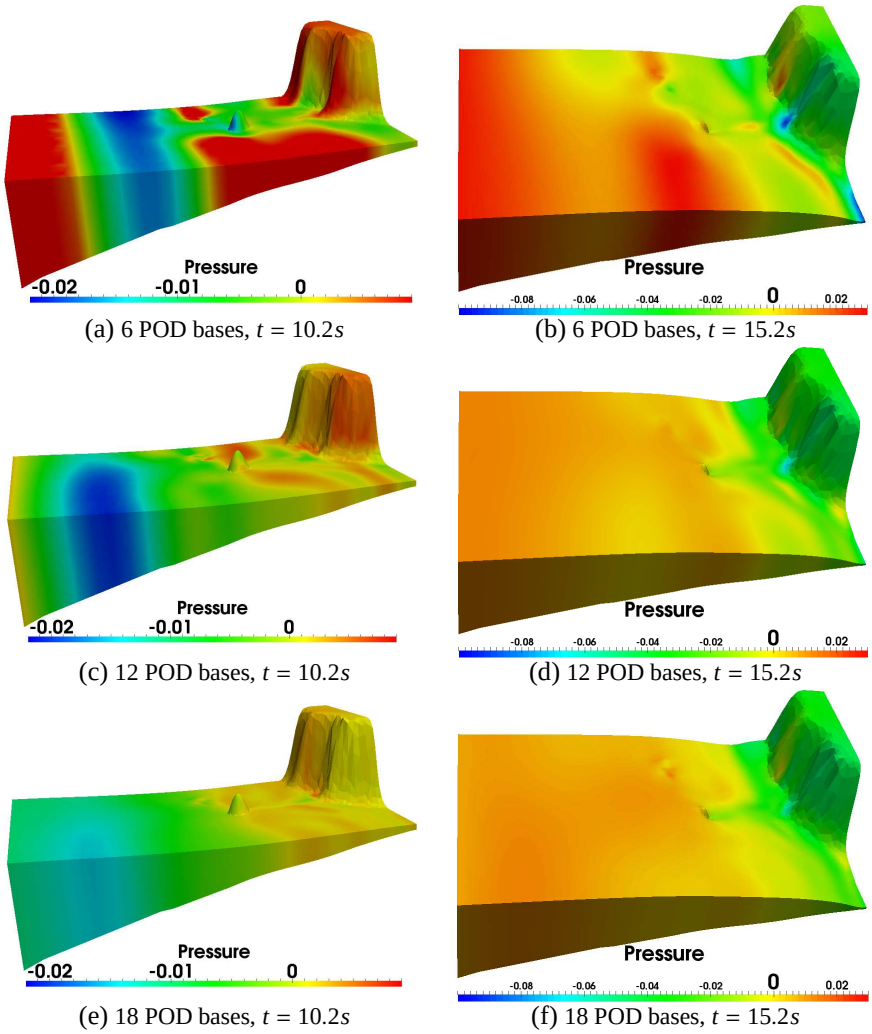
(c) NIROM 18 POD bases, $t = 10.2$ (d) NIROM 18 POD bases, $t = 15.2$

(e) Error of NIROM, $t = 10.2$ (f) Error of NIROM, $t = 15.2$

Figure 15: Okushiri tsunami case: The solutions and errors of pressure from the full model and NIROM at time instances 10.2 (left panel) and 15.2 (right panel). Top panel: the full model; middle panel: the NIROM using 18 POD basis functions; bottom panel: error between the full model and NIROM using 18 POD basis functions.

Table 2: Offline computational cost (seconds) required for constructing POD basis functions using different numbers of POD basis functions.

Number of POD bases	2	6	18	nodes	snapshots
Balzano test case	0.143	0.144	0.152	180	100
Number of POD bases	6	12	18	nodes	snapshots
tsunami test case	10.59	11.03	11.512	6830	100



(a) 6 POD bases, $t = 10.2s$ (b) 6 POD bases, $t = 15.2s$

(c) 12 POD bases, $t = 10.2s$ (d) 12 POD bases, $t = 15.2s$

(e) 18 POD bases, $t = 10.2s$ (f) 18 POD bases, $t = 15.2s$

Figure 16: Okushiri tsunami case: The difference of pressure solutions between the full model and NIROM, using 6, 12 and 18 POD basis at time instances 10.2 (left panel) and 15.2 (right panel) seconds.

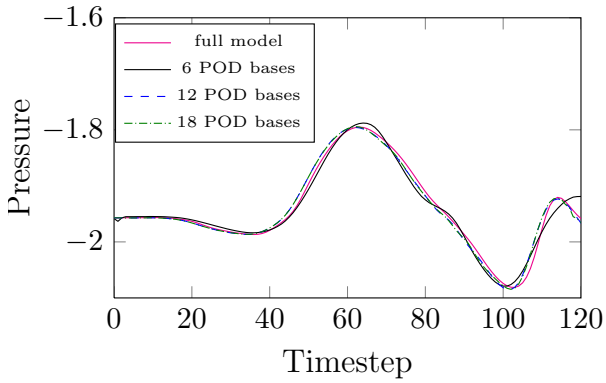
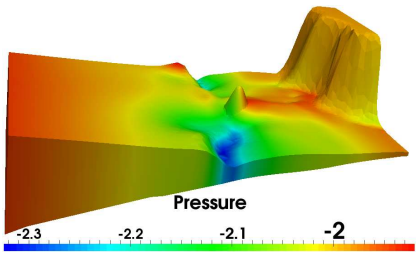
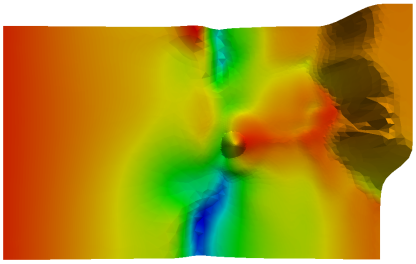


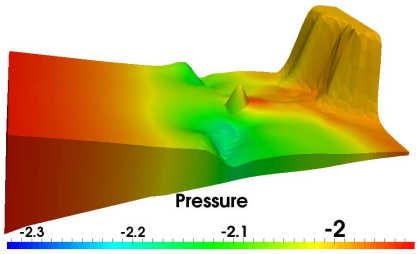
Figure 17: Okushiri tsunami case: The comparison of pressure solutions between the full model and NIROM model at location ($x = 0.6595, y = 1.63$).



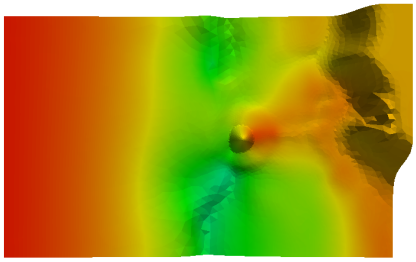
(a) Full model, $t = 26$



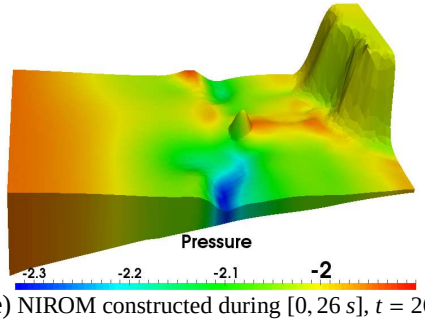
(b) Full model, profile in z direction



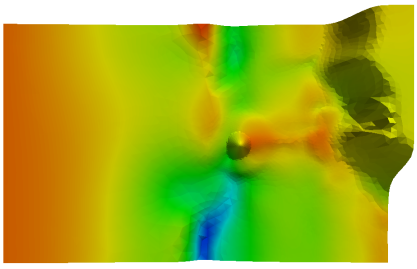
(c) NIROM constructed during [0,20]s, $t = 26$



(d) NIROM constructed during [0,20]s, profile in z direction

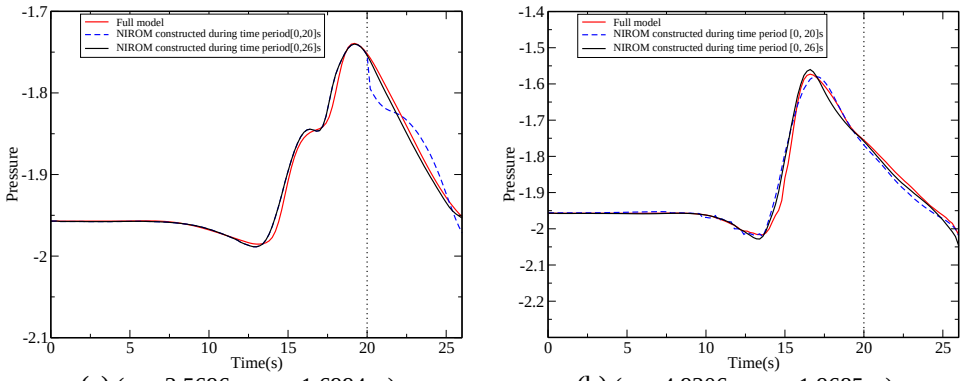


(e) NIROM constructed during $[0, 26\ s]$, $t = 26\ s$



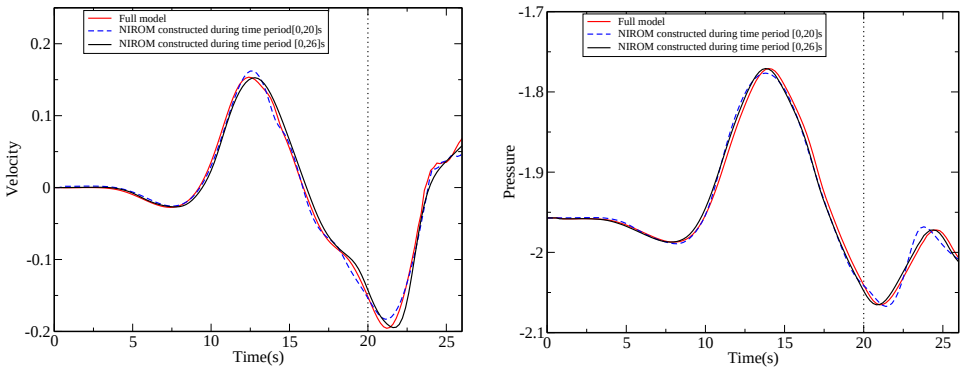
(f) NIROM constructed during $[0, 26\ s]$, profile in z direction

Figure 18: Okushiri tsunami case: The solutions of pressure from the full model (top) and NIROM constructed during time period $[0, 20\ s]$ (middle) and $[0, 26\ s]$ (bottom)at time instances 26 s.



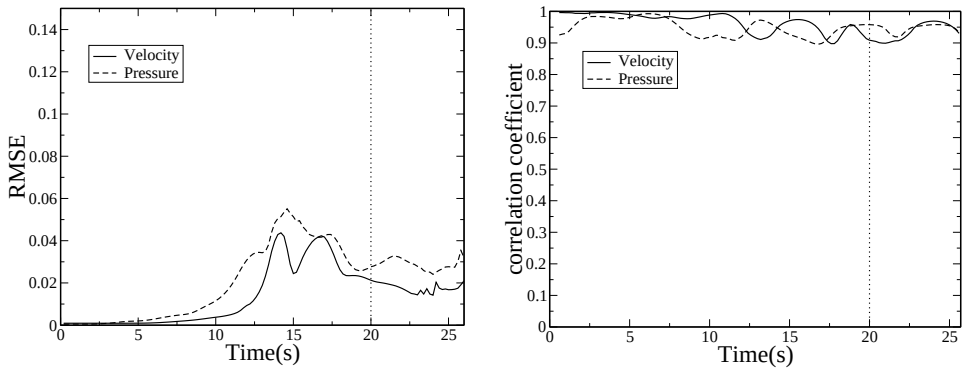
(a) ($x = 3.5696\,m, y = 1.6994\,m$) (b) ($x = 4.9306\,m, y = 1.9685\,m$)

Figure 19: Okushiri tsunami case: The comparison of pressure solutions between the full model, the NIROM constructed during time period $[0, 20]$ and $[0, 26]$ at locations ($x = 3.5696\,m, y = 1.6994\,m$) and ($x = 4.9306\,m, y = 1.9685\,m$).



(a) velocity (b) pressure

Figure 20: Okushiri tsunami case: The comparison of velocity and pressure solutions between the full model, the NIROM constructed during time period [0,20] s and [0,26] s at locations ($x = 1.6892\,m, y = 2.1783\,m$).



(a) RMSE (b) Coorelation coefficient

Figure 21: Tsunami case: The RMSE errors of pressure solutions between the full high fidelity and non-intrusive reduced order models. (b) Correlation coefficient.

6. Conclusions

In this work a non-intrusive reduced order model, based on the Smolyak sparse grid method has been, for the first time, developed for 3D free surface flows and implemented under the framework of advanced 3D unstructured mesh finite element ocean model (Fluidity). The Smolyak sparse grid method is used to construct a set of interpolation functions representing the reduced system. The free surface flow NIROM is generated from the POD bases derived from the snapshots. These snapshots are the full solutions recorded at selected time levels where the details of ocean flow dynamics (velocity, pressure, waves, eddies, wetting-drying front etc.) are included. The performance of the new POD-Smolyak 3D free surface flow NIROM is illustrated using two numerical test cases: Balzano test case and Okushiri tsunami case. To estimate the accuracy of the NIROM, the results obtained from the free surface flow NIROM have been compared against those from the high fidelity ocean model. It is shown that the accuracy of solutions from free surface flow NIROM is maintained while the CPU cost is reduced by several orders of magnitude. An error analysis has also been carried out for the validation of the free surface flow NIROM through comparing the results with results of high fidelity full model. The NIROM shows a good agreement with the high fidelity full ocean model. It was also shown that the accuracy can be improved by increasing the number of POD bases.

Importantly, the predictive ability of NIROM was tested, for test case 2, by predicting, with good accuracy, the dynamics of the final part of the time domain that the NIROM had not seen before. This is a small step towards showing that NIROM can have 'predictive skill'. Thus, the free surface NIROM may have a role to play in applications to uncertainty analysis, optimisation and data assimilation where massive numbers (*e.g.* hundreds or thousands) of runs of the ocean model are required. This will be our focus in future work. More recently, parametric ROMs for various parameter inputs (e.g. boundary conditions) have been developed. A hyper-surface can also be constructed for various parameter inputs using Smolyak sparse grids (for details, see [52]). This work will be combined, in our future work, with the NIROM developed here for 3D free surface flows.

Since NIROM works just from the snapshots of the forward solution it is ideally placed to construct rapid surrogate models from complex modelling codes (e.g. multi-physics codes) and commercial software where the source codes are unavailable or difficult to modify. However, unlike many intrusive ROMs NIROMs may have difficulty in achieving conservation as there is no underlying conservation equation - just an approximation to it. In the longer term these conservation issues need to be addressed. Future work will investigate the effects of applying this new NIROM to more complex free surface flows (for example, urban flooding), varying parametric non-intrusive cases and applications to uncertainty analysis, optimisation control and data assimilation.

Acknowledgments

This work was carried out under funding from Janet Watson scholarship at Department of Earth Science and Engineering. Authors would like to acknowledge the support

of the UK's Natural Environment Research Council projects(NER/A/S/2003/00595, NE/C52101X/1 and NE/C51829X/1), the Engineering and Physical Sciences Research Council (GR/R60898, EP/I00405X/1 and EP/J002011/1), and the Imperial College High Performance Computing Service. Prof. I.M. Navon acknowledges the support of NSF/CMG grant ATM-0931198. Xiao acknowledges the support of NSFC grant 11502241. Pain and Fang are greatful for the support provided by BP Exploration. The authors are greatful for the support of the EPSRC MEMPHIS multi-phase flow programme grant. The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/20072013) under grant agreement NO. 603663 for the research project PEARL (Preparing for Extreme And Rare events in coastaL regions). The authors acknowledge the support of EPSRC grant: Managing Air for Green Inner Cities (MAGIC)(EP/N010221/1).

References

[1] D. Xiao, F. Fang, A.G. Buchan, C.C. Pain, I.M. Navon, and A. Muggeridge. Non-intrusive reduced order modelling of the Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 293:552–541, 2015.

[2] K. Fukunaga. Introduction to statistical recognition(2nd edn). *Computer Science and Scientific Computing Series*, Academic Press,Academic Press: Boston, MA.:5–33, 1990.

[3] K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2:559–572, 1901.

[4] D.T. Crommelin. and A.J. Majda. Strategies for model reduction: Comparing different optimal bases. *Journal of the Atmospheric Sciences*, 61:2206–2217, 2004.

[5] D. Xiao, F. Fang, J. Du, C.C. Pain, I.M. Navon, A. G. Buchan, A.H. ElSheikh, and G. Hu. Non-linear Petrov-Galerkin methods for reduced order modelling of the Navier-Stokes equations using a mixed finite element pair. *Computer Methods In Applied Mechanics and Engineering*, 255:147–157, 2013.

[6] Y. Cao, J. Zhu, I.M. Navon, and Z. Luo. A reduced order approach to four dimensional variational data assimilation using proper orthogonal decomposition. *International Journal for Numerical Methods in Fluids*, 53:1571–1583, 2007.

[7] D Xiao, F Fang, C Pain, and G Hu. Non-intrusive reduced order modelling of the Navier-Stokes equations based on RBF interpolation. *International Journal for Numerical Methods in Fluids*, 79(11):580–595, 2015.

[8] Athanasios C Antoulas. *Approximation of large-scale dynamical systems*, volume 6. Siam, 2005.

[9] Joanna S Pelc, Ehouarn Simon, Laurent Bertino, Ghada El Serafy, and Arnold W Heemink. Application of model reduced 4d-var to a 1d ecosystem model. *Ocean Modelling*, 57:43–58, 2012.

[10] Kathrin Wahle, Joanna Staneva, and Heinz Guenther. Data assimilation of ocean wind waves using neural networks. a case study for the german bight. *Ocean Modelling*, 2015.

[11] MU Altaf, M Ambrozic, MF McCabe, and I Hoteit. A study of reduced-order 4dvar with a finite element shallow water model. *International Journal for Numerical Methods in Fluids*, 2015.

[12] PTM Vermeulen, AW Heemink, and CBM Te Stroet. Low-dimensional modelling of numerical groundwater flow. *Hydrological processes*, 18(8):1487–1504, 2004.

[13] F Fang, T Zhang, D Pavlidis, C.C. Pain, AG Buchan, and I.M. Navon. Reduced order modelling of an unstructured mesh air pollution model and application in 2D/3D urban street canyons. *Atmospheric Environment*, 96:96–106, 2014.

[14] Matteo Diez, Emilio F Campana, and Frederick Stern. Design-space dimensionality reduction in shape optimization by Karhunen–Loève expansion. *Computer Methods in Applied Mechanics and Engineering*, 283:1525–1544, 2015.

[15] Andrea Manzoni, Filippo Salmoiraghi, and Luca Heltai. Reduced basis isogeometric methods (RB-IGA) for the real-time simulation of potential flows about parametrized NACA airfoils. *Computer Methods in Applied Mechanics and Engineering*, 284:1147–1180, 2015.

[16] Emiliano Iuliano and Domenico Quagliarella. Proper orthogonal decomposition, surrogate modelling and evolutionary optimization in aerodynamic design. *Computers and Fluids*, 84:327–350, 2013.

[17] O San and J Borggaard. Principal interval decomposition framework for POD reduced-order modeling of convective Boussinesq flows. *International Journal for Numerical Methods in Fluids*, 78(1):37–62, 2015.

[18] F Fang, C.C. Pain, I.M. Navon, GJ Gorman, MD Piggott, PA Allison, and AJH Goddard. A POD goal-oriented error measure for mesh optimization. *International Journal for Numerical Methods in Fluids*, 63(2):185–206, 2010.

[19] Razvan Stefanescu and I.M. Navon. POD/DEIM nonlinear model order reduction of an adi implicit shallow water equations model. *Journal of Computational Physics*, 237:95–114, 2013.

[20] Razvan Stefanescu, Adrian Sandu, and I.M. Navon. Comparison of POD reduced order strategies for the nonlinear 2D shallow water equations. *International Journal for Numerical Methods in Fluids*, 76(8):497–521, 2014.

[21] D.N. Daescu and I.M. Navon. A dual-weighted approach to order reduction in 4D-Var data assimilation. *Monthly Weather Review*, 136(3):1026–1041, 2008.

[22] D.A. Bistrian and I.M. Navon. An improved algorithm for the shallow water equations model reduction: Dynamic Mode Decomposition vs POD. *International Journal for Numerical Methods in Fluids*, 78(9):552–580, 2015.

[23] X. Chen, I.M. Navon, and F. Fang. A dual-weighted trust-region adaptive POD 4D-Var applied to a finite-element shallow-water equations model. *International Journal for Numerical Methods in Fluids*, 65(5):520–541, 2011.

[24] X. Chen, S. Akella, and I.M. Navon. A dual-weighted trust-region adaptive POD 4-D Var applied to a finite-volume shallow water equations model on the sphere. *International Journal for Numerical Methods in Fluids*, 68(3):377–402, 2012.

[25] J. Du, F. Fang, C.C. Pain, I.M. Navon, J. Zhu, and D.A. Ham. POD reduced-order unstructured mesh modeling applied to 2D and 3D fluid flow. *Computers and Mathematics with Applications*, 65:362–379, 2013.

[26] F.Fang, C.Pain, I.M. Navon, A.H. Elsheikh, J. Du, and D.Xiao. Non-linear Petrov-Galerkin methods for Reduced Order Hyperbolic Equations and Discontinuous Finite Element Methods. *Journal of Computational Physics*, 234:540–559, 2013.

[27] F Fang, CC Pain, I.M. Navon, MD Piggott, GJ Gorman, PA Allison, and AJH Goddard. Reduced-order modelling of an adaptive mesh ocean model. *International journal for numerical methods in fluids*, 59(8):827–851, 2009.

[28] Alireza Jafarpour Feriedoun Sabetghadam. α Regularization of the POD-Galerkin dynamical systems of the Kuramoto-Sivashinsky equation. *Applied Mathematics and Computation*, 218:6012–6026, 2012.

[29] C. Bou-Mosleh K. Carlberg and C. Farhat. Efficient non-linear model reduction via a least-squares Petrov-Galerkin projection and compressive tensor approximations. *International Journal for Numerical Methods in Engineering*, 86:155–181, 2011.

[30] M. Serpas Y. Chu and J. Hahn. State-preserving nonlinear model reduction procedure. *Chemical Engineering Science*, 66:3907–3913, 2011.

[31] Karen Willcox and Alexandre Megretski. Model reduction for large-scale linear applications. In *Proc. of 13th IFAC Symposium on System Identification, Rotterdam, Netherlands*, pages 1431–1436, 2003.

[32] M. Barrault, Y. Maday, N.C. Nguyen, and A.T. Patera. An empirical interpolation method: application to efficient reduced-basis discretization of partial differential equations. *C. R. Acad. Sci. Paris, Ser*, 339:667–672, 2004.

[33] S. Chaturantabut and D.C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM J. Sci. Comput*, 32:2737–2764, 2010.

[34] D. Xiao, F. Fang, A. G. Buchan, C.C. Pain, I.M. Navon*, J. Du, , and G. Hu. Non-linear model reduction for the Navier-Stokes equations using Residual DEIM method. *Journal of Computational Physics*, 263:1–18, 2014.

[35] Kevin Carlberg, Charbel Farhat, Julien Cortial, and David Amsallem. The GNAT method for nonlinear model reduction: effective implementation and application to computational fluid dynamics and turbulent flows. *Journal of Computational Physics*, 242:623–647, 2013.

[36] Juan Du, Fangxin Fang, Christopher C Pain, I.M. Navon, Jiang Zhu, and David A Ham. POD reduced-order unstructured mesh modeling applied to 2d and 3d fluid flow. *Computers and Mathematics with Applications*, 65(3):362–379, 2013.

[37] Chen Han. Blackbox stencil interpolation method for model reduction. Master's thesis, Massachusetts Institute of Technology, 2012.

[38] D. Wirtz, N. Karajan, and B. Haasdonk. Model order reduction of multiscale models using kernel methods. Technical Report SRC SimTech, University of Stuttgart, 2013.

[39] D Wirtz and Bernard Haasdonk. Efficient a-posteriori error estimation for nonlinear kernel-based reduced systems. *Systems and Control Letters*, 61(1):203–211, 2012.

[40] Christophe Audouze, Florian De Vuyst, and Prasanth B Nair. Nonintrusive reduced-order modeling of parametrized time-dependent partial differential equations. *Numerical Methods for Partial Differential Equations*, 29(5):1587–1628, 2013.

[41] C Audouze, F De Vuyst, and PB Nair. Reduced-order modeling of parameterized PDEs using time–space-parameter principal component analysis. *International journal for numerical methods in engineering*, 80(8):1025–1057, 2009.

[42] Hector Klie et al. Unlocking fast reservoir predictions via nonintrusive reduced-order models. In *SPE Reservoir Simulation Symposium*. Society of Petroleum Engineers, 2013.

[43] S Walton, O Hassan, and K Morgan. Reduced order modelling for unsteady fluid flow using proper orthogonal decomposition and radial basis functions. *Applied Mathematical Modelling*, 37(20):8930–8945, 2013.

[44] R. Noori, A.R. Karbassi, Kh. Ashrafi, M. Ardestani, and N. Mehrdadi. Development and application of reduced-order neural network model based on proper orthogonal decomposition for BOD5 monitoring: Active and online prediction. *Environmental Progress and Sustainable Energy*, 32(1):120–127, 2013.

[45] B. R. Noack, M. Morzynski, and G. Tadmor. *Reduced-Order modelling for flow control*, volume 528. Springer, 2011.

[46] D. Xiao, P. Yang, F. Fang, J. Xiang, C.C. Pain, and I.M. Navon. Non-intrusive reduced order modelling of fluidstructure interactions. *Computer Methods in Applied Mechanics and Engineering*, 303:35 – 54, 2016.

[47] D Xiao, P Yang, F Fang, J Xiang, CC Pain, IM Navon, and M Chen. A non-intrusive reduced-order model for compressible fluid and fractured solid coupling and its application to blasting. *Journal of Computational Physics*, 330:221–244, 2017.

[48] F Fang, CC Pain, IM Navon, MD Piggott, GJ Gorman, PE Farrell, PA Allison, and AJH Goddard. A POD reduced-order 4D-Var adaptive mesh ocean modelling approach. *International Journal for Numerical Methods in Fluids*, 60(7):709–732, 2009.

[49] Dao My Ha, Pavel Tkalich, and Eng Soon Chan. Tsunami forecasting using proper orthogonal decomposition method. *Journal of Geophysical Research: Oceans (1978–2012)*, 113(C6), 2008.

[50] Jean-Marie Zokagoa and Azzeddine Soulaïmani. A POD-based reduced-order model for free surface shallow water flows over real bathymetries for monte-carlo-type applications. *Computer Methods in Applied Mechanics and Engineering*, 221:1–23, 2012.

[51] Prasad Sumant, Hong Wu, Andreas Cangellaris, and Narayana Aluru. Reduced-order models of finite element approximations of electromagnetic devices exhibiting statistical variability. *IEEE Transactions on Antennas and Propagation*, 60(1):301–309, 2012.

[52] D. Xiao, F. Fang, C.C. Pain, and I.M. Navon. A parameterized non-intrusive reduced order model and error analysis for general time-dependent nonlinear partial differential equations and its applications. *Computer Methods in Applied Mechanics and Engineering*, (minor revision), 2017.

[53] S. Chaturantabut. Dimension reduction for unsteady nonlinear partial differential equations via empirical interpolation methods. Master's thesis, Rice university, 2008.

[54] Sergey A Smolyak. Quadrature and interpolation formulas for tensor products of certain classes of functions. *Dokl. Akad. Nauk SSSR*, 4(240-243):123, 1963.

[55] Thomas Gerstner and Michael Griebel. Numerical integration using sparse grids. *Numerical algorithms*, 18(3-4):209–232, 1998.

[56] Kenneth L Judd, Lilia Maliar, Serguei Maliar, and Rafael Valero. Smolyak method for solving dynamic economic models: Lagrange interpolation, anisotropic grid and adaptive domain. *Journal of Economic Dynamics and Control*, 44:92–123, 2014.

[57] Abebe Geletu. Orthogonal Polynomials, Quadratures and Sparse-Grid Methods for Probability Integrals. *Technische Universitt Ilmenau, Institut fr Automatisierungs- und Systemtechnik Fachgebiet Simulation und Optimale Prozesse*, presentation, 2010.

[58] S.W. Funke, C.C. Pain, S.C. Kramer, and M.D. Piggott. A wetting and drying algorithm with a combined pressure/free-surface formulation for non-hydrostatic models. *Advances in Water Resources*, 34(11):1483–1495, 2011.

[59] Pain C.C., Piggott M.D., Goddard A.J.H., and et al. Three-dimensional unstructured mesh ocean modelling. *Ocean Modelling*, 10:5–33, 2005.

[60] Karl Kunisch and Stefan Volkwein. Optimal snapshot location for computing pod basis functions. *ESAIM: Mathematical Modelling and Numerical Analysis*, 44(03):509–529, 2010.

[61] Adam J Siade, Mario Putti, and William W-G Yeh. Snapshot selection for groundwater model reduction using proper orthogonal decomposition. *Water Resources Research*, 46(8), 2010.

[62] Andrea Balzano. Evaluation of methods for numerical simulation of wetting and drying in shallow water flow models. *Coastal Engineering*, 34(1):83–107, 1998.

[63] A. N. Kolmogorov. Uber die beste anniiherung von f unktionen einer gegebener funktionklasse. *Ann. Math.*, 37:107–111, 1936.

[64] P.L.F. Liu, H.H. Yeh, and C. Synolakis. *Advanced Numerical Models for Simulating Tsunami Waves and Runup*. Advances in coastal and ocean engineering. World Scientific, 2008.