A Reduced Order Model for Turbulent Flows in the Urban Environment Using Machine Learning

D. Xiao[a,b,c,*], C.E. Heaney[a], L. Mottet[a,e], F. Fang[a,c], W. Lin[f], I.M. Navon[d], Y. Guo[c], O.K. Matar[g], A.G. Robins[f], C.C. Pain[a,c]

[a] *Applied Modelling and Computation Group, Department of Earth Science and Engineering, Imperial College London, Prince Consort Road, London, SW7 2BP, UK*

[b] *ZCCE College of Engineering, Swansea University, Bay Campus, Fabian Way, Swansea SA1 8EN, UK*

[c] *Data Assimilation Lab, Data Science Institute, Imperial College London, Prince Consort Road, London, SW7 2BP, UK*

[d] *Department of Scientific Computing, Florida State University, Tallahassee, FL, 32306-4120, USA*

[e] *Department of Architecture, University of Cambridge, 1-5 Scroope Terrace, Trumpington Street, Cambridge, CB2 IPX, UK*

[f] *Department of Mechanical Engineering Sciences, University of Surrey, Guildford, UK*

[g] *Department of Chemical Engineering, Imperial College London, UK*

Abstract

To help create a comfortable and healthy indoor and outdoor environment in which to live, there is a need to understand turbulent air flows within the urban environment. To this end, building on a previously reported method [1], we develop a fast-running Non-Intrusive Reduced Order Model (NIROM) for predicting the turbulent air flows found within an urban environment. To resolve larger scale turbulent fluctuations, we employ a Large Eddy Simulation (LES) model and solve the resulting computational model on unstructured meshes. The objective is to construct a rapid-running NIROM from these results that will have 'similar' dynamics to the original LES model. Based on Proper Orthogonal Decomposition (POD) and machine learning techniques, this Reduced Order Model (ROM) is six orders of magnitude faster than the high-fidelity LES model and we demonstrate how 'similar' it can be to the high-fidelity model by comparing statistical quantities such as the mean flows, Reynolds stresses and probability densities of the velocities. We also include validation of the high-fidelity model against data from wind tunnel experiments.

This paper represents a key step towards the use of reduced order modelling for operational purposes with the tantalising possibility of it being used in place of Gaussian plume models, and the potential for greatly improved model fidelity and confidence.

Keywords: Non-intrusive Reduced Order Modelling, urban flows, Proper Orthogonal Decomposition, machine learning, Gaussian Process Regression, operational modelling

*Corresponding author

Email address: dh.xiao@imperial.ac.uk (D. Xiao)

1. Introduction

Turbulent flows are important in many engineering fields such as geophysics, astrophysics and meteorology [2]. For example, understanding how turbulent flow moves around buildings or cities is increasingly important to policy makers and environmental engineers for designing new urban spaces that provide comfortable and healthy indoor and outdoor built environments [3]. Numerical simulation is one of the main tools allowing us to understand urban turbulent flows and has demonstrated its importance for a wide range of research areas such as atmospheric physics [4], pollution dispersion [5, 6, 7] and urban planning [8, 9].

The prediction of air flow around buildings and in street canyons is challenging due to the chaotic nature of turbulent flows [5] and the complexity of geometries in the urban environment [3, 10]. These flows have a wide range of spatial and temporal scales [11, 5], all of which influence the local flow dynamics. A detailed description of turbulent flow is important in order to model accurately the dispersion or accumulation of pollution. However, capturing the higher order moments that contribute to turbulence represents a challenge for any numerical model as they are notoriously difficult to reproduce [6]. Two main approaches exist for the modelling of turbulent flows: the Reynolds-Averaged Navier-Stokes (RANS) approach and the Large Eddy Simulation (LES) approach. It has been shown that LES methods are more accurate than RANS methods and are able to capture the unsteady fluctuation terms of turbulence, the turbulent mixing processes and the transport of pollution in an urban landscape [12, 13, 14, 15, 11, 16].

In comparison with 2D modelling, 3D urban flow modelling provides better understanding and much more information about local flow structure, vertical inertia, unsteady dynamics and pollution fluctuations. However, the majority of existing 3D Computational Fluid Dynamics (CFD) urban flow models suffer from an extremely high computational cost and cannot respond rapidly enough for real-time forecasting [17, 18]. Model reduction technology is therefore being developed to mitigate the computational cost, since it offers the potential to simulate complex systems with substantially reduced computational requirements.

Among model reduction methods, Proper Orthogonal Decomposition (POD) has proven to be an efficient means of deriving the reduced basis functions for physical systems [19, 20]. In this work we choose to derive POD basis functions of all velocity components combined, motivated by the fact that the velocity components are physically correlated. This will reduce the amount of training data required and also the number of POD basis functions required, see [21, 22]. POD and its variants have been successfully applied to a number of research fields, for example, pattern recognition [23], fluids problems [24, 25], air pollution [26], shape optimisation [27], aerospace design and optimisation [28], and shallow water equations [29, 21, 22, 30]. POD is often used in combination with Galerkin projection to form intrusive ROMs, however, for non-linear problems, the resulting model can suffer from instability and inefficiency issues. Various methods for improving stability have been proposed such as Petrov−Galerkin projection [24, 31], regularisation methods [32], the introduction of numerical diffusion [33, 34] and Fourier expansion of transfer functions [35]. To increase the efficiency of the solution of the non-linear terms, a number of methods have been proposed,

such as the empirical interpolation method [36], the discrete empirical interpolation method [37], Gauss−Newton method with approximated tensors [38], the quadratic expansion method [39, 40] and the residual discrete empirical interpolation method [41]. Intrusive ROMs have been developed to model air flows inside and outside of buildings and also for environmental modelling. For example, Wu *et al.* [42] proposed a fast and accurate method to model turbulent flows in a square duct with high Reynolds numbers using improved wall-modelled LES. Cao *et al.* [17] used a discrete Green's function method to derive a low dimensional indoor ventilation model. Liu *et al.* [18] used a fast fluid dynamics and a local searching method to derive a fast model for indoor airflow simulation. Vervecken *et al.* [43] proposed a stable ROM for pollutant dispersion outdoors, which is derived by projecting the transient advection-diffusion equation onto a Krylov subspace with an Arnoldi algorithm. Cao *et al.* [44] proposed a ROM for indoor pollutant dispersion using a number of eigenmodes to represent the system in a manner similar to POD. Fang *et al.* [26] produced an intrusive ROM which was capable of modelling tracer dispersion in urban street canyons.

Intrusive ROMs require modification of the source code of the high-fidelity model. For complex codes this can be troublesome, and, furthermore, maintaining these modifications can be cumbersome [45]. To circumvent this shortcoming, non-intrusive model reduction approaches have been developed. As a precursor to such methods, Wirtz *et al.* [46, 47] proposed kernel methods, based on support vector machines and a vectorial kernel greedy algorithm. Audouze *et al.* [48] proposed a NIROM for nonlinear parametrised time-dependent PDEs using radial basis functions (RBF) and POD. Walton *et al.* [49] developed a NIROM for unsteady fluid flows using the RBF interpolation and POD. Noori *et al.* [50] and Noack *et al.* [51] chose a neural network from which to construct a NIROM. Xiao *et al.* [25] presented a NIROM for the Navier-Stokes equations based on POD and RBF interpolation and applied it successfully to fluid-structure interaction problems [52].

Machine learning is gaining more and more attention, and has been applied successfully to a number of research fields, such as image recognition [53, 54], driver-less cars [55], assessing the visual environment of cities [56], personal comfort model [57], speech recognition [58], language processing [59] and the control of complex non-linear systems [60]. There exist a number of open-source machine learning libraries, for example, TensorFlow [61] and Keras [62], both used here. In this work, to construct our NIROM, we use a machine learning method called Gaussian Process Regression (GPR). Although the NIROM could be formed by other means, such as Smolyak sparse grids [63] or the quadratic perturbation approach [41], in this paper we exploit the high-dimensional surface fitting capabilities of machine learning. The GPR method captures the uncertainty in the data directly, and produces good fits when the number of data points is not large [64].

The aim of this work is to demonstrate a key step towards the use of Reduced Order Models (ROMs) in operational modelling with the tantalising possibility, for urban flows, of replacing Gaussian plume models [65]. The introduction of ROMs could lead to greatly improved model fidelity and confidence for air flow and pollution modelling in urban environments. When developing an operational model based on ROM, the model should (1) be able to reproduce the high-fidelity solutions that have been used to train/develop the model and (2) have 'similar' dynamics to the high-fidelity

model for as long a time as the NIROM is run. In this paper we demonstrate that the proposed NIROM has both these properties, by comparing the statistics of mean flows and Reynolds stress distributions as well as velocity time series, probability density functions and frequency spectra. Our ultimate goal is to model a range of parameters of interest, such as wind direction, buoyancy and inlet conditions. This is beyond the scope of the current paper, which focuses on urban planning or operational modelling applications where one wind direction is sufficient. To this end, buoyancy is neglected, and the inlet conditions and wind direction are fixed. We also fix the geometry and assume incompressible flow. Future studies will introduce these effects as part of the parametrisation of the physics.

In this work, a Gaussian Process Regression (GPR) machine learning method is combined with POD to construct a NIROM, which is applied to urban flows. It is able to provide us a real-time tool to understand the urban turbulence flows and help us live in a comfortable and healthy indoor and outdoor built environments. The structure of the paper is as follows. Section 2 presents the governing equations of urban flows. Section 3 presents the derivation of the POD model reduction and re-formulation of the governing equations using the Gaussian Process Regression method. Section 4 illustrates the methodology derived above through the challenging problem of modelling urban air flow in the neighbourhood of London South Bank University. We also include a validation of the high-fidelity model against physical measurements from a wind tunnel experiment. In section 5, we discuss the results and in section 6, conclusions are presented and the novelty of the manuscript is highlighted.

2. Governing equations of a 3D large eddy simulation model

This work considers the 3D non-hydrostatic Navier-Stokes (NS) equations describing the conservation of mass and momentum of a fluid,

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

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + \nabla \cdot \boldsymbol{\tau}, \tag{2}$$

where $\mathbf{u} \equiv (u, v, w)^T$ denotes the velocity vector, $p = \tilde{p}/\rho_0$ denotes the normalised pressure, $\tilde{p}$ denotes the pressure, ρ_0 represents the constant reference density and $\boldsymbol{\tau}$ represents the stress tensor. We apply filtering to the NS equations in order to model the fine scale behaviour (i.e. the fluctuations that occur on scales smaller than the grid scale), so the stress tensor includes an LES sub-grid scale viscosity calculated by an anisotropic Smagorinsky model [66].

The discretised form of the system can be written as

$$C^T \mathbf{u} = \mathbf{0}, \tag{3}$$

$$M \frac{\partial \mathbf{u}}{\partial t} + A(\mathbf{u})\mathbf{u} + K\mathbf{u} + C\mathbf{p} = \mathbf{0}, \tag{4}$$

where C denotes a pressure gradient matrix, M is the mass matrix, $A(\mathbf{u})$ is the solution-dependent

streaming operator and K is the matrix related to the remaining linear velocity terms. The velocity, $\mathbf{u}$, is now a vector containing nodal values of all three components, likewise, $\mathbf{p}$ is a vector containing the nodal pressure values.

3. Reduced Order Modelling

In this section, we derive a Reduced Order Model for 3D flows in the urban environment using POD and a machine learning method called Gaussian Process Regression (GPR). First, we describe how POD is used to obtain basis functions from the high-fidelity model. We briefly describe GPR and then explain the construction of the NIROM which is done by training a neural network using GPR with data from the high-fidelity model. These two steps (finding the POD basis functions and training the neural network) make up the off-line stage of the reduced order model, which is expected to be computationally intensive. The on-line stage, which involves running the NIROM, is much less computationally intensive due to the reduced dimension of the model. We finish this section by presenting the algorithm which describes the on-line stage.

3.1. Proper Orthogonal Decomposition

In order to obtain the POD basis functions that are used to represent the behaviour of the system, we apply a Singular Value Decomposition (SVD) to a snapshots matrix, whose columns are snapshots of the solution to the high-fidelity model taken at certain times. Rather than finding basis functions for each velocity component independently, we consider all the components simultaneously. This approach was motivated by the desire to capture the physical correlations which arise naturally between the velocity components. This may lead to less data being required to train the neural network and fewer basis functions being required to represent the model (to a given accuracy). The form of the snapshots matrix is therefore

$$S = \begin{bmatrix} \mathbf{u}^1 & \mathbf{u}^2 & \cdots & \mathbf{u}^{\mathcal{N}_s} \end{bmatrix} \tag{5}$$

where $\mathcal{N}_s$ is the total number of snapshots and the velocity at the nth time level, $\mathbf{u}^n$, has the form

$$\mathbf{u}^n = \{u_1^n, u_2^n, \ldots, u_{\mathcal{N}}^n, v_1^n, v_2^n, \ldots, v_{\mathcal{N}}^n, w_1^n, w_2^n, \ldots, w_{\mathcal{N}}^n\}^T, \tag{6}$$

in which $\mathcal{N}$ is the number of nodes and u_i^n is the value of the horizontal velocity component at time level n and node i.

Applying an SVD to the snapshots matrix results in

$$S = U\Sigma V^T, \tag{7}$$

where $U \in \mathbb{R}^{3\mathcal{N}\times 3\mathcal{N}}$ and $V \in R^{\mathcal{N}_s\times\mathcal{N}_s}$ are matrices whose columns are orthogonal, and $\Sigma \in \mathbb{R}^{3\mathcal{N}\times\mathcal{N}_s}$ has zero entries except for the leading diagonal, on which are the singular values in order of decreasing magnitude. The POD basis functions are the first $\mathcal{N}_s$ columns of U. The number of basis

functions can be reduced if some singular values are small in comparison with a given tolerance. For instance, for a tolerance $\eta \lesssim 1$, we set the number of POD basis functions to be the smallest integer value of $m \leqslant \mathcal{N}_s$ such that

$$\frac{\sum_{j=1}^{m} \sigma_j^2}{\sum_{j=1}^{\mathcal{N}_s} \sigma_j^2} \geqslant \eta\,, \tag{8}$$

where the quantity on the left-hand side of the equation is the fraction of total energy captured by the first m POD basis functions. With only m non-zero singular values, the truncated snapshots matrix, S_m, formed by taking the product of U, Σ (with m non-zero entries) and V^T is a rank m approximation of S. This is an optimal approximation in the sense that no other rank m matrix can be closer to the snapshot matrix S in the Frobenius norm. For computational efficiency, we carry out an eigen-decomposition of the matrix $S^T S$ rather than performing an SVD upon S. For more details on this procedure see algorithm 1 in [22] for instance. Although we solve for velocity and pressure in the high-fidelity model, the reduced model is formulated entirely in terms of velocity, so we only require POD basis functions and coefficients for velocity.

In POD, a variable $\mathbf{v}$ can be expressed by the expansion,

$$\mathbf{v} = \overline{\mathbf{v}} + \sum_{j=1}^{m} \alpha_j \boldsymbol{\phi}_j \tag{9}$$

where α_j denotes the jth coefficient of the POD expansion, $\boldsymbol{\phi}_j$ represents the jth POD basis function and $\overline{\mathbf{v}}$ is the mean of snapshots for the variable $\mathbf{v}$. The snapshots satisfy this expression to machine precision if no truncation takes place, that is $m = \mathcal{N}_s$. For the velocity solution at other time levels, this is an approximation. In some cases, the results may be more accurate if the mean of snapshots is not included in the above expansion [67], and indeed, we omit the mean value in the model formulation presented here.

3.2. Using Gaussian Process Regression to Construct the NIROM

Gaussian Process Regression (GPR) uses a linear combination of Gaussian-shaped basis functions to provide the surface representation necessary for this application [68]. The main advantages of GPR are that: it typically does not require much data to provide accurate surfaces; it is linear and, thus, does not require extensive training. By contrast, feed-forward neural networks require large amounts of data. They involve the solution of an optimisation problem to form the surface fitting, and there can be hard-to-tune generalisation parameters, e.g. based on weight decay. However, feed-forward neural networks are often effective at very high-dimensional surface fitting. In this paper, just 500 data points are used in training by the GPR. This is shown to be sufficient in the results section, where it is seen that the NIROM has enough information in order to exhibit the same quasi-steady-state statistics as the high-fidelity model. To perform the GPR required to generate our NIROMs we used the open-source Keras library [62].

3.3. Producing a Non-Intrusive Reduced Order Model

Having obtained the POD basis functions in section 3.1 we complete the off-line stage of NIROM by approximating the governing equations. We do this by training a neural network to predict how the governing equations would behave. All snapshots are projected onto reduced space (spanned by the POD basis functions) and are used to train the neural network. This training procedure results in a function f_j for each POD basis function, which maps the set of POD coefficients from one time level ($\boldsymbol{\alpha}^{k-1}$) to the associated POD coefficient at the next time level (α_j^k), i.e.

$$\alpha_j^k = f_j(\boldsymbol{\alpha}^{k-1}) = f_j(\alpha_1^{k-1}, \alpha_2^{k-1}, \ldots, \alpha_m^{k-1}), \qquad \forall k \in \{1, 2, \ldots, \mathcal{N}_s\}\,. \tag{10}$$

By including the initial condition, we have $\mathcal{N}_s$ pairs of input and output data that are used to form the function f_j

$$\text{input:} \qquad \boldsymbol{\alpha}^{k-1} = \left(\alpha_1^{k-1}, \alpha_2^{k-1}, \ldots, \alpha_m^{k-1}\right) \tag{11}$$

$$\text{output:} \qquad \alpha_j^k\,, \tag{12}$$

for all $k \in \{1, 2, \ldots, \mathcal{N}_s\}$. This procedure is repeated for each POD coefficient (i.e. for $j \in \{1, 2, \ldots, m\}$), and once all the functions $\{f_j\}_{j=1}^m$ have been determined, the off-line stage is complete. It is summarised in the flowchart given in figure 1.

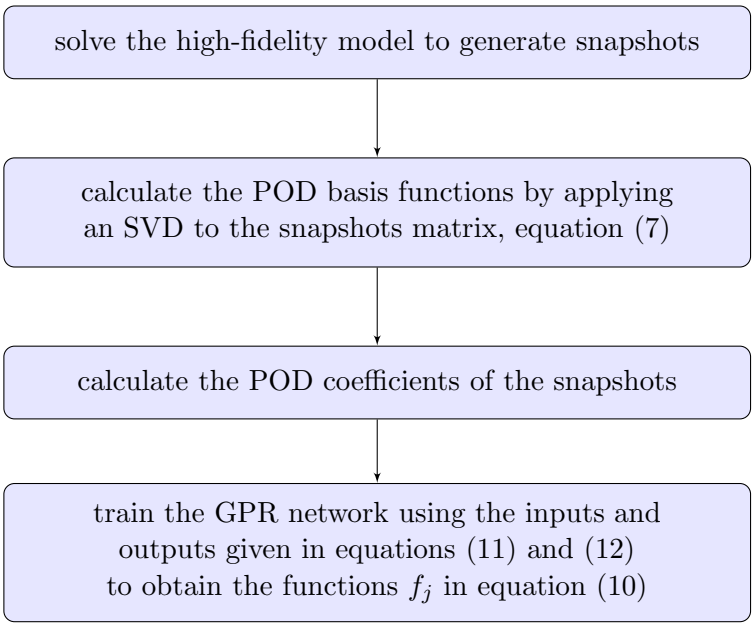


Figure 1: The off-line stage of the NIROM as described in sections 3.1 and 3.3.

3.4. Running simulations with the non-intrusive reduced order model

For running the NIROM (i.e. the on-line stage), the functions $\{f_j\}_{j=1}^m$ are treated as response functions allowing the POD coefficients at one time level to be predicted given those at a previous

time level

$$\alpha_j(t + \Delta t) = f_j(\boldsymbol{\alpha}(t)) \quad \forall j \in \{1, 2, \ldots, m\}\,. \tag{13}$$

We remark that when running the NIROM, the time step, Δt, will coincide with that of the high-fidelity model. The procedure of on-line prediction using the NIROM is summarised in Algorithm 1. The number of time levels used in the on-line stage, $\mathcal{N}_t$, is independent of the number of time levels used in training. That is, the NIROM can be run for as short a time or as long a time as desired. The initial condition can be different to that used in the high-fidelity model.

Algorithm 1: On-line NIROM calculation

!! *The hypersurfaces,* $\{f_j\}_{j=1}^m$*, and the POD basis functions,* $\{\boldsymbol{\phi}_j\}_{j=1}^m$*, are known.*

!! *The initial condition (*$\boldsymbol{\alpha}^0$*), time step (*Δt*), initial time (*t_0*) and number of time steps (*$\mathcal{N}^t$*) are given.*

for $n = 1$ *to* $\mathcal{N}_t$ **do**
  $t = t_0 + n\Delta t$!! *Current time*

  !! *Step (a): calculate the POD coefficients,* $\boldsymbol{\alpha}^n$*, at the current time step:*
  for $j = 1$ *to* m **do**
    $\alpha_j^n = f_j(\alpha_1^{n-1}, \alpha_2^{n-1}, \ldots, \alpha_m^{n-1})$
  endfor

  !! *Step (b): obtain the solution* $\mathbf{u}^n$ *in the full space at the current time,* t*, by projecting* α_j^n *onto the full space using* $\mathbf{u}^n = \sum_{j=1}^m \alpha_j^n \boldsymbol{\phi}_j$*:*
  $\mathbf{u}^n = \mathbf{0}$
  for $j = 1$ *to* m **do**
    $\mathbf{u}^n = \mathbf{u}^n + \alpha_j^n \boldsymbol{\phi}_j$
  endfor

endfor

4. Modelling air flow around London South Bank University

In this section we study the air flow around buildings in an area around London South Bank University (LSBU). We present a validation of the high-fidelity model; we demonstrate the ability of NIROM to reproduce snapshots; and we show that NIROM is capable of making predictions beyond the range of the snapshots.

The system of equations we solve is chaotic, so mesh convergence studies cannot be easily performed [69]. In order to mitigate this fact we use mesh adaptivity to optimise the initial mesh and we also compare the high-fidelity model to time-averaged data from wind tunnel experiments. Given the complexity of the LSBU test case, presenting a full validation of the high-fidelity model represents a large undertaking and will therefore be the subject of a future paper. In addition, performing error analyses both of the high-fidelity model and the NIROM is complicated and outside the scope of this paper.

4.1. Validation of the high-fidelity model

The LSBU test site covers a region of 1 km in diameter with the tallest building measuring 81 m in height. A scaled down version of the test site has been constructed at the EnFlo Meteorological Wind Tunnel, a UK National Centre for Atmospheric Sciences facility at the University of Surrey, shown in figure 2. Vertical profiles of mean velocity and turbulence intensity were generated in the laboratory (see figure 3) in order to simulate the atmospheric boundary layer winds that approach buildings in London Southwark. Appropriate flow profiles were developed using a suckdown fan to draw air through a wind tunnel working section of 20 m length, and over spires at the inlet of the working section and roughness elements in staggered arrangement along the initial 12 m of fetch. The width and height of the working section were 3.5 m and 1.5 m respectively. Lengths in the wind tunnel model were reduced from full-scale by a factor of 200. The mean wind speed at the top of the boundary layer was set to $2.00\,\mathrm{m\,s^{-1}}$. Wind speeds near ground level in the building array were highly variable and typically in the range of $0.5\,\mathrm{m\,s^{-1}}$ to $1\,\mathrm{m\,s^{-1}}$. The test conditions corresponded to neutral atmospheric stability.

The high-fidelity model results used in to construct the NIROM are generated by Fluidity, a finite element code with the ability to solve on unstructured meshes, developed by the Applied Modelling and Computation Group at Imperial College London [70]. For the purposes of the validation, the computational domain is set to be the same size as the wind tunnel, and the mean streamwise velocity and turbulence intensities at the inlet are shown in Figure 3. In the computational model, the other velocity components were set to zero. The xx, yy and zz components of Reynolds stresses and turbulent length scales are also given in this figure. Off diagonal terms are taken to be zero. The filled circles represent measurements from the wind tunnel, and these are linearly interpolated to form the inlet conditions for the computational model. Zero velocity is prescribed on the bottom and wall boundaries and zero stress conditions at the outlet which sets $p = 0$ at the outflow boundary. Perfect slip conditions are used on the vertical lateral boundaries as well as top boundary of the domain. This is because we are not interested in the far field flows generated by these boundary

Figure 2: The photograph shows the wind tunnel model of turbulent atmospheric boundary-layer wind approaching London Southwark from the northwest.

conditions and perfect slip reduces the length scales that need to be resolved. The P1-P1 finite element pair is used for the spatial discretisation and the Crank-Nicolson scheme is used for the temporal discretisation. For more details see [70].

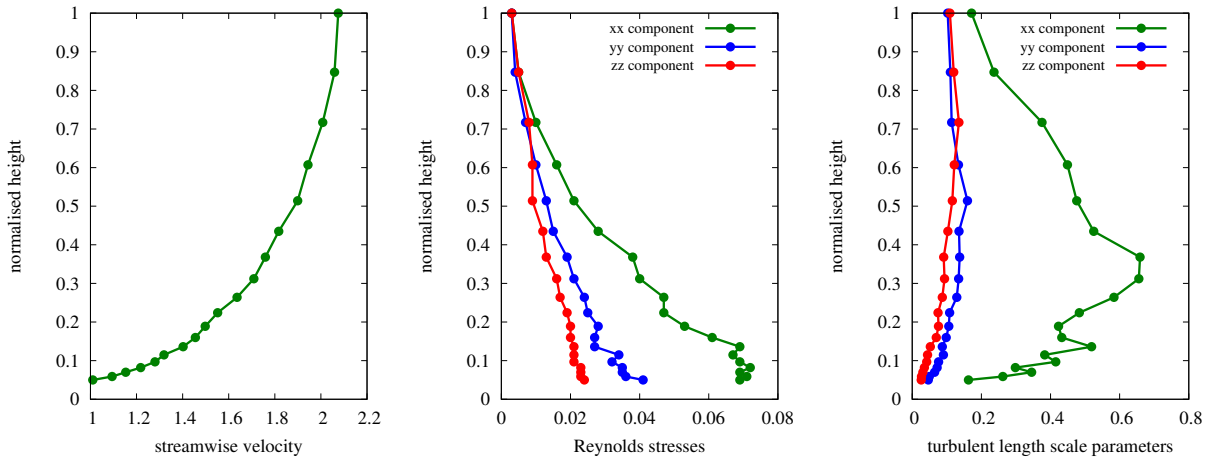


Figure 3: Vertical profiles of mean streamwise velocity (left), Reynolds stresses (middle) and their associated length parameters (right) used at the inlet. The measurements taken from the wind tunnel are shown as filled circles. These points are linearly interpolated and imposed at the inlet of the computational model.

Mesh adaptivity is used here to refine the mesh in regions where important small-scale physical processes occur, and to coarsen elements elsewhere, see [71]. The transition from finer to coarser regions is kept smooth by using an anisotropic gradation parameter in the adaptivity algorithm. Mesh adaptivity has a large number of associated parameters: maximum and minimum edge lengths, gradation parameter, interpolation errors, maximum aspect ratio. We will not explore this parameter

space here as it is an involved procedure and is not the focus of this paper. Such a study has been performed in [72] in which a range of adaptivity settings are investigated for the test case of air flow around a building. In the example considered here, the mesh was adapted every 10 time levels with a minimum edge length of 1.47×10^{-3} m and a maximum edge length of 0.15 m. The velocity interpolation error bound was set to $0.05\,\mathrm{m\,s^{-1}}$ and the maximum number of nodes was set to 2 million.

Two wind directions were used for the validation seen in figure 5, north-westerly (left) and south-westerly (right). The plots show the mean streamwise velocity profiles after normalisation at two points in the domain (see figure 4). There is promising agreement between the experiment and computations for the general profile shapes. The LES model (high-fidelity model) underpredicts the velocity at street level and slightly overpredicts at heights above the urban canopy. For more information about the wind tunnel experiments and the computational modelling, the reader is referred to [7].

Figure 4: This figure shows the two points where the validation is performed: a pink dot represents point 1 and a blue dot represents point 2.

4.2. NIROM results

To generate the high-fidelity model results required for the NIROM, we use the Fluidity code as in the previous section. The computational model used to generate the snapshots is at the full scale using a domain size of $[0, 2041] \times [0, 2288] \times [0, 250]$ (metres). In keeping with standard CFD conventions, the distance between the buildings and the boundaries is larger than in the previous section, where the purpose was to model the set-up in the wind tunnel as closely as possible. For further information about accepted CFD practices is outlined in [73]. The boundary conditions are those given in the previous section with two exceptions. First, the wind direction is westerly and second, the turbulent inlet boundary condition is now based on a synthetic eddy method as presented in [74]. It was demonstrated that this method is able to reproduce the atmospheric

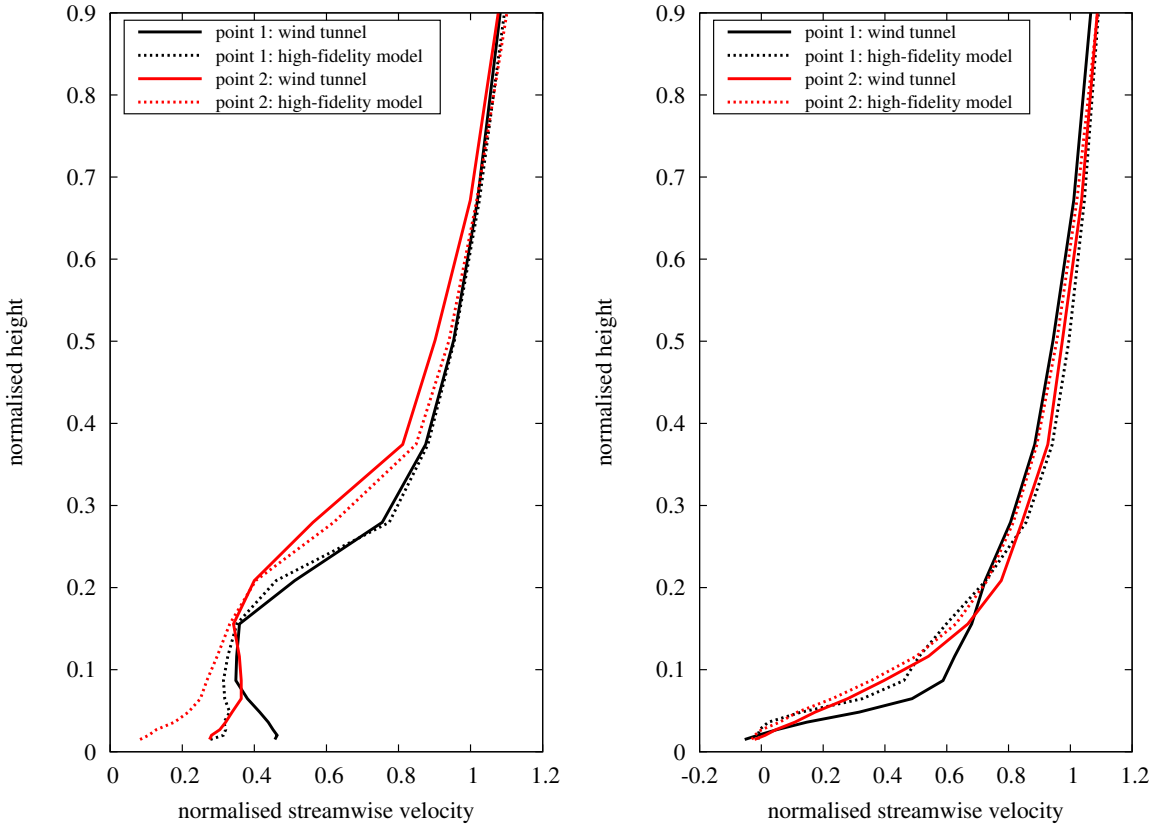


Figure 5: Streamwise mean velocity profiles from the wind tunnel experiment (solid lines) and the high-fidelity model (dotted lines) are shown at two points shown in figure 4. The plot on the left shows results for a north-westerly wind direction and the plot on the right shows results for a south-westerly wind.

boundary layer accurately. This method is based on four given parameters: the number of eddies (a large number taken to be 4000 in our simulations); the mean velocity profiles

$$(u, v, w) = \left(0.97561 \ln\left(\frac{z}{0.01}\right), 0, 0\right), \tag{14}$$

where z is the height; profiles of the Reynolds stresses, $\boldsymbol{u'u'}$, and the associated length scales, $\boldsymbol{L}$,

$$\boldsymbol{u'u'} = \begin{pmatrix} 0.8 & 0 & 0 \\ 0 & 0.8 & 0 \\ 0 & 0 & 0.8 \end{pmatrix}, \tag{15}$$

$$\boldsymbol{L} = \begin{pmatrix} 100 & 0 & 0 \\ 0 & 100 & 0 \\ 0 & 0 & 100 \end{pmatrix}. \tag{16}$$

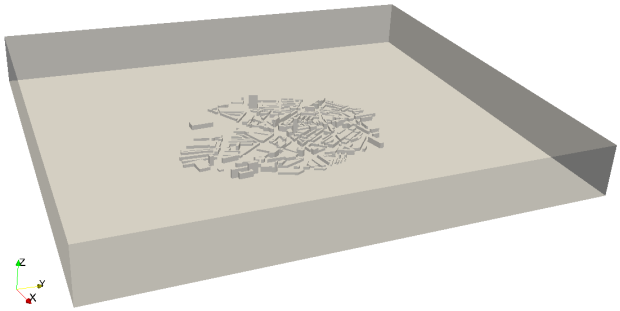
Ideally, the snapshot solutions will all reside on the same mesh. Therefore, unlike the validation results, in this section the use of mesh adaptivity is confined to an 'initialisation period'. During this period, an adaptive mesh LES simulation was performed for one hour (in real time), which is sufficient for the flow statistics to reach a quasi-steady state. From this point onwards, the mesh is fixed, as shown in Figure 6. The adaptivity settings used in this case were a minimum edge length of 0.3 m and a maximum edge length of 50 m. The velocity interpolation error bound was set to $0.3\,\mathrm{m\,s^{-1}}$ and the maximum number of nodes was set to 1 million.

After the initialisation stage, the simulation is continued for 2000 seconds with a time-step size of $\Delta t = 4/3$ s on the fixed (unstructured) mesh of 767,559 nodes. During this time, snapshots were taken every 4 seconds from the high-fidelity simulation results and used to train the NIROM. Once the NIROM has been constructed, it is then used to predict for a further 2000 seconds, over the time period $[2000, 4000]$ seconds. During this time the NIROM is purely predicting and has seen no results from the high-fidelity model. Times within the training period are often referred to as 'seen', whereas times that have not been used to train the NIROM are referred to as 'unseen'. Figure 7 illustrates when both the high-fidelity model and NIROM were active.

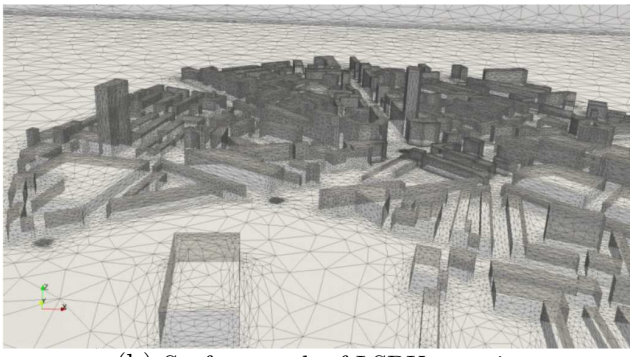
Figure 8 shows the singular values and logarithmic singular values of the snapshots. The initial, rapid reduction in the magnitude of the singular values, flattens off considerably after about the 8th POD basis function. This suggests that one needs to have a large number of basis functions to accurately represent the dynamics, and this is borne out by the results shown later in this section. We will construct NIROM models with 24, 96, 192, and 382 POD basis functions, which corresponds to capturing 29.06%, 50.537%, 70.91%, 88.63% and 97.3% respectively of the energy of the snapshots, see equation (8).

In order to visualise the jth POD basis function, in a similar manner to taking the magnitude of a vector field, we plot the following quantity

$$\sqrt{(\boldsymbol{\phi}^u_{j,i})^2 + (\boldsymbol{\phi}^v_{j,i})^2 + (\boldsymbol{\phi}^w_{j,i})^2} \quad \forall i \in \{1, 2, \ldots \mathcal{N}\} \tag{17}$$

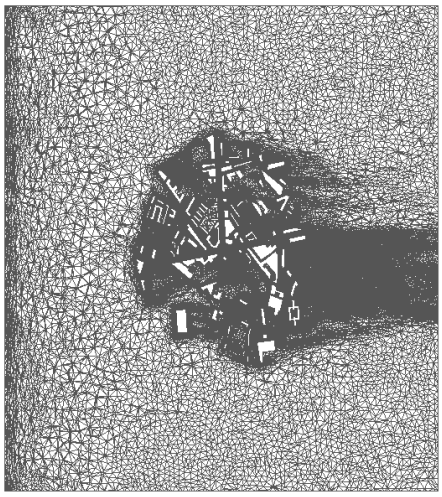


(a) Computational domain of LSBU test site



(b) Surface mesh of LSBU test site

(c) Vertical slice



(d) Horizontal slice

Figure 6: The plots show (a) the computational domain of the LSBU test site (b) the surface mesh of the test site (c) the mesh on a vertical slice through the centre-line of the tallest building and parallel to the streamwise direction and (d) a horizontal slice at a height of 15 m above the ground.

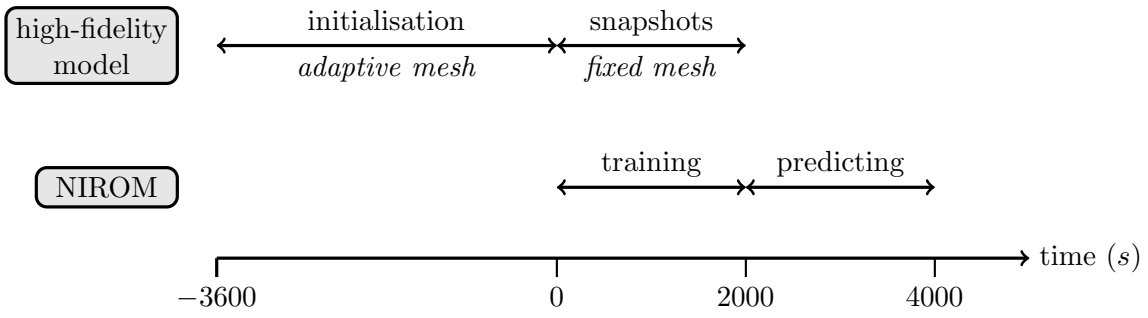
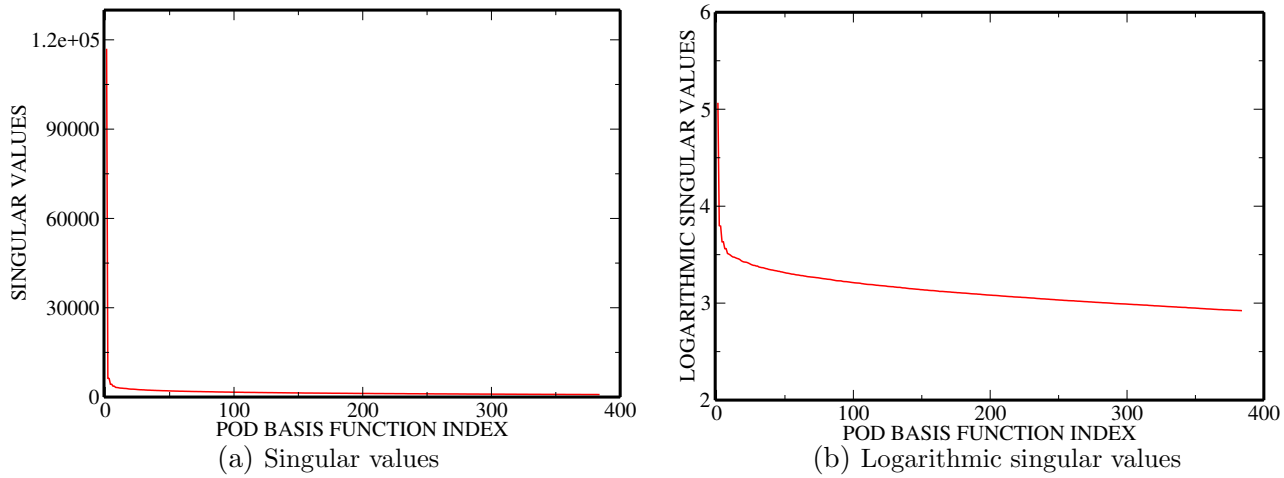


Figure 7: This figure indicates which models are active during the three time periods shown. For the first, initialisation period, $t \in [-3600, 0]$, the high-fidelity model is run with adaptive meshing until it reaches a quasi-steady state. For $t \in [0, 2000]$ the high-fidelity model continues with a fixed mesh. Snapshots are taken from these results and will be used to train the NIROM. For $t \in [2000, 4000]$ the NIROM is used to predict into the future.



(a) Singular values (b) Logarithmic singular values

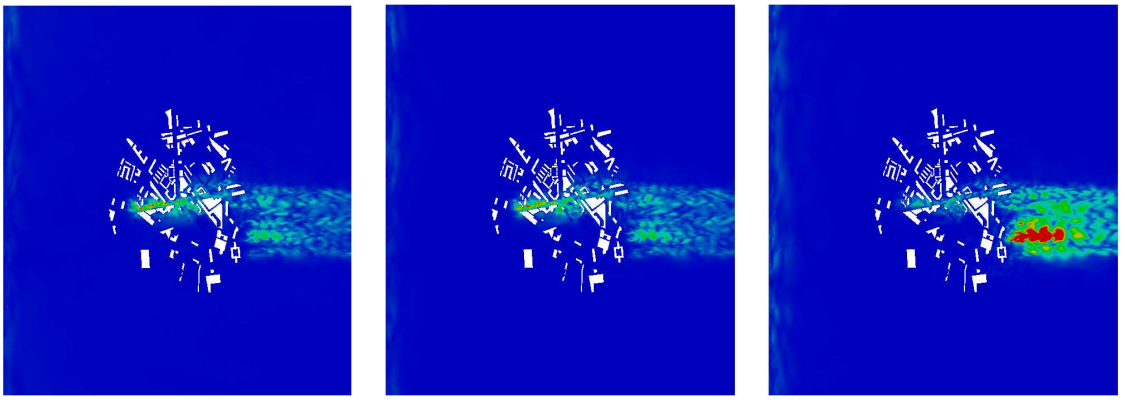
Figure 8: This figure shows the singular values and logarithmic singular values.

where i represents a node and the structure of the jth basis function is $\boldsymbol{\phi}_j = (\boldsymbol{\phi}_j^u, \boldsymbol{\phi}_j^v, \boldsymbol{\phi}_j^w)^T$. Visualised in this way, Figure 9 shows the first, second, third, fourth, fifth and seventh POD basis functions of the velocity field on a horizontal plane at 15 m above ground level, and Figure 10 shows the same, but on a vertical plane that passes through the centre-line of the tallest building and is parallel to the streamwise direction. In these figures, we see the vortex-dominated areas populated with highly oscillatory distributions. Some of the POD basis functions appear similar (at this height, at least) as they are focused on resolving the detailed eddy structures that are advected downstream. These eddies can be picked out in the POD basis functions and can be seen to be slightly displaced from one basis function to the next (e.g. compare the first basis function with the second, and the third basis function with the fourth).

Figures 11 and 12 show the speed of the flow on a horizontal plane at a height of 15 m above the ground and on a vertical plane through the centre-line of the tallest building (where the turbulent intensity is at its highest). Results were obtained from the high-fidelity model and NIROM with 24, 96, 192 and 382 basis functions at a 'seen' time level equivalent to 26.6 minutes. The NIROM with 382 basis functions exhibits excellent agreement with the high-fidelity model.

While the POD basis functions may not be ideally suited for representing the individual eddies that are shed off the tall building, the NIROM does capture these eddies more satisfactorily as the number of POD basis functions is increased, the best result being obtained with 382 basis functions. Using fewer basis functions tends to lead to either a misrepresentation of the smaller eddies or a reduction in intensity of the larger eddies. However, even with a smaller number of basis functions, the NIROM is often able to predict the propagation and position of the largest eddies when compared with the high-fidelity model. In addition, the coarser models seem to be more dissipative than the finer models, which, we speculate, contributes to their stability.

Having compared the high-fidelity model and NIROMs over the training period in Figures 11 and 12, we now assess how well the NIROM predicts for times outside the training period, i.e. during the interval $[2000, 4000]$, in which NIROM is predicting rather than reproducing training data. We



(a) First basis function (b) Second basis function (c) Third basis function

(d) Fourth basis function (e) Fifth basis function (f) Seventh basis function

Figure 9: The plots displayed above show the first, second, third, fourth, fifth and seventh POD basis functions of velocity, on a horizontal plane at a height of 15 m above the ground. The results were visualised using equation (17).

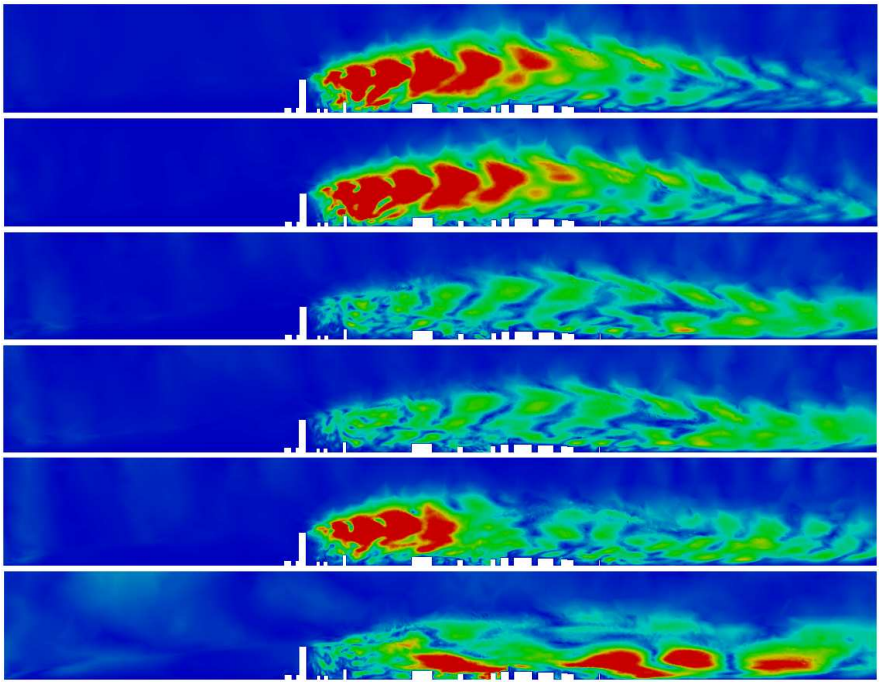
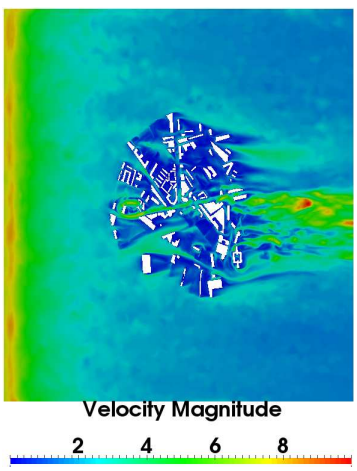


Figure 10: The plots displayed above show, from top to bottom, the first, second, third, fourth, fifth and seventh POD basis functions of velocity on a plane through the centre-line of the tallest building and parallel to the streamwise direction. The results were visualised using equation (17).

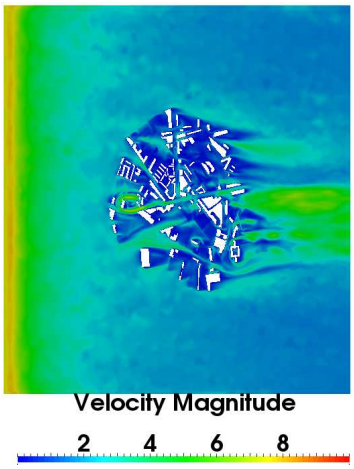
compare statistics from the high-fidelity model taken over the time interval $[0, 2000]$ and from NIROM taken over $[2000, 4000]$. Figures 13 and 14 show the magnitude of the time-averaged velocity field for the high-fidelity model over the time period $[0, 2000]$ seconds, and for NIROM with 24 and 382 POD basis functions over the predictive time period of $[2000, 4000]$ seconds. Figure 13 shows the results on a horizontal plane at a height of 15 m, Figure 14, a vertical plane parallel to the streamwise direction and passing through the centre-line of the tallest building. As shown in these figures, the time-averaged velocity field predicted by both NIROMs that are shown is in close agreement with the high-fidelity model, even though the time period over which the averaging is performed is different (i.e. over $[0, 2000]$ for the high-fidelity model and over $[2000, 4000]$ for the NIROMs). This confirms that the models have reached a quasi-steady state statistically.

Figure 15 shows a comparison between the high-fidelity model and the NIROM with 382 basis functions of the mean streamwise velocity profile for a westerly wind direction, taken at the two observation points indicated in figure 4. The models are in extremely close agreement as expected, as all the NIROMs predict the mean velocity profiles very well (figures 13 and 14).

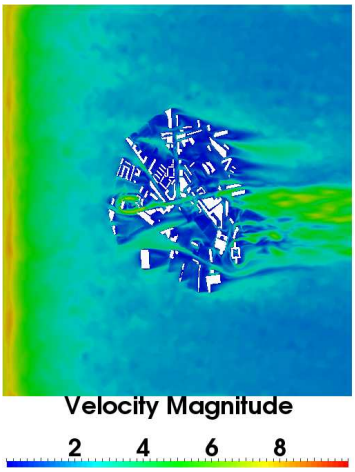
Figure 16 shows the magnitude of the instantaneous velocity obtained from the high-fidelity model at time levels $t = 400\,\text{s}$, $404\,\text{s}$, $408\,\text{s}$, $412\,\text{s}$, $416\,\text{s}$, whilst Figure 17 shows the magnitude of instantaneous velocity obtained from the NIROM using 382 basis functions at time levels $t = 2200\,\text{s}$, $2204\,\text{s}$, $2208\,\text{s}$, $2212\,\text{s}$, $2216\,\text{s}$. Both the high-fidelity model and the NIROMs show the propagation of the complex eddies through the domain in a similar manner. Comparing the high-fidelity model



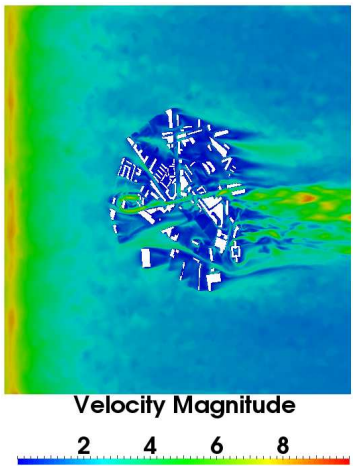
(a) High-fidelity model



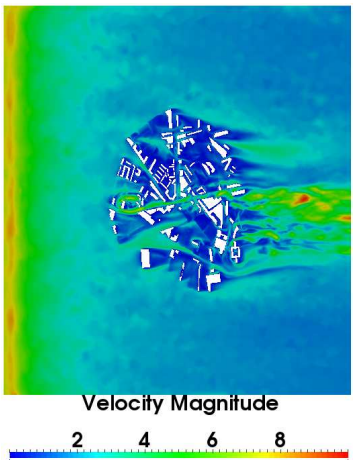
(b) NIROM, 24 basis functions



(c) NIROM, 96 basis functions



(d) NIROM, 192 basis functions



(e) NIROM, 382 basis functions

Figure 11: The plots displayed above show the magnitude of the velocity field on a horizontal plane, 15 m above the ground, obtained from the high-fidelity model, and from NIROM with 24, 96, 192 and 382 basis functions at the seen time of 26.6 min.

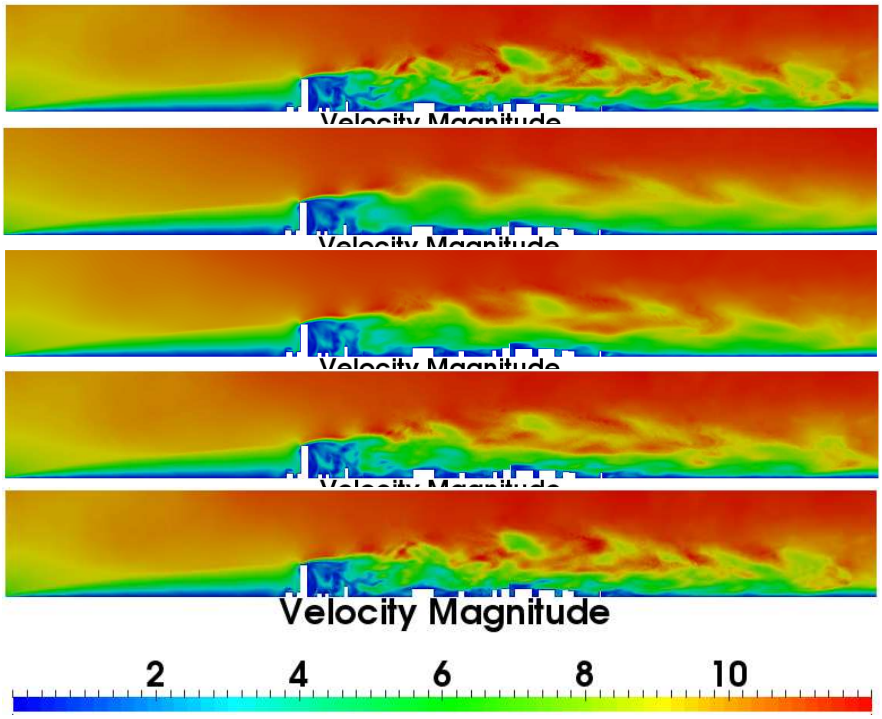
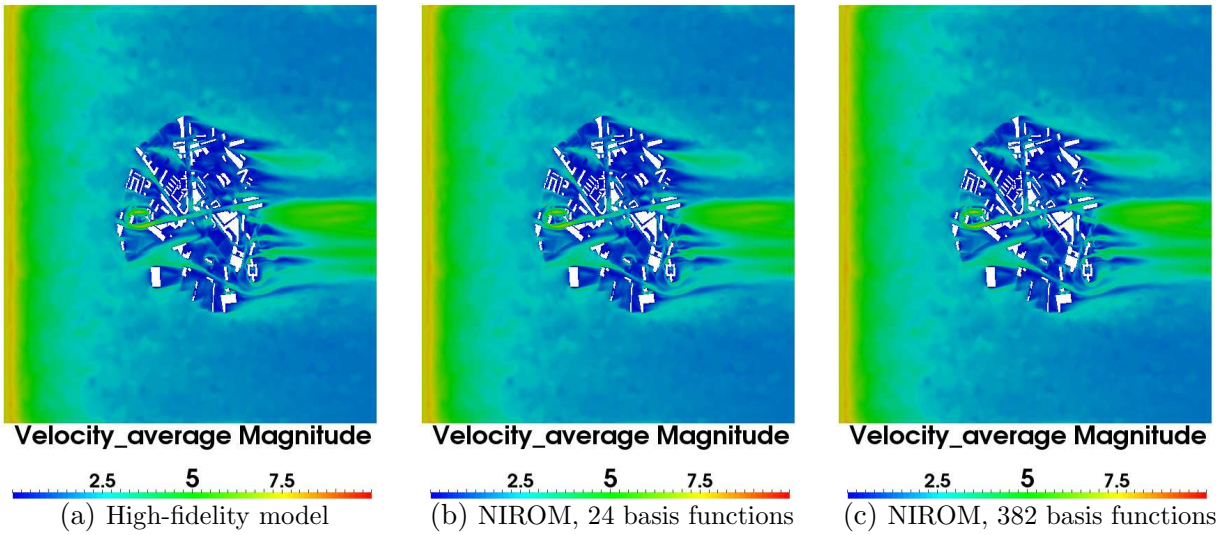


Figure 12: The plots displayed above show the magnitude of the velocity field on a plane through the centre-line of the tallest building and parallel to the streamwise direction. Results from the high-fidelity model, and from NIROM with 24, 96, 192 and 382 basis functions (from top to bottom) are shown at time 26.6 min .



(a) High-fidelity model (b) NIROM, 24 basis functions (c) NIROM, 382 basis functions

Figure 13: The plots show the magnitude of the time-averaged velocity field on a horizontal plane, at 15 m above the ground, for the high-fidelity model over a time period of $[0, 2000]$ seconds and for NIROM with 24 and 382 POD basis functions over the time period $[2000, 4000]$ seconds.

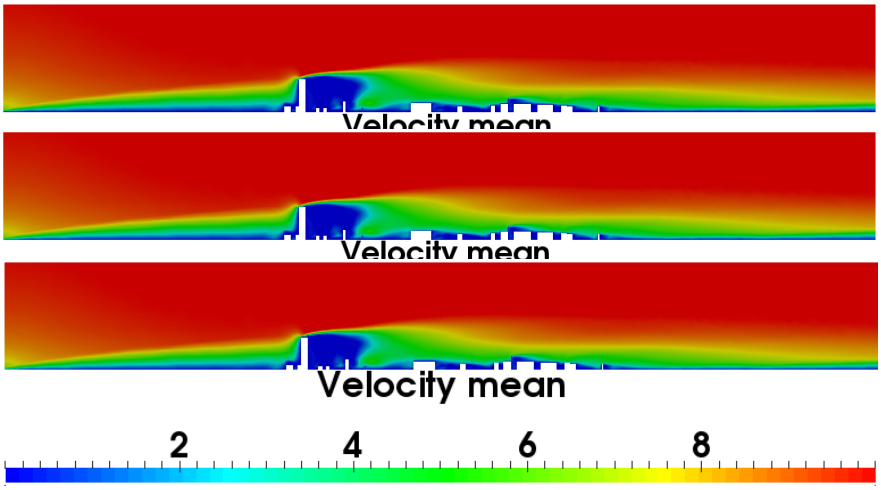


Figure 14: The plots show the magnitude of the time-averaged velocity fields on a vertical plane through the centre-line of the tallest building and parallel to the streamwise direction. The plots show the high-fidelity model averaged over time period $[0, 2000]$ seconds and NIROM averaged over the time period $[2000, 4000]$ seconds with 24 and 382 POD basis functions (top to bottom).

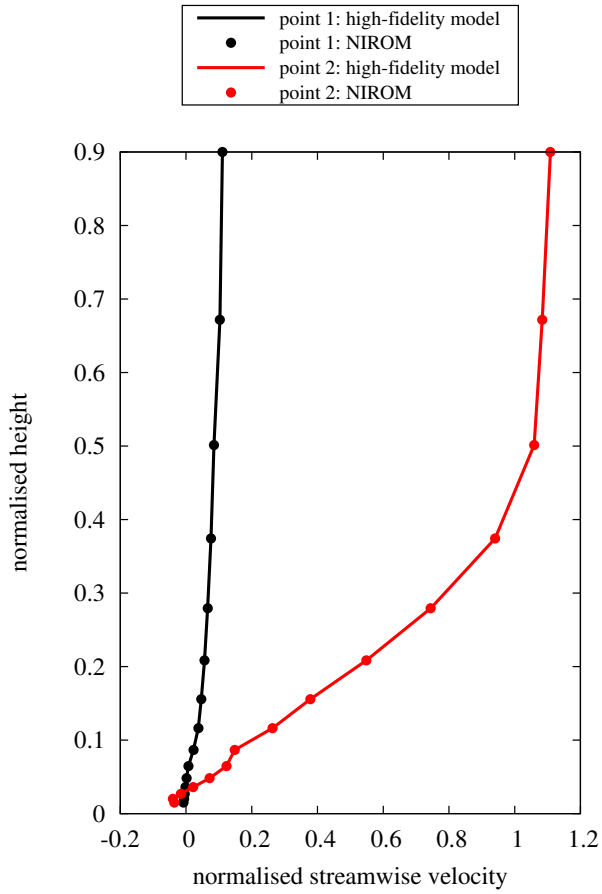


Figure 15: Streamwise mean velocity profiles are shown from the high-fidelity model and NIROM (with 382 basis functions) for a westerly wind at two points (see figure 4). The results have been normalised. The high-fidelity model is shown by solid lines and the NIROM is represented by filled circles.

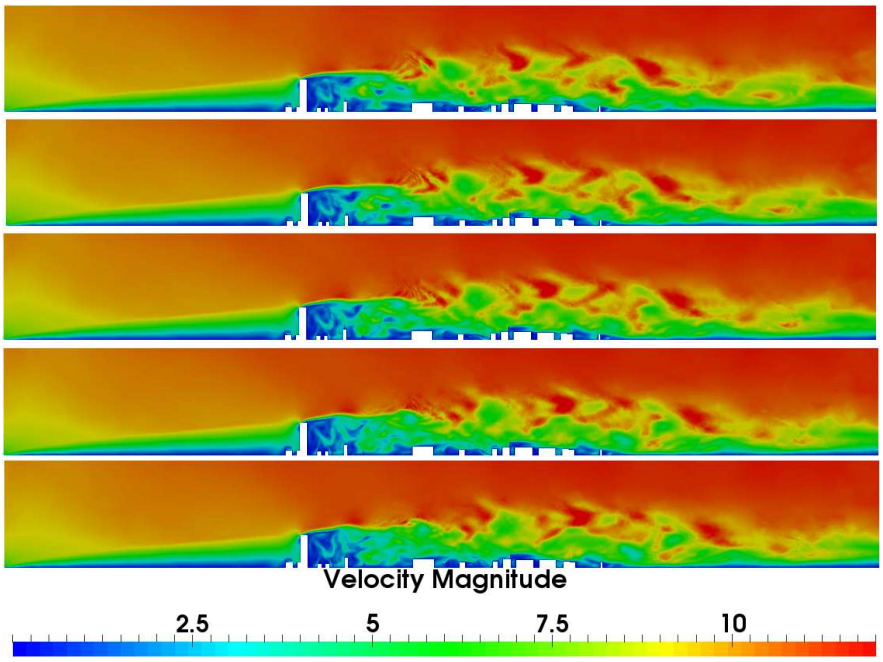


Figure 16: The plots displayed above show the magnitude of the instantaneous velocity field of the high-fidelity model on a vertical plane through the centre-line of the tallest building and parallel to the streamwise direction, obtained from the high-fidelity model at times $t = 400\,\text{s}, 404\,\text{s}, 408\,\text{s}, 412\,\text{s}, 416\,\text{s}$.

and the NIROM over different time intervals, supports our claim that the NIROM runs as a model in its own right, with its own dynamics, but, that, over time, can have similar statistics as the high-fidelity model.

We perform a time series analysis near the large junction in the centre of the domain at two observation points, see figure 18. Here flow prediction is important as this is where we expect a high density of people). The time series of the x-component of the velocity at these two points are shown in Figure 19, for the high-fidelity model over the time interval $[0, 2000]$ seconds and for the 'predictive' NIROM over $[2000, 4000]$ seconds with 24 and 382 basis functions. As shown in Figure 19, the time series of the high-fidelity model and the NIROM are obviously different, due to the chaotic nature of turbulent flows, however some characteristics are similar, such as the frequency and the magnitude of peaks. NIROM with 382 basis functions matches the time series data of the high-fidelity model much better than the NIROM with just 24 basis functions. Qualitatively, the NIROM with 382 basis functions and high-fidelity model have a similar frequency of peaks which have similar magnitudes. The time series show that the magnitude of the oscillations in the x-component of velocity increases with height. This is because the speed of the flow also increases with height, as, the further away from the ground, the less shelter there is from the buildings. Closer to ground level the velocity oscillates about zero, whereas, higher up, the velocity oscillates about a positive value.

Probability density functions (PDFs) of the x-component of the velocity from the high-fidelity model and the predicting NIROM (i.e. for NIROM over the time period $[2000, 4000]$) are given

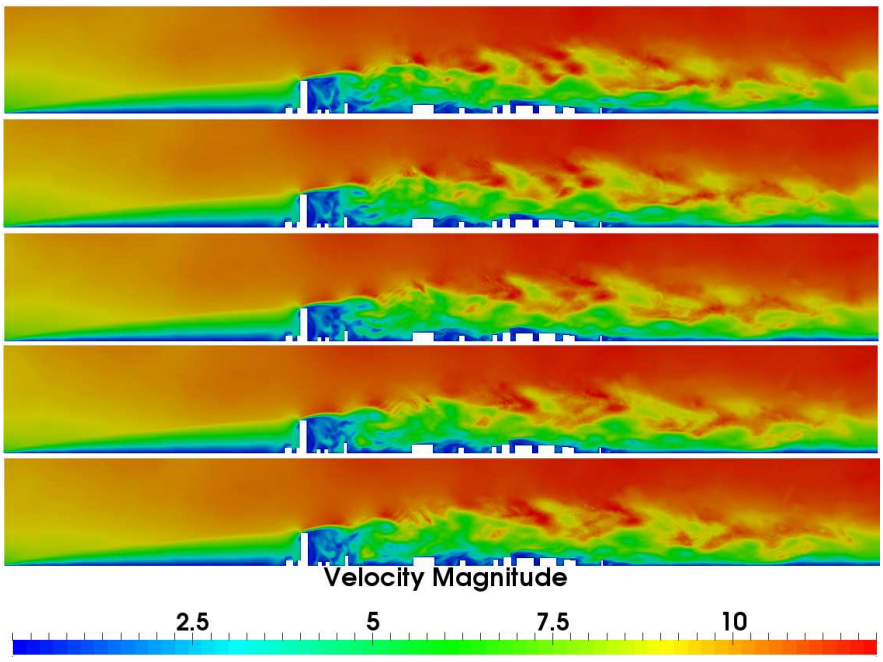


Figure 17: The figures displayed above show the magnitude of the instantaneous velocity field on a vertical plane, through the tallest building parallel to the streamwise direction, obtained from the NIROM with 382 basis functions at times $t = 2200\,\text{s}, 2204\,\text{s}, 2208\,\text{s}, 2212\,\text{s}, 2216\,\text{s}$.

in Figure 20 for all the four observation points shown in figure 18. The histograms in Figure 20 show the number of times that the velocity component falls within a certain interval. As shown in Figure 20, the distribution of the velocities for NIROM with 382 basis functions is similar to that of the high-fidelity model at all four points. The PDFs show that the fewer basis functions that are used in NIROM, the narrower (and less fluctuating) the velocity component is. Although the NIROM with 24 basis functions represents the mean flow well (see Figures 13 and 14), the PDFs reveal that it will struggle to represent the higher frequency fluctuations.

The power spectra at the lowest and highest observation points are shown in Figure 21, which highlight, as expected, that the NIROM with more basis functions matches the high-fidelity model better. The NIROM with 382 basis functions does not quite represent the high frequency components of these velocities as seen by the fact that the magnitude of the power spectra graphs for the NIROM is lower than for the high-fidelity model at the higher frequencies. This suggests that the NIROM, even with 382 POD basis functions, is slightly under resolved. With a small number of basis functions, the NIROM is not able to capture the high frequency components. The x-component of the velocity is smaller at the lower level (the yellow point), see Figure 20, due to the general trend of higher velocities occurring further away from the ground. This results in more highly peaked velocity PDFs (centred on zero). It can also be seen that the higher frequencies have lower amplitudes closer to the ground (Figure 21). Strong peaks in the power spectra at both points can be seen at a frequency of approximately $1/70\,\text{Hz}$ (the dashed line shown on the plots), which ties in with the times series plots, where periodicity over $70\,\text{s}$ is in evidence.

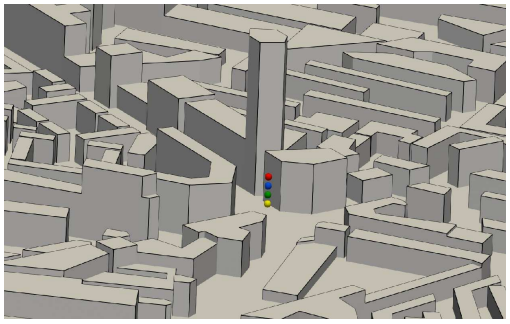


Figure 18: The plots displayed above show locations of four points (red, blue, green, yellow) at coordinates of $(4.660, 11.301, 25.9)$, $(4.660, 11.301, 20.9)$, $(4.660, 11.301, 15.9)$ and $(4.660, 11.301, 10.9)$ respectively.

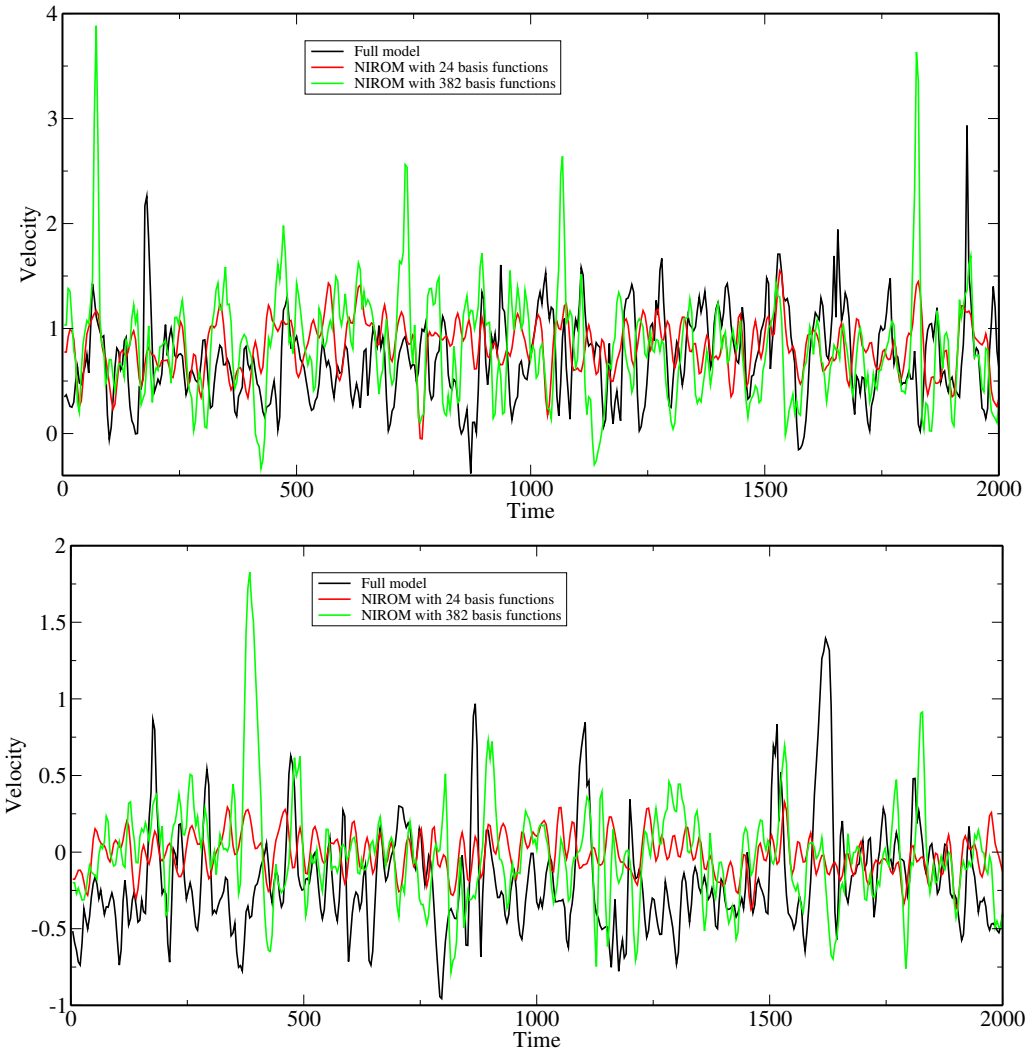
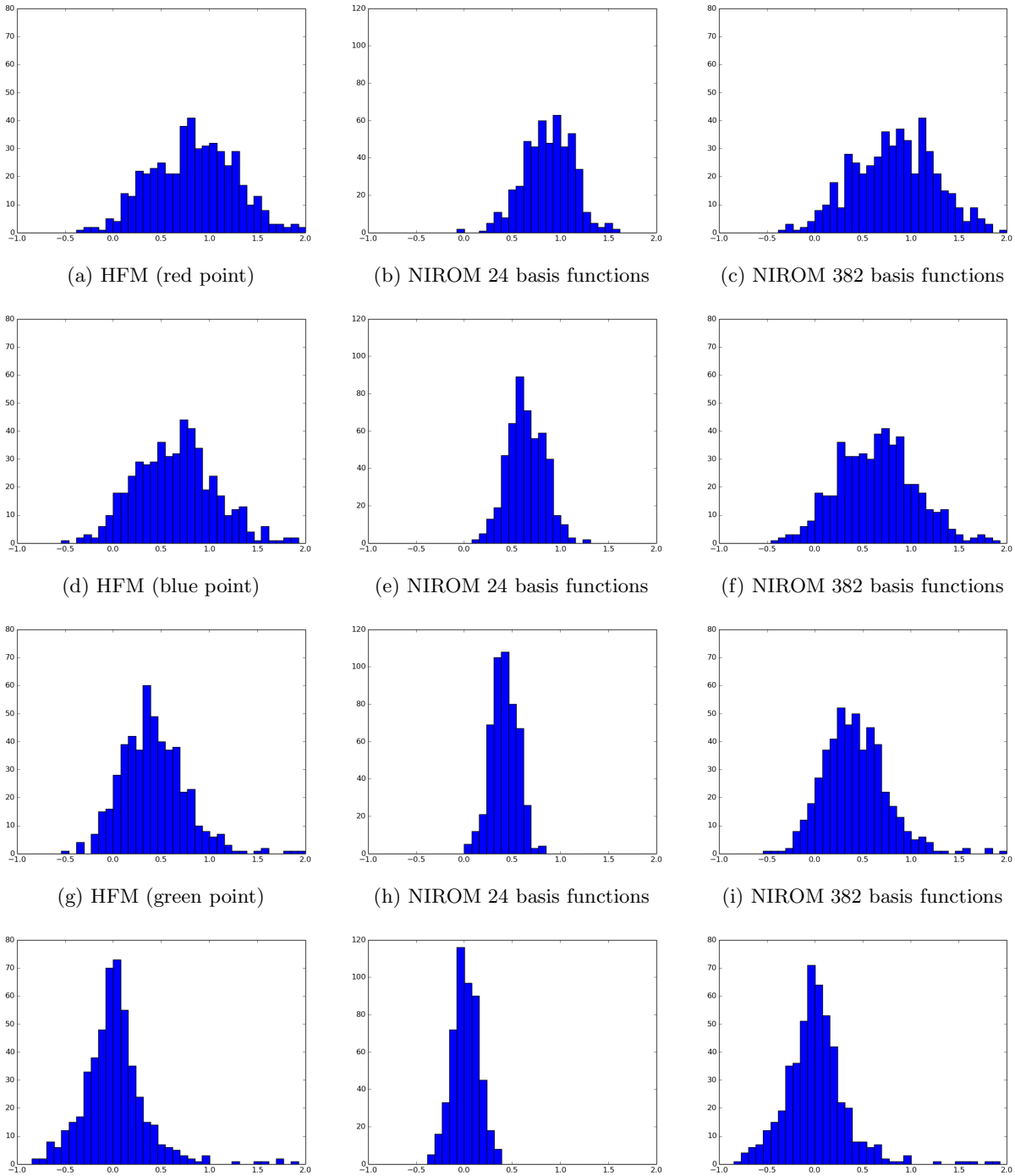


Figure 19: The plots displayed above show the time series of the x-component of the velocity from the high-fidelity model and from NIROM (in the predicting time interval) with 24 and 382 basis functions. The upper plot is the time series taken at the highest point shown in Figure 18 (the red point), the lower plot was taken at the lowest point (the yellow point).



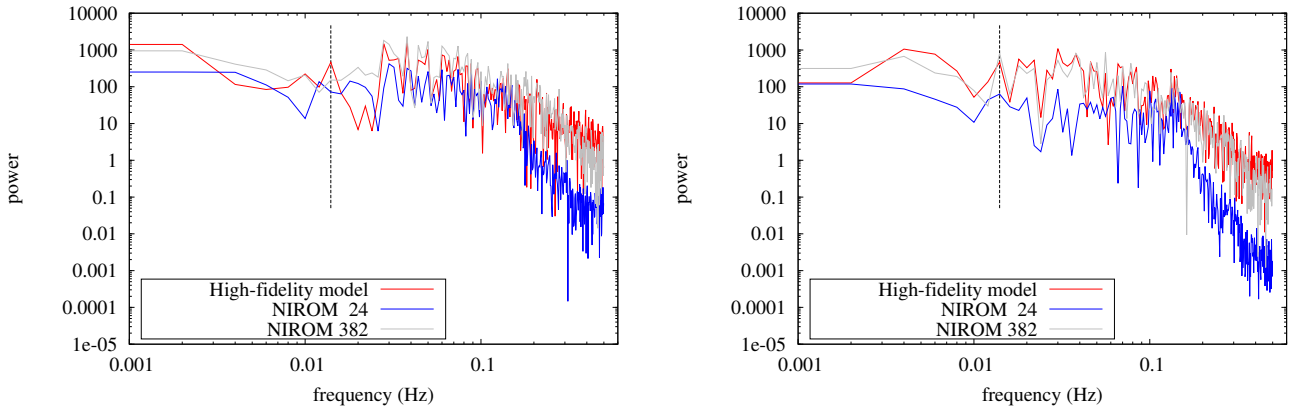
(a) HFM (red point) (b) NIROM 24 basis functions (c) NIROM 382 basis functions

(d) HFM (blue point) (e) NIROM 24 basis functions (f) NIROM 382 basis functions

(g) HFM (green point) (h) NIROM 24 basis functions (i) NIROM 382 basis functions

(j) HFM (yellow point) (k) NIROM 24 basis functions (l) NIROM 382 basis functions

Figure 20: The plots displayed above show the probability density function of the x-component of the velocity at the four points shown in Figure 18 (red, blue, green and yellow) from the high-fidelity model (HFM) and the predicting NIROM solutions with 24 and 382 basis functions.



(a) red point (highest) (b) yellow point (lowest)

Figure 21: The plots displayed above show the power spectra of the time series for the highest and lowest points shown in Figure 18 (the red and yellow points). The dashed line corresponds to a frequency of 0.014 Hz.

Figures 22 and 23 show the xx-component of the Reynolds stress ($\overline{u'u'}$) from the high-fidelity model and the NIROM solutions with 24, 96, 192 and 382 basis functions. The LES statistics are obtained from the high-fidelity model for $t \in [0, 2000]$ seconds and from the NIROM for $t \in [2000, 4000]$ seconds in order to study the 'predictive' NIROM. The xx-component of the Reynolds stress is defined as

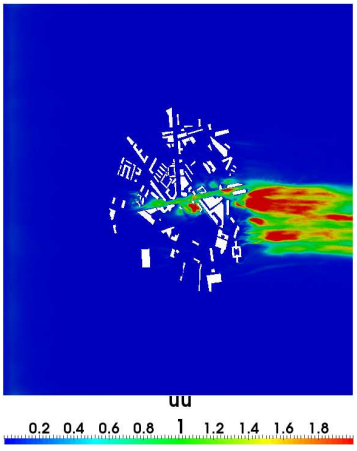
$$\overline{u'u'} = \frac{1}{t_2 - t_1} \int_{t_1}^{t_2} (u - \overline{u})^2 dt \,, \tag{18}$$

for a time interval of $[t_1, t_2]$, u is the x-component of pointwise velocity and the over-bar denotes the time average. Notice that the magnitude of the fluctuations (as characterised by $\overline{u'u'}$) is greatly reduced when fewer basis functions are used. This is also consistent with the PDFs of the x-component of velocity at the 4 points of interest shown in Figure 20. We also observe that the Reynolds stresses are captured very well by the NIROM with 382 basis functions as they closely approach the Reynolds stresses of the high-fidelity model.

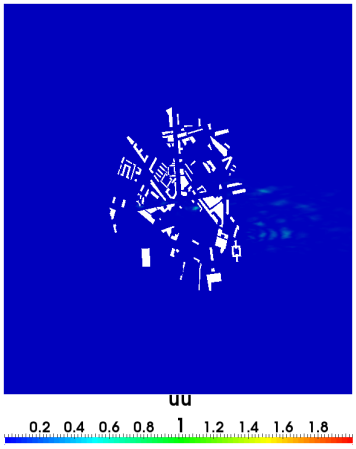
Table 1 shows the average CPU cost required by both the high-fidelity model and NIROM (with 382 basis functions) to solve for 4 seconds in real time (which corresponds to three time steps for the high-fidelity model and one time step for the NIROM). It is worth noting that the CPU time required to solve the NIROM over this time interval is only 0.004 s, whereas the high-fidelity model requires 1555 s running in parallel on 10 cores of a workstation with Intel(R) Xeon(R) X5680 CPU processors of 3.3 GHz and 512 GB RAM. The total amount of time required to generate the snapshots for this example was 9 days.

5. Discussion

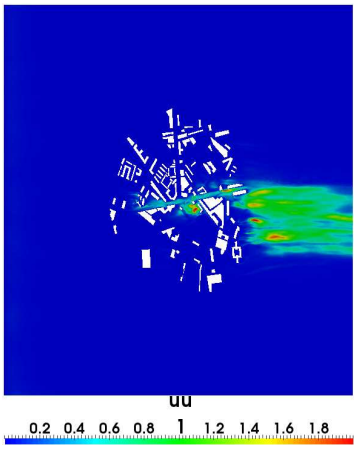
The results show that NIROM (with 382 POD basis functions) is able to represent, accurately, most of the dynamics seen in the high-fidelity solutions of the urban flows. We now comment on how the accuracy of the NIROM dynamics is related to the rate of decrease of the singular values. It was observed that all the NIROMs reproduced the mean flows well, even those using as few as



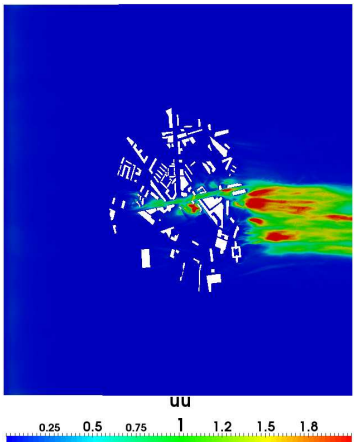
(a) High-fidelity model



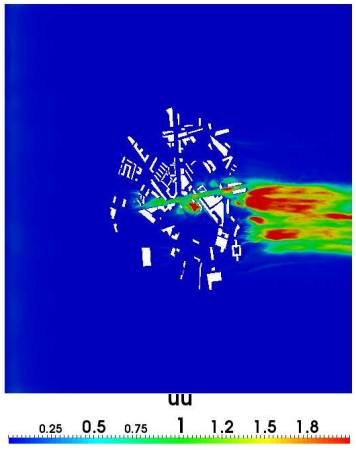
(b) NIROM, 24 basis functions



(c) NIROM, 96 basis functions



(d) NIROM, 192 basis functions



(e) NIROM, 382 basis functions

Figure 22: The plots displayed above show the Reynolds stresses from the high-fidelity model and NIROM (predicting) with 24, 96, 192 and 382 basis functions. These are shown on a horizontal plane at a height of 15 m above ground level.

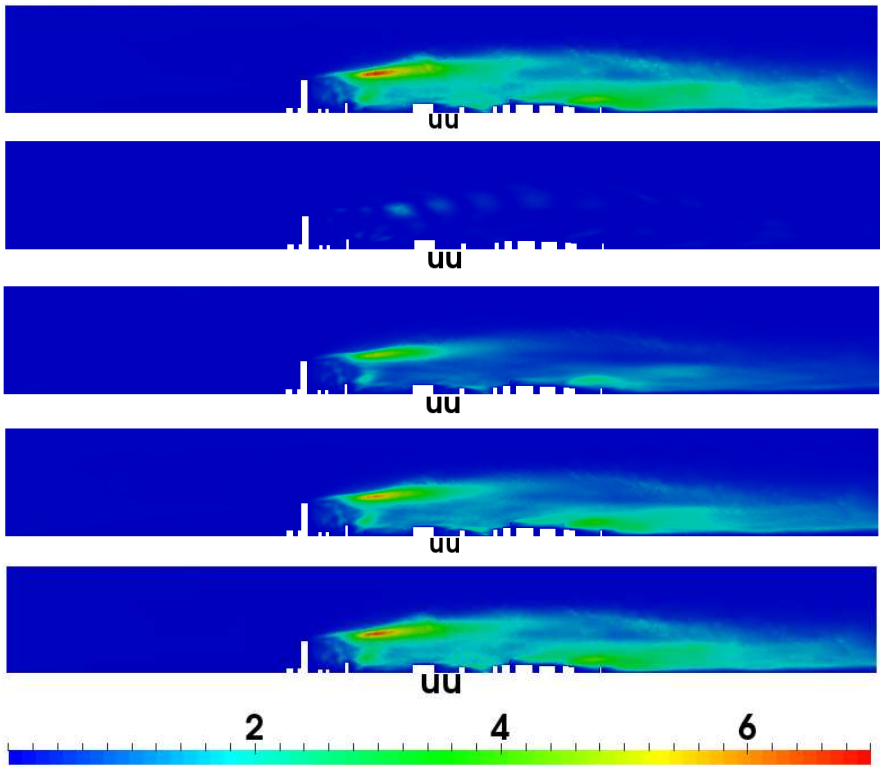


Figure 23: The plots displayed above show the Reynolds stresses of the high-fidelity model and NIROM (predicting) with 24, 96, 192 and 382 basis functions. These results are shown on a vertical plane through the centre-line of the tallest building and parallel to the streamwise direction.

Table 1: Comparison of the CPU time (in seconds) required to solve the high-fidelity model and NIROM (with 382 basis functions) in 4 s of real time. For the NIROM, 'solution' in the left column corresponds to step (a) in Algorithm 1 and 'projection' corresponds to step (b).

	high-fidelity model	NIROM
assembly	1555	n/a
solution		0.001
projection	n/a	0.003
total	1555	0.004

24 basis functions. However, the Reynolds stresses, which express the fluctuations from the mean velocities, are not represented well, unless one is willing to use larger numbers of basis functions. In this case we found that, with 382 basis functions, one can capture the Reynolds stresses very well and still have a rapid model that is six orders of magnitude times faster than the high-fidelity model. This is supported by the probability distribution plots of the velocities, which are much narrower when fewer basis functions are used. This may be due to the fact that the lower order modes are simply unable to represent the small scale fluctuations and are only able to represent the fluctuations in the larger scale motions. Using 382 basis functions produces PDFs that are very similar to those of the high-fidelity model. Also, the power spectra of the time series show, again, that the higher fidelity NIROMs are able to capture the higher frequencies whereas the lower fidelity NIROMs cannot, due to their inadequate spatial resolution. All these findings are a consequence of the slow rate of decrease of the singular values seen after the 8th POD basis function index in Figure 8.

Parametrising the NIROM will be the focus of our next research effort, which will have as its objective to solve for air flows within a neighbourhood, with varying wind directions, varying statistics of the urban boundary layer, and varying traffic and temperature conditions.

6. Conclusions

We have developed a non-intrusive reduced order model (NIROM) for modelling flows in the urban environment using machine learning. The proposed NIROM was constructed from a high-fidelity model which uses a Large Eddy Simulation approach to capture larger scale, turbulent fluctuations implemented in a code called Fluidity. We have presented a validation of Fluidity against data collected from experiments conducted at the Enflo Meteorological Wind Tunnel in the University of Surrey. The NIROM was able to replicate accurately the data used in its training, but also was able to predict beyond the training data, displaying flow statistics that were close to those of the high-fidelity model. Furthermore, the NIROM ran six orders of magnitude times faster than the high-fidelity model.

When forming the basis functions by performing a singular value decomposition, we found it best to use information from all velocity components simultaneously, rather than separating the components and deriving basis functions for each component. The improvement found in this case

might due to the fact that we retain aspects of the differential equation such as correlations between the velocity components. This, together with the use of a Gaussian Regression Process machine learning method, has contributed to the ability of the NIROM to reproduce the results of the high-fidelity model and to predict into the future, as a model in its own right.

Acknowledgments

The 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 grants GR/R60898, EP/I00405X/1, EP/J002011/1, EP/N010221/1 (Managing Air for Green Inner Cities - MAGIC),the Innovate UK Smart-GeoWells consortium and EP/P033148/1 (MUFFINS) and the EPSRC MEMPHIS multi-phase flow programme grant (EP/K003976/1); and the Imperial College High Performance Computing Service. The authors are grateful for the funding from the European Union Seventh Frame work Programme (FP7/20072013) under grant agreement No.603663 for the research project PEARL (Preparing for Extreme And Rare events in coastal regions). Dr. Xiao acknowledges the support of NSFC grant 11502241. Prof. I.M. Navon acknowledges the support of NSF/CMG grant ATM-0931198.

References

[1] Z. Wang, D. Xiao, F. Fang, R. Govindan, C.C. Pain, and Y. Guo. Model identification of reduced order fluid dynamics systems using deep learning. *International Journal for Numerical Methods in Fluids*, 86(4):255–268, 2018.

[2] Fernando F Grinstein. On integrating large eddy simulation and laboratory turbulent flow experiments. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 367(1899):2931–2945, 2009.

[3] Bert Blocken. Computational fluid dynamics for urban physics: Importance, scales, possibilities, limitations and ten tips and tricks towards accurate and reliable simulations. *Building and Environment*, 91:219–245, 2015.

[4] Marcel Vonlanthen, Jonas Allegrini, and Jan Carmeliet. Urban climate multiscale interaction between a cluster of buildings and the abl developing over a real terrain. *Urban Climate*, 20:1–19, 2017.

[5] M. Lateb, Robert N. Meroney, M. Yataghene, H. Fellouah, F. Saleh, and M. C. Boufadel. On the use of numerical modelling for near-field pollutant dispersion in urban environments - a review. *Environmental Pollution*, 208:271–283, 2016.

[6] E. Aristodemou, L.M. Boganegra, L. Mottet, D. Pavlidis, A. Constantinou, C.C. Pain, A. Robins, and H. ApSimon. How tall buildings affect turbulent air flows and dispersion of pollution within a neighbourhood. *Environmental Pollution*, 233:782–796, 2018.

[7] J. Song, S. Fan, W. Lin, L. Mottet, H. Woodward, M. Davies Wykes, R. Arcucci, D. Xiao, J.-E. Debay, H. ApSimon, E. Aristodemou, D. Birch, M. Carpentieri, F. Fang, M. Herzog, G.R. Hunt, R.L. Jones, C.C. Pain, D. Pavlidis, A.G. Robins, C.A. Short, and P. F. Linden. Natural ventilation in cities: the implications of fluid mechanics. *Building Research & Information*, 0(0):1–20, 2018.

[8] S. Omrani, V. Garcia-Hansen, B. Capra, and R. Drogemuller. Natural ventilation in multi-storey buildings: Design process and review of evaluation tool. *Building and Environment*, 116:182–194, 2017.

[9] W. Guo, X. Liu, and X. Yuan. Study on natural ventilation design optimization based on cfd simulation for green buildings. In *9th International Symposium on Heating, Ventilation and Air Conditioning (ISHVAC) and the 3rd International Conference on Building Energy and Environment (COBEE), Procedia Engineering*, volume 121, pages 573–581, 2015.

[10] Xiaomin Xie, Zhen Huang, and Jia Song Wang. Impact of building configuration on air quality in street canyon. *Atmospheric Environment*, 39:4519–4530, 2005.

[11] Bert Blocken. 50 years of Computational Wind Engineering: Past, present and future. *Journal of Wind Engineering and Industrial Aerodynamics*, 129:69–102, 2014.

[12] S. M. Salim, K. C. Ong, and S. C. Cheah. Comparison of RANS, URANS and LES in the Prediction of Airflow and Pollutant Dispersion. In *Proceedings of the World Congress on Engineering and Computer Science*, 2011.

[13] P Gousseau, Bert Blocken, Theodore Stathopoulos, and G Van Heijst. CFD simulation of near-field pollutant dispersion on a high-resolution grid: A case study be LES and RANS for a building group in downtown Montreal. *Atmospheric Environment*, 45:428–438, 2011.

[14] R. Ramponi and B. Blocken. CFD simulation of cross-ventilation for a generic isolated building: Impact of computational parameters. *Building and Environment*, 53:34–48, 2012.

[15] N. Koutsourakis, J.G. Bartzis, and N.C. Markatos. Evaluation of Reynolds stress, k-ε and RNG k-ε turbulence models in street canyon flows using various experimental datasets. *Environmental fluid mechanics*, 12(4):379–403, 2012.

[16] T. Van Hooff, B. Blocken, and Y. Tominaga. On the accuracy of CFD simulations of cross-ventilation flows for a generic isolated building: Comparison of RANS, LES and experiments. *Building and Environment*, 114:148–165, 2017.

[17] S-J Cao and Johan Meyers. On the construction and use of linear low-dimensional ventilation models. *Indoor Air*, 22(5):427–441, 2012.

[18] Wei Liu, Mingang Jin, Chun Chen, Ruoyu You, and Qingyan Chen. Implementation of a fast fluid dynamics model in OpenFOAM for simulating indoor airflow. *Numerical Heat Transfer, Part A: Applications*, 69(7):748–762, 2016.

[19] A Tallet, C Allery, and F Allard. POD approach to determine in real-time the temperature distribution in a cavity. *Building and Environment*, 93:34–49, 2015.

[20] 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*, 293:522–541, 2015.

[21] R. 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.

[22] R. Stefanescu, A. 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.

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

[24] 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.

[25] D. Xiao, F. Fang, C.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.

[26] F. Fang, T. Zhang, D. Pavlidis, C.C. Pain, A.G. 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.

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

[28] A. Manzoni, F. Salmoiraghi, and L. 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.

[29] 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.

[30] 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*, 2015.

[31] F. Fang, C.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.

[32] F. Sabetghadam and A. Jafarpour. α regularization of the POD-Galerkin dynamical systems of the Kuramoto–Sivashinsky equation. *Applied Mathematics and Computation*, 218(10):6012–6026, 2012.

[33] K. Carlberg, C. Bou-Mosleh, 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(2):155–181, 2011.

[34] Y. Chu, M. Serpas, and J. Hahn. State-preserving nonlinear model reduction procedure. *Chemical engineering science*, 66(17):3907–3913, 2011.

[35] K. Willcox and A. Megretski. Model reduction for large-scale linear applications. *Proc.of 13th IFAC Symposium on System Identification, Rotterdam, Netherlands*, 1431–1436, 2003.

[36] 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. *Comptes Rendus Mathematique*, 339(9):667–672, 2004.

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

[38] K. Carlberg, C. Farhat, J. Cortial, and D. 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.

[39] F. Fang, C.C. Pain, I.M. Navon, M.D. Piggott, G.J. Gorman, P.A. Allison, and A.J.H. Goddard. Reduced-order modelling of an adaptive mesh ocean model. *International journal for numerical methods in fluids*, 59(8):827–851, 2009.

[40] 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(3):362–379, 2013.

[41] 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.

[42] Peng Wu, Zhuangbo Feng, and Shi-jie Cao. Fast and accurate prediction of airflow and drag force for duct ventilation using wall-modeled large-eddy simulation. *Building and Environment*, 2018.

[43] Lieven Vervecken, Johan Camps, and Johan Meyers. Stable reduced-order models for pollutant dispersion in the built environment. *Building and Environment*, 92:360–367, 2015.

[44] Shi-Jie Cao and Johan Meyers. Fast prediction of indoor pollutant dispersion based on reduced-order ventilation models. In *Building Simulation*, volume 8, pages 415–420. Springer, 2015.

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

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

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

[48] C. Audouze, F.D. Vuyst, and P.B. Nair. Nonintrusive reduced-order modeling of parametrized time-dependent partial differential equations. *Numerical Methods for Partial Differential Equations*, 29(5):1587–1628, 2013.

[49] 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.

[50] 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.

[51] B.R. Noack, M. Morzynski, and G. Tadmor. *Reduced-Order Modelling for Flow Control*, volume 528. Springer, 2011.

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

[53] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet Classification with Deep Convolutional Neural Networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[54] M. Wieland and M. Pittore. Performance evaluation of machine learning algorithms for urban pattern recognition from multi-spectral satellite images. *Remote Sensing*, 6(4):2912–2939, 2014.

[55] M.M. Waldrop. No drivers required. *Nature*, 518:20–23, 2015.

[56] L. Liu, H. Wang, and C. Wu. A machine learning method for the large-scale evaluation of urban visual environment. *arXiv preprint arXiv:1608.03396*, 2016.

[57] J. Kim, Y. Zhou, S. Schiavon, P. Raftery, and G. Brager. Personal comfort models: Predicting individuals' thermal preference using occupant heating and cooling behavior and machine learning. *Building and Environment*, 129:96–106, 2018.

[58] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97, 2012.

[59] S. Jean, K. Cho, R. Memisevic, and Y. Bengio. On using very large target vocabulary for neural machine translation. *arXiv preprint arXiv:1412.2007*, 2014.

[60] T. Duriez, S.L. Brunton, and B.R. Noack. *Machine Learning Control - Taming Nonlinear Dynamics and Turbulence*, volume 123. Springer, 2017.

[61] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D.G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: A system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283, 2016.

[62] F. Chollet et al. Keras. https://keras.io, 2015.

[63] D. Xiao, Z. Lin, F. Fang, C.C. Pain, I.M. Navon, P. Salinas, and A. Muggeridge. Non-intrusive reduced-order modeling for multiphase porous media flows using Smolyak sparse grids. *International Journal for Numerical Methods in Fluids*, 83(2):205–219, 2017.

[64] C.E. Rasmussen. Gaussian Processes in Machine Learning. In *Advanced lectures on machine learning*, pages 63–71. Springer, 2004.

[65] Y. Wang, H. Huang, L. Huang, and B. Ristic. Evaluation of Bayesian source estimation methods with Prairie Grass observations and Gaussian plume model: A comparison of likelihood functions and distance measures. *Atmospheric environment*, 152:519–530, 2017.

[66] E. Aristodemou, T. Bentham, C.C. Pain, R. Colvile, A. Robins, and H. ApSimon. A comparison of mesh-adaptive LES with wind tunnel data for flow past buildings: Mean flows and velocity fluctuations. *Atmospheric Environment*, 43(39):6238–6253, 2009.

[67] D. Xiao, P. Yang, F. Fang, J. Xiang, C.C. Pain, I.M. 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.

[68] C.E. Rasmussen and C.K.I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

[69] L-S Yao. Is a direct numerical simulation of chaos possible? A study of a model nonlinearity. *International Journal of Heat and Mass Transfer*, 50:2200–2207, 2007.

[70] C.C. Pain, M.D. Piggott, A.J.H. Goddard, F. Fang, G.J. Gorman, D.P. Marshall, M.D. Eaton, P.W. Power, and C.R.E. De Oliveira. Three-dimensional unstructured mesh ocean modelling. *Ocean Modelling*, 10(1-2):5–33, 2005.

[71] C.C. Pain, A.P. Umpleby, C.R.E. De Oliveira, and A.J.H. Goddard. Tetrahedral mesh optimisation and adaptivity for steady-state and transient finite element calculations. *Computer Methods in Applied Mechanics and Engineering*, 190(29):3771–3796, 2001.

[72] J.H.T. Bentham. *Microscale modelling of air flow and pollutant dispersion in the urban environment*. PhD thesis, Imperial College London, 2004.

[73] J. Franke, A. Hellsten, H. Schlünzen, and B. Carissimo. *Best practice guideline for the CFD simulation of flows in the urban environment*. Meteorological Institute, 2007.

[74] D. Pavlidis, G. Gorman, J. Gomes, C.C. Pain, and H. ApSimon. Synthetic-eddy method for urban atmospheric flow modelling. *Boundary-Layer Meteorology*, 136:285–299, 2010.