

A non-intrusive reduced order model using deep learning for realistic wind data generation for small unmanned aerial systems in urban spaces

Cite as: AIP Advances **12**, 085020 (2022); <https://doi.org/10.1063/5.0098835>

Submitted: 12 May 2022 · Accepted: 25 July 2022 · Published Online: 19 August 2022

 Rohit K. S. S. Vuppala and  Kursat Kara



View Online



Export Citation



CrossMark

ARTICLES YOU MAY BE INTERESTED IN

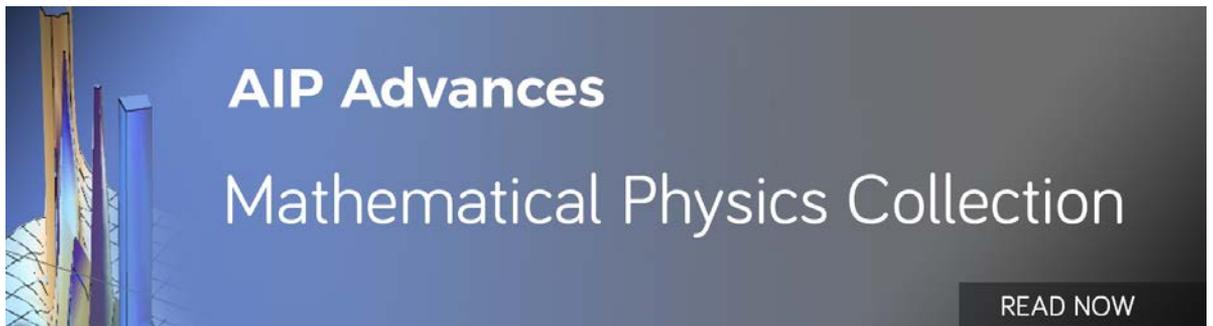
Physics-informed neural networks for solving Reynolds-averaged Navier–Stokes equations
Physics of Fluids **34**, 075117 (2022); <https://doi.org/10.1063/5.0095270>

Functional mobile-based two-factor authentication by photonic physical unclonable functions

AIP Advances **12**, 085316 (2022); <https://doi.org/10.1063/5.0101483>

Singly split single ring resonator: Fitting of lumped-element circuit model parameters and some aspects of resonator analysis and design

AIP Advances **12**, 085213 (2022); <https://doi.org/10.1063/5.0097371>



AIP Advances
Mathematical Physics Collection

READ NOW

A non-intrusive reduced order model using deep learning for realistic wind data generation for small unmanned aerial systems in urban spaces

Cite as: AIP Advances 12, 085020 (2022); doi: 10.1063/5.0098835

Submitted: 12 May 2022 • Accepted: 25 July 2022 •

Published Online: 19 August 2022



View Online



Export Citation



CrossMark

Rohit K. S. S. Vuppala  and Kursat Kara^{a)} 

AFFILIATIONS

School of Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, Oklahoma 74078, USA

^{a)} Author to whom correspondence should be addressed: kursat.kara@okstate.edu

ABSTRACT

Realistic wind data are essential in developing, testing, and ensuring the safety of unmanned aerial systems in operation. Alternatives to Dryden and von Kármán turbulence models are required, aimed explicitly at urban air spaces to generate turbulent wind data. We present a novel method to generate realistic wind data for the safe operation of small unmanned aerial vehicles in urban spaces. We propose a non-intrusive reduced order modeling approach to replicate realistic wind data and predict wind fields. The method uses a well-established large-eddy simulation model, the parallelized large eddy simulation model, to generate high-fidelity data. To create a reduced-order model, we utilize proper orthogonal decomposition to extract modes from the three-dimensional space and use specialized recurrent neural networks and long-term short memory for stepping in time. This paper combines the traditional approach of using computational fluid dynamic simulations to generate wind data with deep learning and reduced-order modeling techniques to devise a methodology for a non-intrusive data-based model for wind field prediction. A simplistic model of an isolated urban subspace with a single building setup in neutral atmospheric conditions is considered a test case for the demonstration of the method.

© 2022 Author(s). All article content, except where otherwise noted, is licensed under a Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>). <https://doi.org/10.1063/5.0098835>

I. INTRODUCTION

Unmanned Aerial Systems (UAS) are an integral part of Urban Air Mobility (UAM) to safely and efficiently carry out scheduled and on-demand, short, point-to-point transportation services in low-altitude urban or metropolitan environments. It is also a significant part of Advanced Air Mobility (AAM), an emerging concept for urban, rural, regional, and inter-regional transportation using revolutionary new aircraft. In recent years, there has been significant interest from industry and government agencies in using aviation technologies like Unmanned Aerial Systems (UAS) to revolutionize our transportation capabilities. With their diverse applications like law enforcement, general surveillance, aerial structure inspection, disaster management,¹ urban mapping,² and also door-to-door delivery and catering services,³

unmanned aerial systems have proved to be a worthy contender. As the applications and requirements for these unmanned systems grew, so did the complexity around them, necessitating the need to augment with ground control stations (GCS), command and communication links, on-demand trajectory planners, and other auxiliary subsystems.^{4,5} Due to such complexity, exposure, and use in myriad applications, these systems have now reached maturity and technological know-how for implementing urban air mobility for next-generation state-of-the-art transportation capabilities in cities.

For efficient use of available airspace in dense urban environments, significant weight and size restrictions have to be imposed on the UAS, persuading the use of small Unmanned Aerial Systems (sUAS). However, sUAS, due to their miniaturized physical dimensions, are low-weight and are more susceptible to wind disturbances

such as atmospheric turbulence and gusts. In the past decade, there has been a significant focus in using sUAS for urban applications^{6,7} and mitigating urban field effects, such as heat islands.^{8,9} While recently, researchers have investigated the influence of urban effects on flight planning and trajectories^{10,11} to develop various control strategies to minimize the effects, they face a significant challenge for *in situ* development and testing their algorithms in the absence of realistic wind data. Furthermore, with the advent of augmented reality and virtual reality technologies, realistic wind data inclusion is a key enabler in modeling simulation environments for UAS pilot training.¹²

Turbulence models, such as Dryden¹³ and von Kármán,¹³ are often used to generate gusty wind fields with predefined parameters in large unobstructed domains. However, they may not be suitable for flow simulations of urban spaces since urban structures, such as buildings, create flow obstruction for the smaller computational domains. While Reynolds-Averaged Navier–Stokes (RANS) solutions have been used for computing flow fields in urban spaces, they do not represent the unsteady nature of the flow required for effectively devising various control and path-planning algorithms.

On the other hand, Large Eddy Simulations (LESs) provide a more realistic depiction of the flow field^{14,15} and enable the testing and implementation of better control algorithms for UAS.¹⁶ However, the computational cost of LES limits the possibility of testing at different flow scenarios in urban spaces.¹⁷ Hence, we need a novel way of generating flow fields in urban spaces and testing for real-time or close to real-time prediction and flight correction strategies for the sUAS operation.

Data-driven, Reduced Order Models (ROMs) have been widely adopted to efficiently reproduce and simulate fluid flow using available high-fidelity data from Computational Fluid Dynamics (CFD) simulations. They have also been used in many diverse applications, such as process simulation and optimization,¹⁸ flow control,¹⁹ and fluid flows.²⁰ With the advent of machine learning and deep learning techniques, there has been wide-spread utilization of these techniques for advancement in various fields of flow modeling and its applications, such as modeling wind fields.^{21–27} In recent years, many such machine learning/deep learning based models have been successfully implemented for model order reduction in fluid flow problems.^{28,29} Using traditional techniques, such as computational fluid dynamics, researchers have been able to generate necessary data to train such models non-intrusively.^{30,31} Non-Intrusive Reduced Order Models (NIROMs) have not only been used for model reduction but also for hidden model recovery in cases where the physics of the problem is not entirely known.^{32,33} Using techniques, such as Proper Orthogonal Decomposition (POD) and Galerkin Projections (GPs) in conjunction with Machine Learning/Deep Learning (ML/DL) techniques,^{34,35} reduced-order models could be relatively efficient and accurate while only utilizing a few underlying features of the data for reconstructing the flow field. Access to these ROMs with low-computational costs could enable wind-aware *in situ* navigation and planning strategies, previously thought of as impossible.

The current work is an attempt to use a machine learning-based reduced-order modeling approach, specifically a Non-Intrusive Reduced Order Model (NIROM) using proper orthogonal decomposition and Long Short-Term Memory (LSTM) networks.³¹

We chose a simplistic domain with a single building for demonstrating its utility in urban wind field prediction. This lets us test the method as an effective alternative to generate more realistic wind data using high-fidelity LES data. Furthermore, we also attempt to predict the flow-field for all future time instances. Both of the above capabilities let researchers devise more robust, wind-aware algorithms and sub-systems for UAS operations in urban spaces.

This paper is organized as follows: Sec. II contains the methodology adopted for both generating the data and creating the Machine Learning based Reduced Order Model (ML-ROM). The large eddy simulation setup, along with the governing equations, discretization, and boundary condition treatment, is briefly discussed in Subsection II A. Some brief details about the non-intrusive ROM-LSTM method are described in Subsection II B. We present our results from the NIROM-LSTM approach in Sec. III, which has comparisons between predictions and actual data for the velocity field. We end the paper with general conclusions and remarks in Sec. IV.

II. METHODOLOGY

In this section, we discuss the methodology for our approach that uses large eddy simulation data and a non-intrusive reduced order model using LSTM networks (ROM-LSTM) to predict the flow field in a given domain of interest.

A. LES simulation setup

Large eddy simulation data are obtained using Parallelized Large-Eddy Simulation Model (PALM).³⁶ PALM is a turbulence-resolving, large eddy simulation solver for atmospheric and oceanic boundary-layer flows. The model is based on solving non-hydrostatic, filtered, incompressible Navier–Stokes equations in Boussinesq-approximated form on a Cartesian grid. Implicit separation of sub-grid scales and resolved scales is achieved by averaging the governing equations over discrete grid volumes as proposed by Schumann.³⁷

1. Governing equations

The model solves for six prognostic quantities, the velocity components u, v, w , the potential temperature θ , specific humidity q_v , and the SGS turbulent kinetic energy e using Eqs. (1)–(8). The potential temperature is defined as

$$\Theta = \frac{T}{\Pi} \quad (1)$$

from absolute temperature T and the Exner function

$$\Pi = \left(\frac{p}{p_0} \right)^{\frac{R_d}{C_p}}, \quad (2)$$

where p is the hydrostatic pressure, p_0 is the reference pressure 1000 hPa, R_d is the gas constant for dry air, and C_p is the specific heat of dry air at constant pressure. Furthermore, a virtual potential temperature could be calculated using the relation

$$\Theta_v = \Theta \left[1 + \left(\frac{R_v}{R_d} - 1 \right) q_v - q_l \right], \quad (3)$$

where R_v is the gas constant for water vapor and q_l is the liquid water-specific humidity calculated based on a chosen cloud microphysics model. (Note: for the present study dry atmospheric boundary conditions with neutral stratification are considered eliminating the need for cloud multi-physics and also making the absolute temperature, potential temperature, and virtual potential temperature the same value.) The governing equations for the conservation of mass, momentum, energy, and moisture filtered over a Cartesian grid are expressed below in Einstein summation notation, where angle brackets denote horizontal domain average, over-bar indicates filtered quantities and double-prime indicates the SGS variables

$$\begin{aligned} \frac{\partial \bar{u}_i}{\partial t} = & -\frac{\partial \bar{u}_i \bar{u}_j}{\partial x_j} - \varepsilon_{ijk} f_j \bar{u}_k + \varepsilon_{ij3} f_3 \bar{u}_{gj} - \frac{1}{\rho_0} \frac{\partial \Pi^*}{\partial x_i} \\ & + g \frac{\bar{\Theta}_v - \langle \bar{\Theta}_v \rangle}{\langle \bar{\Theta}_v \rangle} \delta_{i3} - \frac{\partial (\overline{u_i'' u_j''} - \frac{2}{3} \bar{e} \delta_{ij})}{\partial x_j}, \end{aligned} \quad (4)$$

$$\frac{\partial \bar{u}_j}{\partial x_j} = 0, \quad (5)$$

$$\frac{\partial \bar{\Theta}}{\partial t} = -\frac{\partial \bar{u}_j \bar{\Theta}}{\partial x_j} - \frac{\partial (\overline{u_j'' \Theta''})}{\partial x_j} - \frac{L_v}{C_p \bar{\Pi}} \Psi_{q_v}, \quad (6)$$

$$\frac{\partial \bar{q}_v}{\partial t} = -\frac{\partial \bar{u}_j \bar{q}_v}{\partial x_j} - \frac{\partial \overline{u_j'' q_v''}}{\partial x_j} + \Psi_{q_v}, \quad (7)$$

where $u_i (i = 1, 2, 3)$ represents the components of velocities, f_i is the Coriolis parameter, L_v is latent heat of vaporization, g is the gravitational acceleration, $u_{g,k}$ are the geostrophic wind components, ρ_0 is the density of dry air, p^* is the perturbation pressure, $\Pi^* = p^* + 2/3 \rho_0 e$ is the modified perturbation pressure, and SGS TKE is represented by e .

2. Turbulence closure

The turbulence closure uses a prognostic Eq. (8) for the filtered Sub-grid Scale Turbulent Kinetic energy (SGS-TKE) \bar{e} . The sub-grid scale terms are parameterized using 1.5 order closure by closely following the works Deardorff³⁸ and using a modified version of Wyngaard *et al.*³⁹ and Saiki *et al.*⁴⁰ For further information

TABLE I. Details about the CFD domain used for LES.

CFD domain size	Specification
Upstream (x-direction)	2H
Downstream (x-direction)	7H
Both lateral directions(y-direction)	2.5H
Above building (z-direction)	5H

ALGORITHM 1. ROM-LSTM approach.

- 1: Obtain 3D flow data from large eddy simulations data within the region of interest from the total CFD domain
- 2: Compute the mean and fluctuation flow fields for the given number of snapshots

$$\begin{aligned} \bar{u}(x, y, z, t_n) &= \frac{1}{N} \sum_{n=1}^N u(x, y, z, t_n) \\ u'(x, y, z, t_n) &= u(x, y, z, t_n) - \bar{u}(x, y, z, t_n). \end{aligned}$$

- 3: Compute the proper orthogonal decomposition basis for the data matrix \mathbf{A} , where the columns have the domain data at each snapshot of data using singular value decomposition

$$\mathbf{A} = \Phi \Sigma \mathbf{V},$$

where Φ is the basis vectors matrix and Σ is a diagonal matrix with singular values.

- 4: Using relative information content (RIC) of the singular values, the optimal number of POD modes are selected and corresponding basis vectors are stored.
- 5: Find the corresponding modal coefficients associated with basis vectors matrix Φ_w , obtained from the data matrix \mathbf{A} ,

$$\mathbf{C} = \mathbf{A}^T \Phi_w.$$

- 6: Pre-process the data by scaling and re-arranging data appropriately for LSTM neural network training with necessary look-back window conditions.
- 7: Train the LSTM neural network with the training data.
- 8: Predict the modal coefficients with the trained network for future time instances.
- 9: Using the basis vectors stored calculate the fluctuation field from the modal coefficients predicted, \mathbf{U}' ,

$$\mathbf{U}' = \Phi_w \mathbf{C}^T.$$

- 10: Compute the predicted flow field by adding the mean value to the predicted snapshot data.

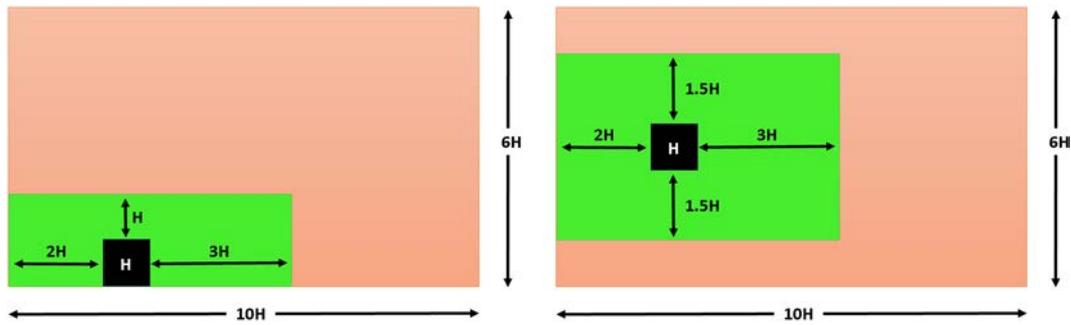


FIG. 1. Left-top view (xy-plane); right-side view (xz-plane) of the domain; green—domain of interest, red—domain neglected from total domain, and LES domain—total domain.

regarding the parameterization of various terms in the equation, the reader is referred to Ref. 36,

$$\frac{\partial \bar{e}}{\partial t} = -\bar{u}_j \frac{\partial \bar{e}}{\partial x_j} - \overline{u_i'' u_j''} \frac{\partial \bar{u}_i}{\partial x_j} + \frac{g}{\Theta_{v,0}} \overline{u_3'' \Theta_v''} - \frac{\partial}{\partial x_j} \left[\overline{u_j'' \left(e + \frac{p''}{\rho_0} \right)} \right] - \varepsilon, \quad (8)$$

where ε is the SGS dissipation rate.

3. Discretization and buildings

a. Discretization. The calculation domain is discretized by using equidistant finite-difference horizontal grid spacings in x and y directions. PALM by default uses constant grid spacing in the vertical or z-direction but allows an option for stretching the grid to reduce computational cost when required. However, in our application of simulating flow over a building, we only used the default constant grid-spacing in all the directions. PALM uses a C-grid topology called Arakawa⁴¹ staggered, which considers all scalar variables, such as the perturbation pressure *past*, SGS-TKE *e*, etc., to be located at the cell centers, while the components of vector variables, such as velocity (*u, v, w*), are shifted by half grid spacing to the corresponding face centers in the respective directions.

b. Buildings. PALM can conduct realistic simulations of atmospheric turbulence and its propagation, including the necessary topography information in the domain. PALM’s 3D topography implementation uses the mask method⁴² for resolving any obstacles in the domain. The method resolves solid obstacles, such as buildings, and treats the grid cell either as 100% fluid cell or 100% solid cell, to create a 2.5D topography map similar to Digital Elevation Model (DEM). For the present study, we focused essentially on

the neutral boundary layer and the obstacle surfaces are essentially approximated by a step-like function by the grid. PALM model can also treat over-hanging or detached structures in the newer versions but can only account for stationary obstacles.

B. Non-intrusive ROM-LSTM methodology

Large-eddy simulation data are pre-processed to obtain the necessary training data for the non-intrusive reduced order modeling approach. We obtain the fluctuation data of the flow field by subtracting the mean flow from the snapshot data at all the time instances required. Proper orthogonal decomposition using the singular value decomposition technique is used to compute the modes and corresponding basis for the complete data. Based on the threshold selected, the Relative Information Content⁴³ (RIC) index is used to decide on a cutoff mode. The modal coefficients until the cutoff mode are then used for training the neural network.

Recurrent Neural Networks (RNNs) are a widely used neural network architecture in time series prediction where the current output state is dependent on data from previous time instances. Recurrent Neural Networks (RNNs) contain cyclic or recurrent

TABLE II. Details about the domain used for ML-ROM.

ML-ROM domain size	Specification
Upstream (x-direction)	2H
Downstream (x-direction)	3H
Both lateral directions (y-direction)	1.5H
Above building (z-direction)	1H

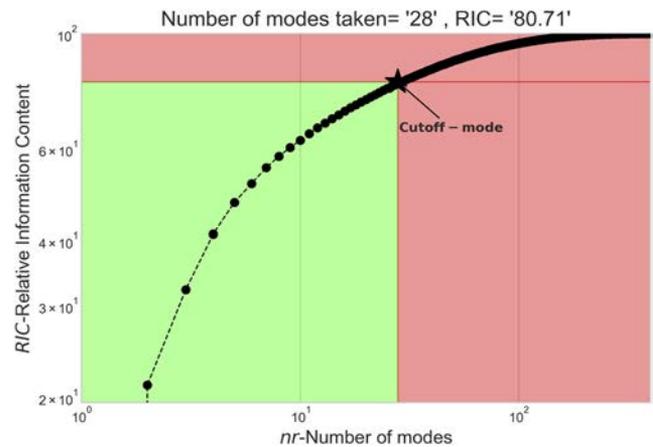


FIG. 2. Modes and their relative information content; green—modes taken and red—modes neglected.

TABLE III. Neural network details.

Parameter	Specification
Number of hidden layers	2
Number of neurons in each hidden layer	64
Activation function	tanh
Look-back time-window	20
Recurrent dropout	0.2
Neuron dropout	0.2
Loss function	MSE
Optimizer	ADAM
Training-testing ratio	4:1

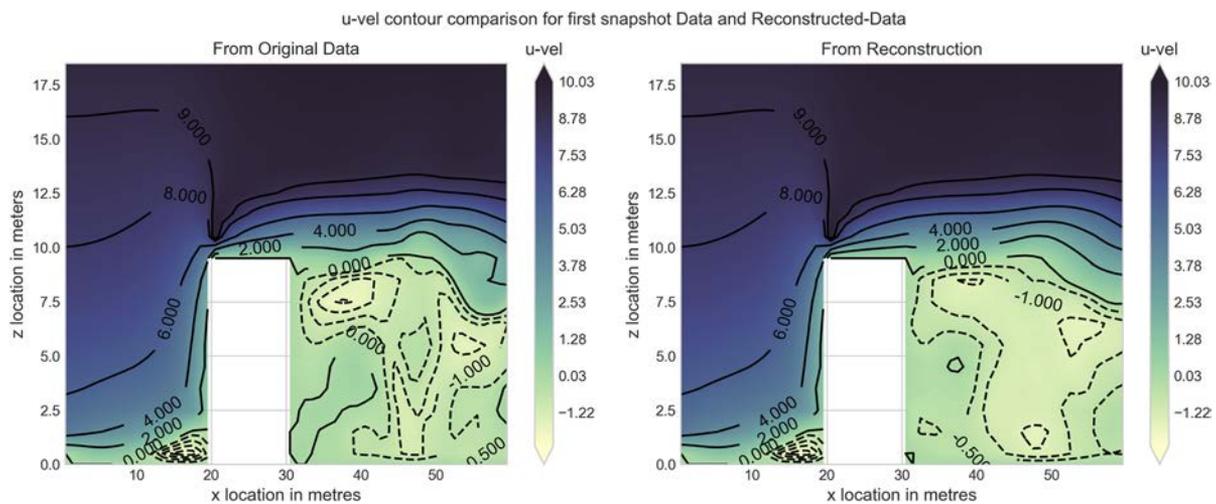
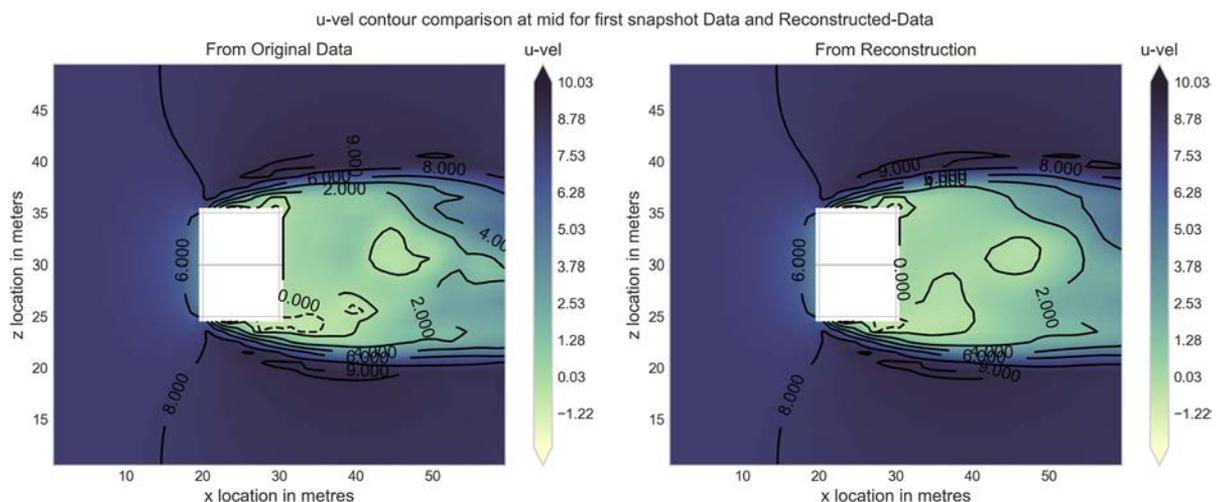
connections that enable them to continuously learn characteristics and remember them. However, they suffer from vanishing gradients and suffer from long training time and can only predict short sequences. Closely following,³¹ we use Long Short-Term Memory

(LSTM) neural networks, a special variant of RNN architecture better suited for learning long-term dependencies in the input data without suffering from the disadvantages associated with RNNs. After sufficiently training the LSTM network, we use it to predict the modal coefficients for future time instances. These predictions are then used to project the modal coefficients using the pre-calculated POD basis to the three-dimensional spatiotemporal domain in the physical space. Using the obtained fluctuation flow field, we can then compute the flow field using the mean flow data calculated in the pre-processing stage. The detailed steps followed are depicted in Algorithm 1.

III. RESULTS AND DISCUSSION

A. Simulation setup

For this study, we chose a simplistic setup of a single cubic building (height and width were equal to each other) in a neutral

**FIG. 3.** u-velocity contour in the XZ plane at the center of domain for the first snapshot.**FIG. 4.** u-velocity contour in the XY plane at the center of domain for the first snapshot.

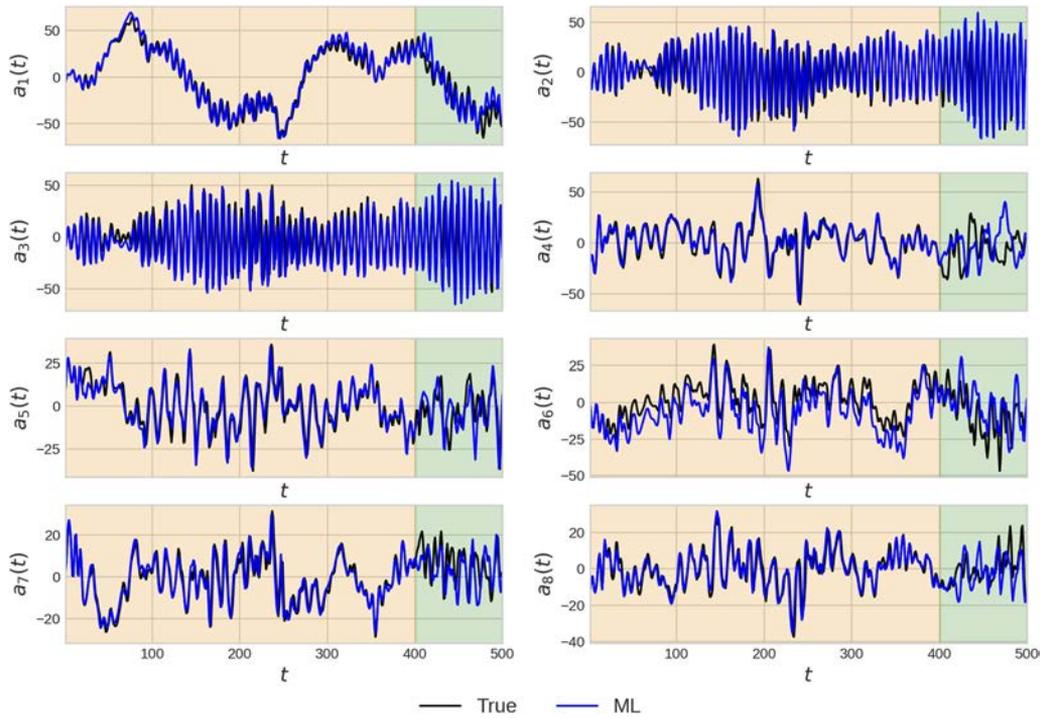


FIG. 5. Comparison between true and non-intrusive (ML) for first 8 modes; background colors: tan/orange—training; green—prediction.

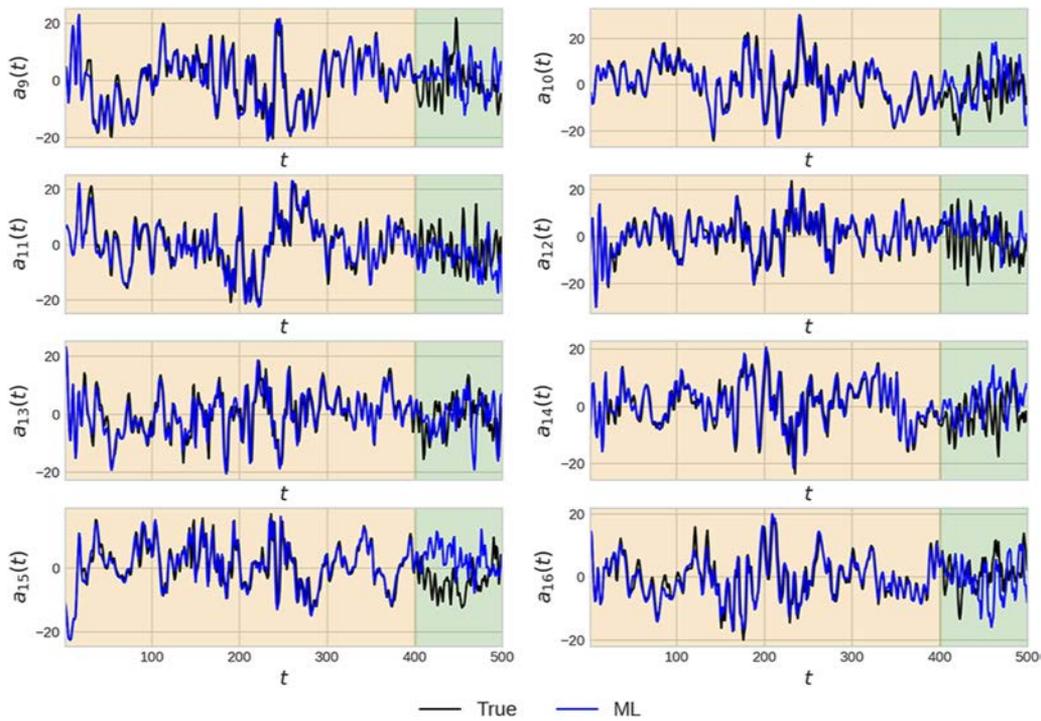


FIG. 6. Comparison between true and non-intrusive (ML) for 9-16 modes; background colors: tan/orange—training; green—Prediction.

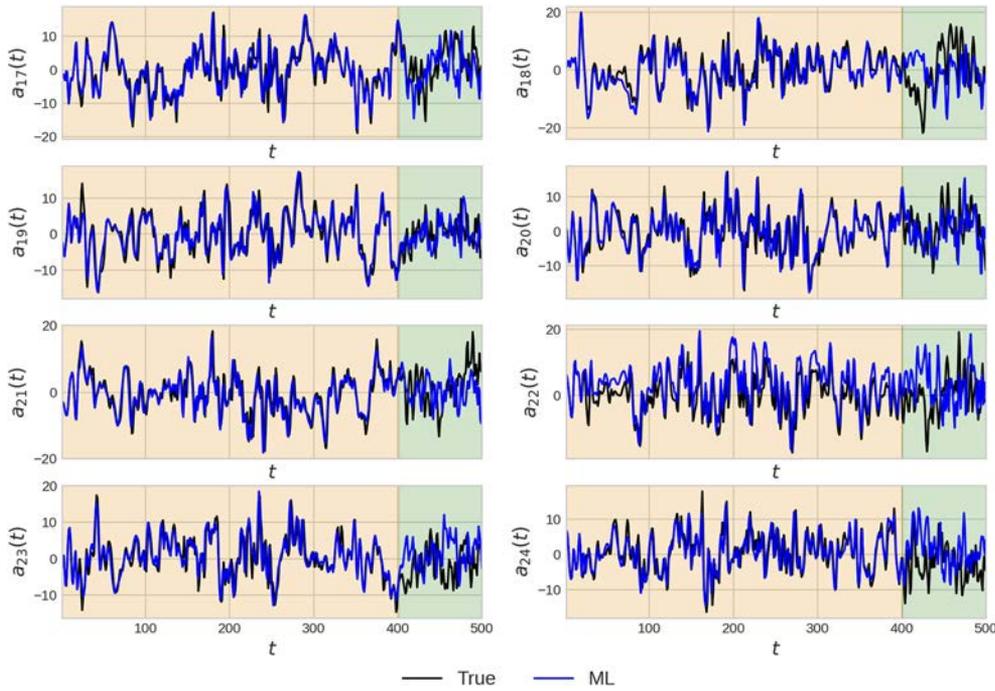


FIG. 7. Comparison between true and non-intrusive (ML) for 17-24 modes; background colors: tan/orange—training; green—prediction.

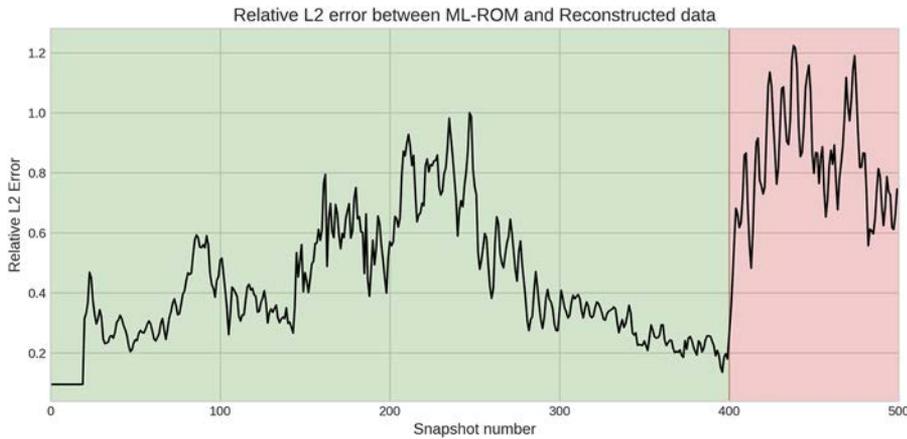


FIG. 8. Relative L2 error of the flow field (normalized with the max error in training phase), calculated between the reconstructed data for the exact modes and ML-ROM modes; background colors: green—training phase; red—prediction phase.

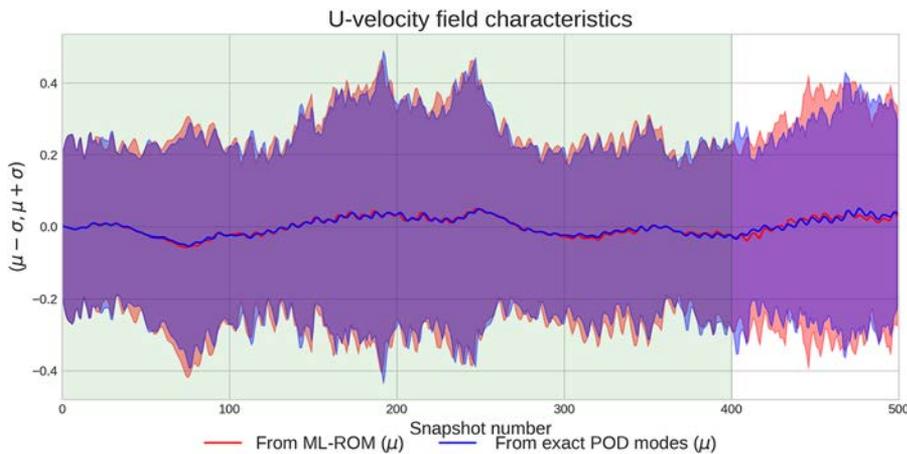


FIG. 9. Comparison between ML-ROM and POD reconstructed values for mean with their corresponding standard deviation ($\mu - \sigma, \mu + \sigma$); background colors: green—training; white—prediction.

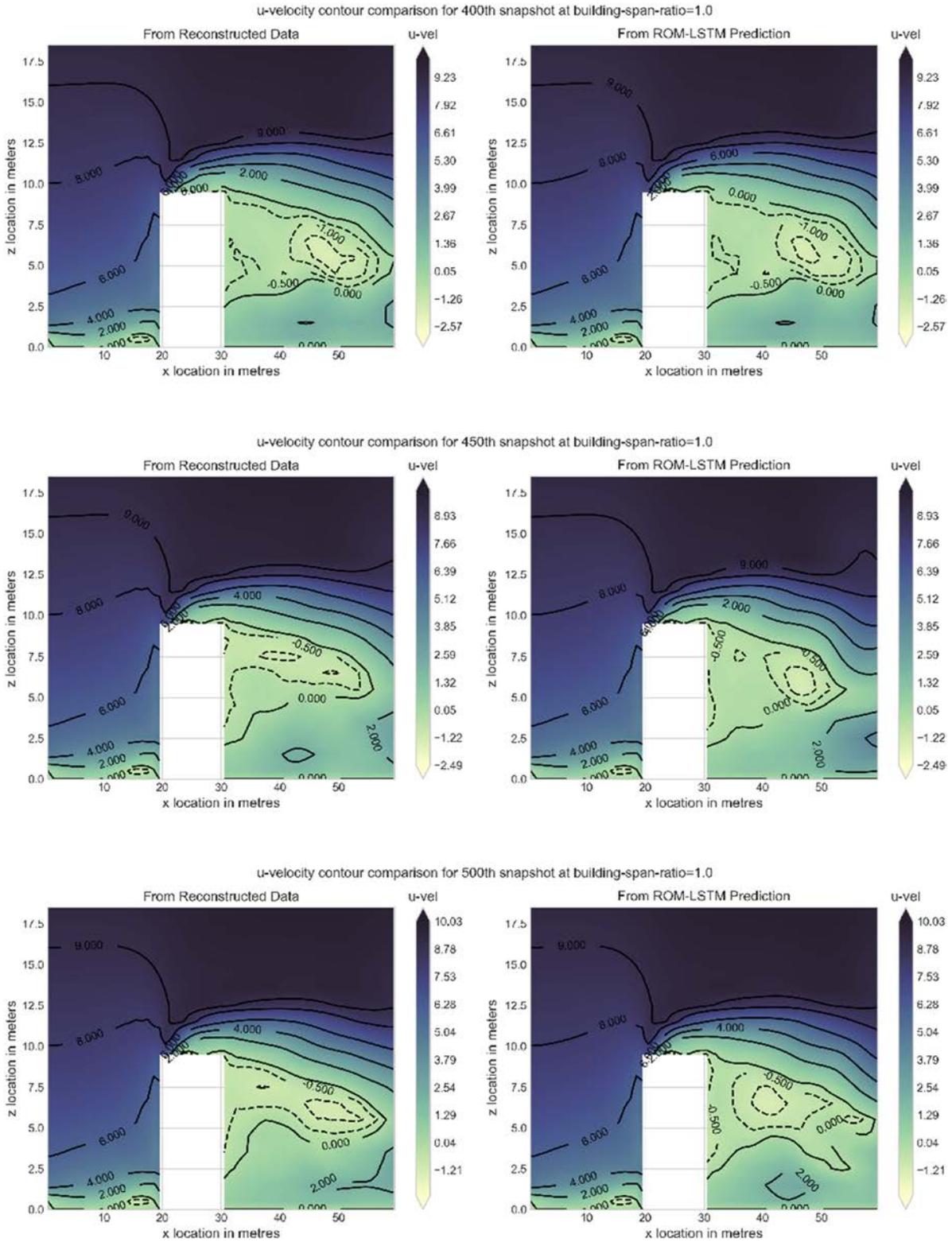


FIG. 10. u-velocity contour for xz plane at building span ratio of 1.0 at different instances.

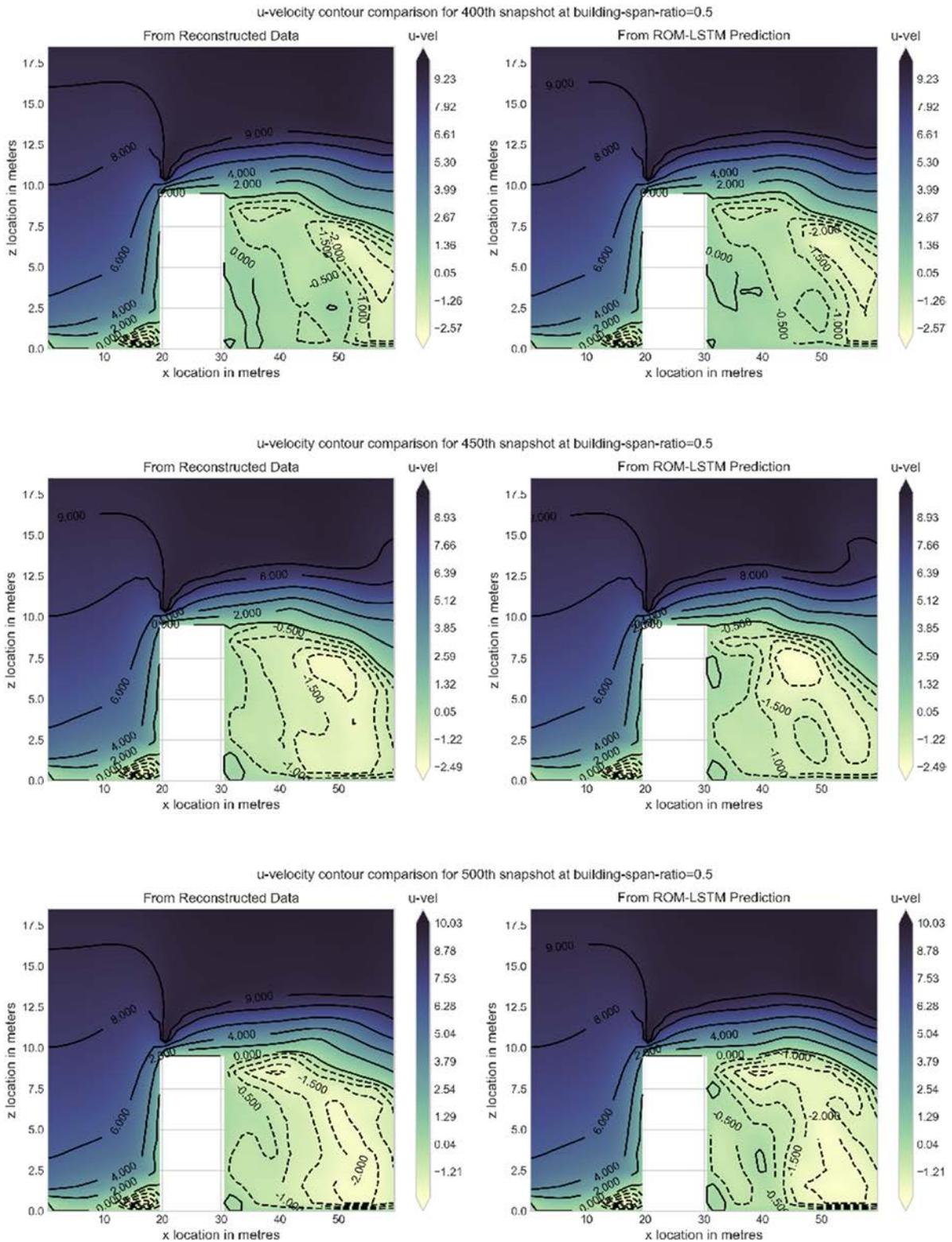


FIG. 11. u-velocity contour for xz plane at building span ratio of 0.5 (mid) at different instances.

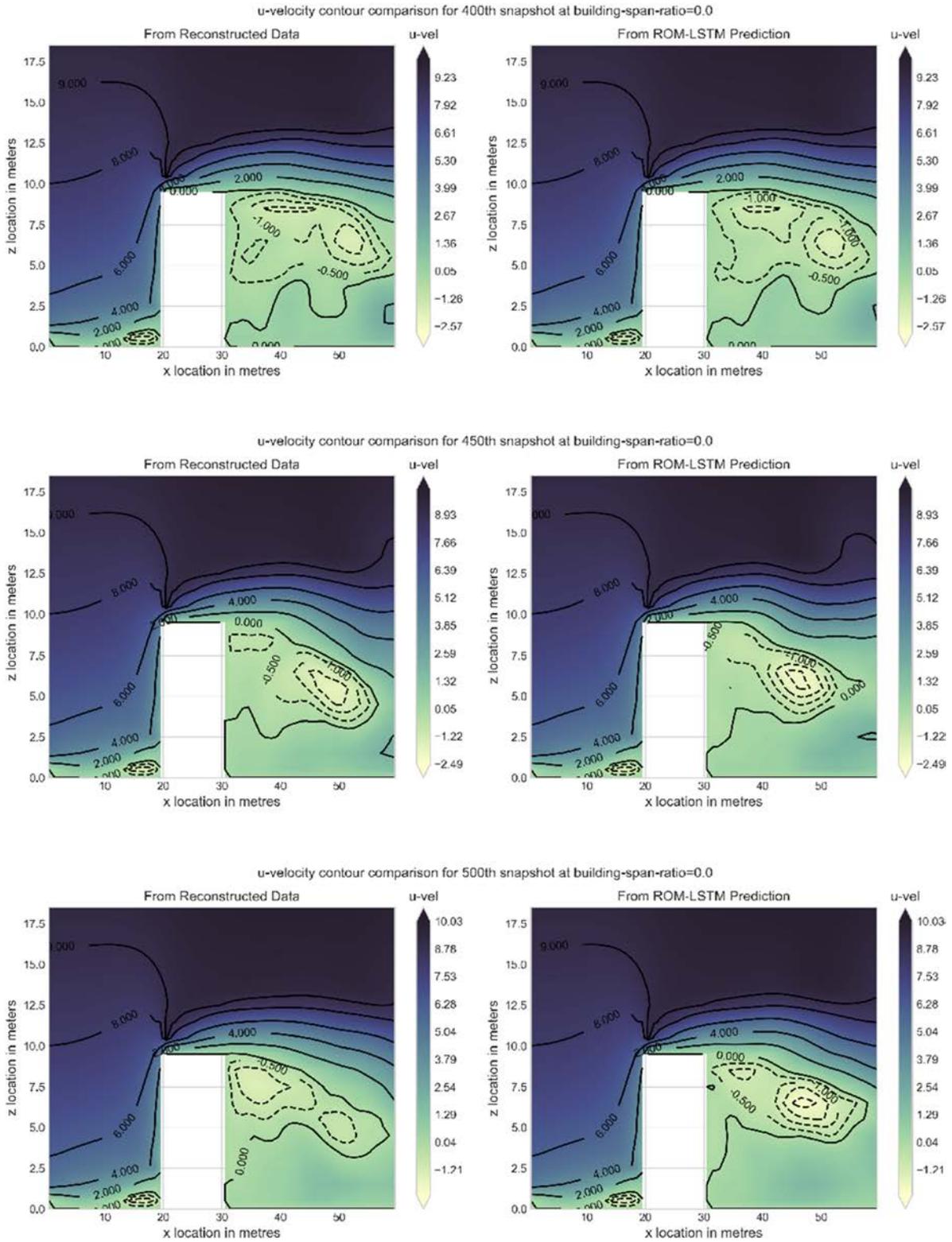


FIG. 12. u-velocity contour for xz plane at building span ratio of 0.0 at different instances.

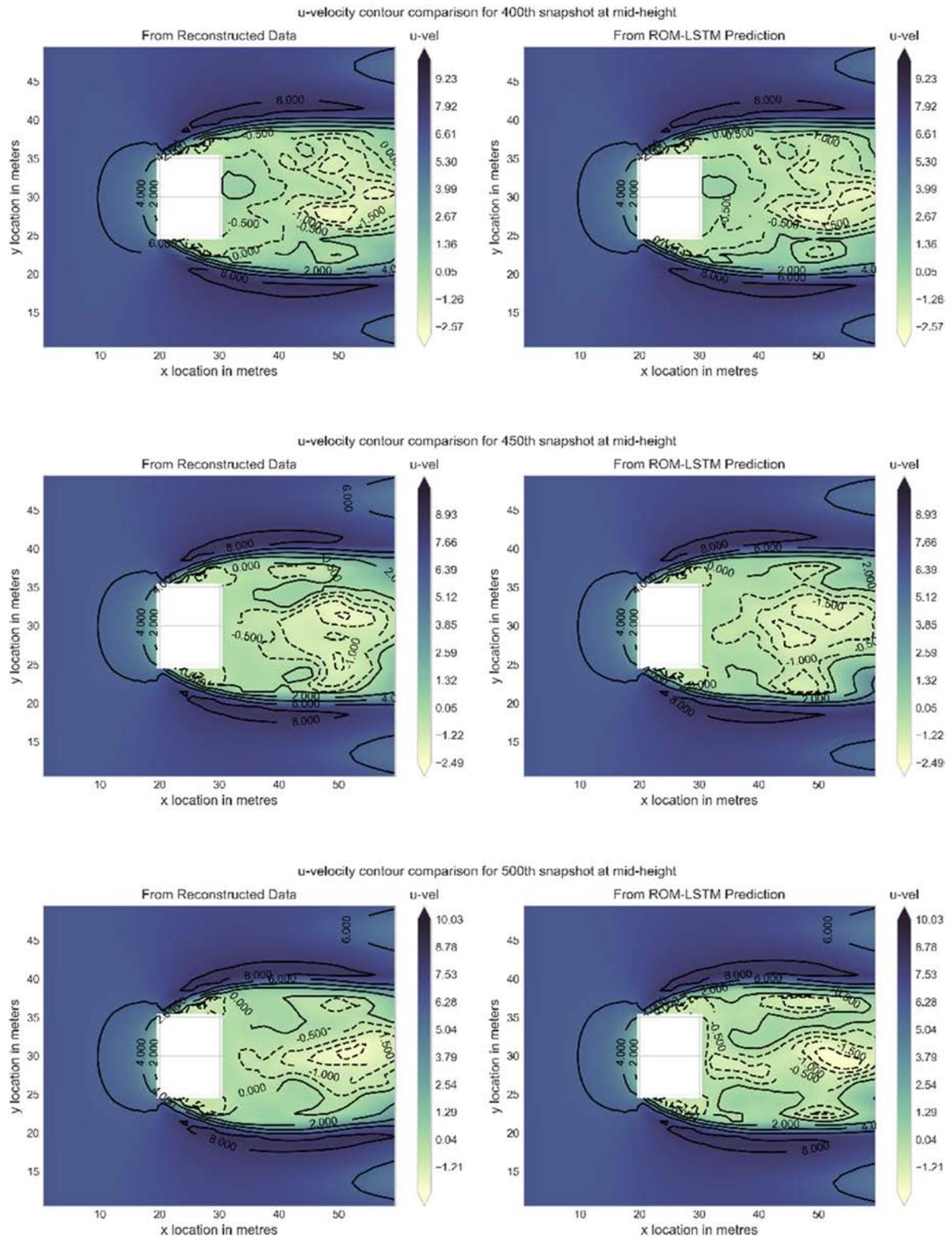


FIG. 13. u-velocity contour for the xy plane at midheight of the building.

atmospheric boundary layer with a specified velocity profile at the inlet of the domain. The large eddy simulation domain was chosen based on closely following the recommendations of Franke *et al.*⁴⁴ and Murakami and Mochida.⁴⁵ Vertical profile of the inflow was chosen proportional to $z^{1/4}$ until a height of $2H$,⁴⁵ similar to the setup of Tutar and Oguz.⁴⁶ The wind velocity at height H was matched to 8 m/s while using the vertical velocity profile $z^{1/4}$. Only the x-component of velocity u was initialized using the profile, while other components v, w were set to be zeros. We used a constant grid spacing of 1 m for the computational domain by following the best practices for CFD simulations of wind flows around buildings.^{44,47} Further details about the domain is tabulated in Table I and presented in Fig. 1. Dry atmospheric conditions were chosen with a Coriolis parameter of 7.3×10^{-5} with boundary conditions on the top and bottom (z-direction) as free-slip and no-slip, left and right (x-direction) as inflow and outflow, front and back (y-direction) as outflows, respectively.

B. Non-intrusive ROM results

To simplify the model and demonstrate the method, the total CFD domain used for the large eddy simulation is not used to build the non-intrusive ROM. Instead, we pick a smaller sub-domain of interest as depicted in Fig. 1 and described in Table II, in the proximity to the building since this represents our region of interest. The CFD simulation is run until a quasi-steady state or statistically, a stationary state is reached, before extracting the LES snapshot data for training the model. Furthermore, we model only the x-component of velocity u using the non-intrusive ROM. The LES data are obtained in this three-dimensional domain every 1 second, which will be referred to as snapshot data going forward. Following Algorithm 1, by picking a threshold of 80% based on the Relative Information Content (RIC), we settle on 28 modes as shown in Fig. 2. By making a comparison of the first snapshot data in the center of the domain in both top-view (xz-plane) and side view (xy-plane), as shown in Figs. 3 and 4, we notice that some finer details of the flow are lost, but the major features still seem to be well represented by these modes. The LSTM neural network compiled and trained on the data for 400 snapshots, and prediction is made for the next 100 snapshots. Some more information about the neural network architecture utilized for the study is listed in Table III.

First the first 24 modes, a comparison is made between the true modal coefficients and non-intrusive ROM predictions in Figs. 5–7. We see a close agreement in the modal coefficient predictions and the true modes obtained from the LES data. However, we still notice a slight mismatch in the amplitudes for all modes. Since they are arranged in the order of importance, this provides us with a good measure of the flow-field predictions made. L2 norm error is computed between the non-intrusive ROM data and the data reconstructed from the modes and normalized with the peak error in the training phase is plotted in Fig. 8. We could notice that the error in the prediction phase is within close limits to that in the training phase and peaks only at about 1.2 times the highest training error. We also compare the mean and standard deviation in Fig. 9 for the ML-ROM and POD reconstructed flow-fields in the domain for the entire duration and see good agreement. Contour plots in xz planes at different building span ratios and xy plane at the center

of the domain (top-view) are also plotted for comparison between the predicted u-velocity field and the flow-field from reconstruction using exact POD modes. Figure 10 represents the 2D flow field at the beginning of the building in a span-wise direction, Fig. 11 represents the 2D flow field at the center, and Fig. 12 represents the 2D flow field at the other end of the building. Additionally, the flow field in the center of the domain when viewed from the top (xy-plane) is depicted in Fig. 13. It could be noticed from these comparisons that there seems to be a mismatch between the finer structures in the contour plot. However, they do tend to have similar larger structures in the snapshots at 400, 450, and 500 s. As expected, the predictions are better at the 400th snapshot and the flow-field predictions deviate slowly as the number of the snapshot increases. However, we still see a close resemblance between the predictions and the POD reconstructed data. The mean with standard deviation values are plotted in 9 and we see good agreement.

IV. CONCLUSION

In this work, we tried to build a machine learning based non-intrusive reduced order model for predicting wind fields in a simplistic urban setup case of a singular cubic building. We intended to utilize an efficient, stable, and robust reduced order model and hence use proper orthogonal decomposition-long short-term memory based ML-ROM to demonstrate the method for this problem. We simplify the process for building the ML-ROM by choosing a smaller domain of interest close to the building and also by predicting only the major component of velocity in the x-direction, u . We see good agreement between the corresponding modal and flow-field predictions obtained from the ML-ROM. However, we do notice a deterioration in the predictions over time. While in work predictions were only made for a short duration, for future studies we intend to address longer predictions. We could utilize convolutional neural networks based auto-encoders to replace the proper orthogonal decomposition to generate a reduced order model. This would be capable of understanding both the structures or terrain inside the domain and the velocity field, improving predictions over a longer time duration. Furthermore, data assimilation could be used to augment the data in the prediction phase to address the deviations from actual data. However, this work provides only provides the preliminary framework and initial steps for devising a ML-based ROM for wind-field predictions in urban spaces.

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation (NSF) under Grant No. 1925147. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. Some of the computing for this project was performed at the High-Performance Computing Center (HPCC) at Oklahoma State University supported, in part, through the National Science Foundation (Grant No. OAC-1531128). We would like to acknowledge high-performance computing support from Cheyenne⁴⁸ (<https://doi.org/10.5065/d6rx99hx>) provided by the National Center for Atmospheric Research's (NCAR) Computational and Information Systems Laboratory (CISL), sponsored by the National Science Foundation (NSF). This work also used

computational resources as part of the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by the National Science Foundation (Grant No. ACI-1548562). Specifically, it used the Bridges2 system, which is supported by the NSF (Award No. ACI-1445606), at the Pittsburgh Supercomputing Center (PSC).

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Rohit K. S. S. Vuppala: Formal analysis (lead); Software (lead); Writing – review & editing (equal). **Kursat Kara:** Project administration (lead).

DATA AVAILABILITY

The data that support the findings of this study are openly available in Github at https://github.com/rohitvuppala/POD-LSTM_BLDG.git.⁴⁹ The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

- L. de Oliveira Silva, R. A. de Mello Bandeira, and V. B. G. Campos, *Transp. Res. Procedia* **27**, 1137 (2017).
- T. H. Grubestic and J. R. Nelson, *UAVs and Urban Spatial Analysis* (Springer, 2020), pp. 13–29.
- C. D. León, Drone delivery? amazon moves closer with f.a.a. approval, 2020 (<https://www.nytimes.com/2020/08/31/business/amazon-drone-delivery.html>).
- I. C. A. Organization, *Unmanned Aircraft Systems (UAS)* (International Civil Aviation Organization, Montreal, 2011), https://www.icao.int/Meetings/UAS/Documents/Circular%20328_en.pdf.
- R. Austin, *Unmanned Aircraft Systems: UAVS Design, Development and Deployment* (John Wiley & Sons, 2011), Vol. 54.
- M. Agarwal and A. Tandon, *Appl. Math. Modell.* **34**, 2520 (2010).
- P. Kastner-Klein, R. Berkowicz, and R. Britter, *Meteorol. Atmos. Phys.* **87**, 121 (2004).
- A. Onishi, X. Cao, T. Ito, F. Shi, and H. Imura, *Urban For. Urban Greening* **9**, 323 (2010).
- J. Corburn, *Urban Stud.* **46**, 413 (2009).
- J. C. D. C. Siqueira, Modeling of Wind Phenomena and Analysis of Their Effects on UAV Trajectory Tracking Performance (Statler College of Engineering and Mineral Resources, 2017), <https://researchrepository.wvu.edu/etd/7347/>.
- S. A. Raza, “Autonomous UAV control for low-altitude flight in an urban gust environment,” Ph.D. thesis, Carleton University, 2015, https://carleton.ca/atarg/wp-content/uploads/Thesis_SAR_Final.pdf.
- K. Kirk, C. Lederman, Y. Wang, and B. Kraczek, “The inclusion of realistic winds in a simulated environment for the study of wind-unmanned aircraft system (UAS) interactions,” Technical Report No. AD1170051 (DEVCOM Army Research Laboratory, 2022); available at <https://apps.dtic.mil/sti/pdfs/AD1170051.pdf>.
- M. Specification, “Flying qualities of piloted airplanes,” Technical Report No. MIL-F-8785C (Department of Defense, US, 1980).
- S. M. Salim, R. Buccolieri, A. Chan, and S. Di Sabatino, *J. Wind Eng. Ind. Aerodyn.* **99**, 103 (2011).
- R. K. S. S. Vuppala and K. Kara, “Large-eddy simulation of atmospheric boundary-layer gusts for small unmanned air systems,” *Bulletin. of Americal. Physical. Society.* **65**(13), (2020) <https://meetings.aps.org/Meeting/DFD20/Session/H12.7>.
- A. Tabassum, R. K. S. S. Vuppala, H. Bai, and K. Kara, “Variance reduction of quadcopter trajectory tracking in turbulent wind,” *IFAC-PapersOnLine* **54**(20), 102–107 (2021).
- M. Sutherland, “Urban wake field generation using LES for application to quadrotor flight,” Ph.D. thesis, Carleton University, 2015, https://curve.carleton.ca/system/files/etd/6508407f-0cd2-4e17-ada8-5610a5662e0f/etd_pdf/91f3d0b9b-769af97087d53d70b96580/sutherland-urbanwakefieldgenerationusinglesfor-application.pdf.
- Y.-d. Lang, A. Malacina, L. T. Biegler, S. Munteanu, J. I. Madsen, and S. E. Zitney, *Energy Fuels* **23**, 1695 (2009).
- B. R. Noack, M. Morzynski, and G. Tadmor, *Reduced-Order Modelling for Flow Control* (Springer Science & Business Media, 2011), Vol. 528.
- X. Xie, M. Mohebujjaman, L. G. Rebholz, and T. Iliescu, *SIAM J. Sci. Comput.* **40**, B834 (2018).
- B. D. Tracey, K. Duraisamy, and J. J. Alonso, in *53rd AIAA Aerospace Sciences Meeting* (AIAA, 2015), p. 1287.
- L. Zhu, W. Zhang, J. Kou, and Y. Liu, *Phys. Fluids* **31**, 015105 (2019).
- R. Maulik, H. Sharma, S. Patel, B. Lusch, and E. Jennings, “A turbulent eddy-viscosity surrogate modeling framework for Reynolds-averaged Navier-Stokes simulations,” *Comput. Fluids* **227**, 104777 (2021).
- N. Thuerey, K. Weissenow, L. Prantl, and X. Hu, *AIAA J.* **58**, 25 (2020).
- R. K. Vuppala and K. Kara, in *AIAA Aviation 2021 Forum* (AIAA, 2021), p. 2505.
- R. K. S. S. Vuppala and K. Kara, “Realistic Wind Data Generation for Small Unmanned Air Systems in Urban Environment using Convolutional Autoencoders,” *Bull. Am. Phys. Soc.* **66**(17), T29. 005 (2021) <https://meetings.aps.org/Meeting/DFD21/Session/T29.5>.
- T. R. Landua, R. K. S. S. Vuppala, and K. Kara, in *AIAA Scitech 2022 Forum* (AIAA, 2022), p. 1688.
- H. Eivazi, H. Veisi, M. H. Naderi, and V. Esfahanian, *Phys. Fluids* **32**, 105104 (2020).
- S. A. Renganathan, R. Maulik, and V. Rao, *Phys. Fluids* **32**, 047110 (2020).
- S. Pawar, S. M. Rahman, H. Vaddirreddy, O. San, A. Rasheed, and P. Vedula, *Phys. Fluids* **31**, 085101 (2019).
- S. M. Rahman, S. Pawar, O. San, A. Rasheed, and T. Iliescu, *Phys. Rev. E* **100**, 053306 (2019).
- S. Pawar, S. E. Ahmed, O. San, and A. Rasheed, *Phys. Fluids* **32**, 036602 (2020).
- S. Pawar, S. E. Ahmed, O. San, and A. Rasheed, *Mathematics* **8**, 570 (2020).
- S. Rahman, O. San, and A. Rasheed, *Fluids* **3**, 86 (2018).
- O. San and R. Maulik, *Phys. Rev. E* **97**, 042322 (2018).
- B. Maronga, M. Gryscha, R. Heinze, F. Hoffmann, F. Kanani-Sühring, M. Keck, K. Ketelsen, M. O. Letzel, M. Sühring, and S. Raasch, *Geosci. Model Dev.* **8**, 2515 (2015).
- U. Schumann, *J. Comput. Phys.* **18**, 376 (1975).
- J. W. Deardorff, *Boundary-Layer Meteorol.* **18**, 495 (1980).
- J. C. Wyngaard, L. J. Peltier, and S. Khanna, *J. Atmos. Sci.* **55**, 1733 (1998).
- E. M. Saiki, C.-H. Moeng, and P. P. Sullivan, *Boundary-Layer Meteorol.* **95**, 1 (2000).
- A. Arakawa and V. R. Lamb, *Methods Comput. Phys.* **17**, 173 (1977), part of Special Issue on General Circulation Models of the Atmosphere.
- M. Briscolini and P. Santangelo, *J. Comput. Phys.* **84**, 57 (1989).
- M. D. Gunzburger, *Perspectives in Flow Control and Optimization* (SIAM, 2002).
- J. Franke, A. Hellsten, K. Schlünzen, and B. Carissimo, in *11th Conference on Harmonisation within Atmospheric Dispersion Modelling for Regulatory Purposes* (Cambridge Environmental Research Consultants, Cambridge, 2007).

⁴⁵S. Murakami and A. Mochida, *J. Wind Eng. Ind. Aerodyn.* **31**, 283 (1988).

⁴⁶M. Tutar and G. Oguz, *Fluid Dyn. Res.* **31**, 289 (2002).

⁴⁷J. Franke, A. Hellsten, K. H. Schlunzen, and B. Carissimo, *Int. J. Environ. Pollut.* **44**, 419 (2011).

⁴⁸Computational and Information Systems Laboratory. 2019. Cheyenne: HPE/SGI ICE XA System (NCAR Community Computing). Boulder, CO: National Center for Atmospheric Research, <https://doi.org/10.5065/D6RX99HX>.

⁴⁹R. K. S. S. Vuppala and K. Kara, POD-LSTM-BLDG, 2022, https://github.com/rohitvuppala/POD-LSTM_BLDG.