Accelerated Computational Fluid Dynamics in Built Environment

Cormac McCumiskey[a], Nima Gerami Seresht[a,*]

[a]*Durham University, Department of Engineering, South Road, Durham, DH1 3LE, UK*

Abstract

Computational fluid dynamics (CFD) is an indispensable tool in several building engineering applications, including ventilation system design and building energy modeling. In these applications, CFD helps engineers improve buildings' energy efficiency and resilience. Despite these successful implementations, the high computational cost of CFD simulation precludes its application in several engineering problems. As a remedy, proxy models of CFD have been recently introduced to replicate their simulation results by machine learning models at a significantly reduced computational cost. However, existing proxy models are lacking in terms of accuracy and generalizability. This research aims to alleviate these issues. In the context of CFD proxy modeling, this paper introduces a novel two-layer machine-learning architecture and a preprocessing algorithm that transforms spatial coordinates to improve the model's accuracy and scalability (AI Novelties). The novel CFD proxy model introduced in this paper can accelerate CFD simulation by 99.996% while achieving 95% prediction accuracy. This model can expand the application of CFD simulation in several building engineering problems (Engineering Application).

Keywords: Computational Fluid Dynamics, Proxy Modeling, Ventilation, Building Modeling, Sustainability, Resilience

1. Introduction

Indoor environmental modeling is crucial in several applications in building engineering, such as energy efficiency optimization [1], air quality as-

*Corresponding author: nima.gerami-seresht@durham.ac.uk

sessment [2], and the aerosol transmission of infectious diseases [3]. In this context, computational fluid dynamics (CFD) is widely applied to predict velocity flow fields, which, in turn, helps assess the efficiency of natural and mechanical ventilation, as well as the dispersion of aerosols inside indoor spaces. CFD is proven to be an effective tool for this purpose; however, its implementation often requires domain expertise, high-performance computers, and long run times, particularly in larger indoor environments. These challenges preclude its application in several building engineering problems, particularly where simulation results are required at an accelerated pace, such as real-time control systems or coupling with meta-heuristic optimization algorithms.

The rise of the COVID-19 pandemic brought this limitation to the attention of the research and innovation community since aerosol has been identified as the dominant transmission route of the coronavirus [3], which is best addressed by mechanical ventilation systems. This highlights the need for a better understanding of aerosol dynamics in indoor spaces in real-time to improve air ventilation in buildings and enhance their resilience against infectious diseases. The National Library of Medicine emphasizes the potential of CFD in elucidating disease transmission mechanisms but notes a research gap in understanding how room conditions such as windows, doors, and ventilation rates affect pathogen spread [4]. This gap exists due to the high computational cost of CFD simulation, which makes it almost impossible to run several simulations to test the impact of changes in the building layout [5].

During the COVID-19 pandemic, maintaining well-ventilated public spaces became crucial for reducing pathogen concentrations. However, if a comprehensive grasp of airflow dynamics is lacking, increased ventilation might inadvertently exacerbate pathogen dispersal. This assertion is supported by the findings of Birnir et al. [6], who analyzed fluid dynamics in a restaurant setting with recommended air-conditioning rates. Their findings indicate that the risk of transmission when dining at separate tables within fifteen minutes is comparable to that of sitting adjacent to an infected individual. In another example, Ren et al., [7] introduced a zonal demand-controlled ventilation system that uses cameras to detect room occupancy and determine the appropriate ventilation rate based on CFD simulation data. Their findings show that such CFD-informed control systems could reduce the infection rate to 2.8% and reduce energy consumption by 34%.

Another prominent CFD application in building engineering focuses on

modeling and optimizing building energy consumption. According to the World Green Building Council [8], 28% of global carbon emissions come from heating, cooling, and powering buildings, yet very little is known about how real-time control mechanisms can improve operational efficiency. Wu et al. [9] proposed a house with over 300 sensors to act as inputs to a CFD system, which would then inform how the actuators should operate to increase efficiency. These sensors would then be combined with local weather forecasts to create a digital twin of the house so that actuators could optimize the thermal comfort of the house while reducing its energy consumption. This study was limited by the fact that the actuators cannot react to real-time changes to boundary conditions because of long simulation times. There are also several efforts made to improve the energy efficiency of ventilation systems by introducing model predictive control (MPC) systems [1, 10]. MPC leverages the power of CFD simulation to determine the most efficient ventilation strategy in a multi-zone indoor space, reducing the energy consumption of ventilation systems by up to 35% [1]. Although several studies have confirmed MPC's effectiveness, its real-world application is still hindered by the high computational cost of CFD simulation and the lack of powerful hardware to run it and inform the control system in real time.

Recognizing these limitations, there is growing interest in developing smart proxy models (SPMs) to replicate the CFD simulation results at a significantly reduced computational cost. SPMs leverage the power of machine learning (ML) to overcome the high computational cost of CFD simulation. While numerical simulations rely on iterative mesh-grid updates, smart proxy models use data-driven approaches to predict complex indoor airflow dynamics. This research introduces an SPM for CFD applications in building engineering. Our proposed model uses an original architecture that improves its accuracy and scalability compared to the existing state-of-the-art models. The proposed SPM can predict indoor airflow velocity fields with 95% accuracy (mean absolute percentage error) at only 0.004% computational cost of equivalent CFD simulation. Hence, it offers great potential in several applications in building engineering, including real-time ventilation control and building layout optimization using heuristic methods.

The structure of this paper is as follows: Section 2 reviews previous work in accelerated CFD simulation, providing insights into the advantages and limitations of research on CFD applications in building engineering and existing SPMs. Section 3 introduces the methodological structure of the SPM introduced in this paper. Section 4 introduces the development iterations

of the proposed SPM and its final architecture. Section 5 outlines the results and discusses the applications of SPM in real-world building engineering problems. Finally, Section 6 presents the research conclusions and identifies the future areas of research and development in this context.

2. Literature Review

2.1. CFD Simulation in Built Environment

Reviewing the building engineering literature reveals two prominent applications of CFD in building engineering: (i) building energy modeling and designing HVAC systems and (ii) simulating the transmission of infectious diseases in indoor environments.

CFD simulation is widely used for modeling and improving the energy efficiency of HVAC systems, which is by far the most energy-consuming end-use in buildings, contributing to 30%-50% of the total building energy consumption [11, 12]. Model-Predictive Control (MPC) is a mechanism developed for the optimal control of HVAC systems based on the velocity flow fields acquired by CFD simulation. Several studies confirm the superior efficiency of MPC for HVAC systems in comparison to manually- or Demand-Controlled Ventilation (DCV) [1, 10, 13]. Since generating CFD simulation results in real-time is impractical, existing MPC methods often use CFD simulation results generated during the design stage by only considering static building features and constant, pre-set ventilation rates [11]. These algorithms fail to account for dynamic changes in indoor environments. This oversight results in significant differences between the effectiveness of these algorithms in the simulation environment (i.e., fully controlled) and field tests [13], as such dynamic changes often significantly impact airflow patterns [14, 15, 16, 17]. Coupled with a real-time CFD simulation engine, existing MPC algorithms can significantly reduce buildings' carbon footprint by lowering HVAC systems' energy consumption.

In terms of disease transmission control in buildings, the indoor environment is often considered well-mixed flow conditions; however, several studies confirm that this assumption does not hold true since ventilation in buildings is very complex and requires accurate CFD simulation to determine the risk of infection inside buildings [18]. A comprehensive review by Tsang et al. [19, 20] reveals the significant role of CFD simulation in enhancing the resilience of healthcare facilities against the spread of infectious diseases.

In previous studies, researchers used CFD simulation to determine the impact of static (e.g., layout) [21] and dynamic building features (e.g., occupants, doors) [14] on the effectiveness of mechanical ventilation on controlling disease transmission in healthcare establishments. Despite these successful implementations and promising results in improving ventilation systems' effectiveness, the high computational cost of CFD simulation precludes its application in real-time control of HVAC systems in hospitals [5].

The methodology we introduce in this paper can address these limitations by delivering robust (i.e., fast and accurate) predictions of velocity flow fields in building environments using a novel modeling architecture that improves both accuracy and generalizability of CFD proxy models. The proposed SPM enables building and mechanical engineers to run CFD simulations at an accelerated pace, realizing real-time control and optimization applications.

2.2. Smart Proxy Models for CFD

Smart Proxy Models (SPMs) harness ML algorithms to develop efficient surrogate models for complex numerical simulations, including discrete element method [22, 23], computational fluid dynamics [24, 5], and other complex simulations [25]. These CFD modeling applications, SMPs are proven effective in scenarios involving the intricate simulation of spatio-temporal flow structures in fluids [5]. They bypass computationally intensive tasks by learning patterns in simulation data and provide quick yet accurate approximations of the data.

Aboaba et al. [26] an SPM to replicate thermal-flow patterns of pressure, temperature, and species concentrations (nitrogen, oxygen, and carbon dioxide) to less than 10% error. Their approach also had faster execution times compared to numerical CFD simulation methods. These results were achieved with a limited amount of training data, and the computations took 5-6 minutes on an "inexpensive" laptop, contrasting with the 24 hours needed to collect the training data on a high-performance computer. Shahkarami and Mohaghegh [27] utilized SPMs to expedite subsurface modeling for enhanced oil recovery, achieving highly accurate results with 98.9% less computational time. Wui et al. [28] introduced SPMs to streamline the modeling of fractured reservoirs, notorious for their complexity. By analyzing a database derived from numerical simulations, SPMs identify patterns, reducing computational demands. These models accurately predicted production rates, including fuel and water oil, in a fraction of the time compared to traditional methods, demonstrating their efficiency and effectiveness.

An obstacle that challenges the development of SPMs is their extensive demand for computationally expensive training data. Researchers tackled this challenge using feature engineering methods, such as (i) developing a generalizable form for representing input data, such as defining each point based on its distances to each boundary condition [26, 5], (ii) implementing fuzzy clustering algorithms to assess the importance of each input feature in determining the velocity flow fields in different regions of the indoor environment [29, 30], and (iii) implementing advanced feature selection algorithms to reduce the input space dimension [28].

Haghshenas et al. [31] used SPMs to predict optimal water injection sites, aiming to boost oil recovery rates. They assembled a training dataset using Latin Hypercube Sampling to distribute oil injection sites evenly across the search space. To improve the performance of their SPM, Haghshenas et al. [31] practiced input feature normalization and analyses to reduce the dimensionality of the input space by introducing hyper-features. Despite the improvements in their model accuracy, they observed that inherent errors within the ANN accumulated over time, resulting in a decrease in model performance even with a large dataset size.

An alternative architecture for SPMs in such applications is ensemble learning [32]. Ensemble learning is a machine learning technique that combines multiple models to produce a more accurate predictive model. While it can be challenging to make a single ANN generalizable to different flow scenarios, an ensemble network can provide higher accuracy with a reduced computational load. For instance, Bussov and Nättilä [33] used computer vision to predict turbulent fluid flow, integrating an ensemble of models to extract various shapes in the images and combine them to construct large-scale structures in the data.

This paper introduces an SPM for CFD simulation in building engineering applications, which utilizes (i) a two-layered ensemble learning architecture to improve accuracy and generalizability, (ii) a novel input feature definition protocol based on a combination of spherical and Cartesian coordinate systems to improve the model's scalability, and (iii) an innovative loss-function to improve the model training efficiency in different regions of the building.

3. Methodology

3.1. Fluid Dynamics

In fluid dynamics, the behavior of fluid flow can vary widely depending on factors such as velocity, viscosity, and geometry. Laminar flow is defined by the smooth, orderly movement of fluid particles along parallel paths, dominated by viscosity, ensuring stability. In contrast, turbulent flow is chaotic and typically occurs at higher speeds; small localized swirling eddies promote intense mixing and velocity fluctuations. During the transition from laminar to turbulent flow, small localized eddies begin to disrupt ordered laminar flows. Stagnant flow occurs when fluid particles stop at obstacles or due to the non-slip boundary condition imposed by static surfaces, such as walls.

Different flow regimes are governed by distinct equations and exhibit unique phenomena. CFD flow solvers, such as ANSYS Fluent used in this research, often utilize Reynolds Averaged Navier Stokes (RANS) equations to represent a time-averaged form of the Navier-Stokes equations [34]. In RANS equations, flow variables are decomposed into mean and fluctuating components, and typically, the mean flow is further separated into laminar and turbulent components. Turbulent flow is then handled as a statistical phenomenon.

The fluid flow regimen can be determined based on the Reynolds number of fluid flow, calculated using Equation 1. Typically, a Reynolds number of less than 2,000 will indicate laminar flow, and greater than 4,000 indicates turbulent flow [35].

$$Re = \frac{\rho u L}{\mu} \tag{1}$$

where ρ is the density of the fluid ($\frac{kg}{s}$), u is the fluid velocity ($\frac{m}{s}$), L is the characteristic dimension (m), and μ is the kinematic viscosity ($\frac{Ns}{m^2}$). Reynold's number will be used in this research to segment the entire room airflow into four distinct flow regimes.

For the design of mechanical ventilation systems, the volumetric flow rate of air is designed to meet the minimum requirements by the standards. Given the mass flow rate of air $\dot{m}(\frac{kg}{s})$, its volumetric flow rate ($\frac{m^3}{s}$) can be calculated using Equation 2.

$$\dot{Q}(L) = 1,000\frac{\dot{m}}{\rho} \tag{2}$$

In this research, a simple HVAC system was created to simulate indoor air conditioning systems in a typical office building, using 400mm by 400mm ducts to maintain a comfortable airspeed of 0.2 m/s [36]. Experimentally, this velocity corresponds to a mass flow rate of 0.015 kg/s using equation 2. With a recommended air mass flow rate of 2.5 l/s per occupant [2], the system can accommodate four people in a 5m x 5m x 3m room, typical for office spaces.

The model was created in ANSYS Discovery, and a smooth-transition boundary layer has been introduced along all four walls to enforce a no-slip boundary condition, an approximation for the transition from stagnation flow on the walls to the free-stream flow in the body of the room. The volume enclosed by the surface was given the thermodynamic properties of air at 25°C, and the four walls were given the surface roughness properties of concrete to replicate a modern office design. A maximum mesh size of 0.025 m was chosen so that the tolerance for velocity variations between trials is less than 5%.

For this investigation, we utilized the steady-state solution scheme, ideal for scenarios where HVAC rates remain constant for extended periods, resulting in stable velocity-flow fields within the office. In most commercial buildings, including office buildings, the mechanical ventilation systems are controlled by the building automation systems (BAS), which the steady-state solution scheme can best represent. Initial results confirmed that the laminar, turbulent regions around the inlet and outlet significantly affect fluid velocity profiles; hence, the $k - \omega$ turbulence model is chosen to represent turbulent flow accurately. Following the simulation completion, data consisting of x, y, z positions and velocities at each node in the mesh were exported for training, testing, and validating the SPM.

3.2. Coordinate Transformation

In several existing CFD proxy models, the input features are the location of each grid point on the mesh, represented in the 3-dimensional Cartesian coordinate system. The outputs of these models —velocity vectors— are also often represented in a Cartesian system. However, this representation of the input and output features diminishes the generalisability of the model as it is dependent on the environmental context [37]. Furthermore, representing the velocity vectors in the Cartesian system causes further challenges due to independent variation of velocity components along different axes (e.g., moving in the x direction can significantly change velocity in the z direction).

As a result, the ML models developed with this set of input features require more hidden layers to extract abstract spatial features of the room, which, in turn, increases the computational cost of these models.

Through our pilot study, these limitations led to instabilities during the training process, hindering the SPM's ability to discern any meaningful patterns in the training data (results are presented in Table 3). To tackle this challenge, an alternative approach is utilized by representing the model's input features and the outputs (i.e., velocity vectors) by transforming them into the spherical coordinate system using equations 3, 4, and 5.

$$r = \sqrt{(v_x)^2 + (v_y)^2 + (v_z)^2} \tag{3}$$

$$\theta = \arccos\left(\frac{v_z}{r}\right) \tag{4}$$

$$\phi = \text{atan}\left(\frac{v_y}{v_x}\right) \tag{5}$$

In this system, the velocity components are combined into a single radial magnitude, while θ and ϕ represent the inclinations of the vector to the vertical and horizontal axes, respectively.

Additionally, transforming the coordinate distances to the HVAC inlet and outlet into spherical vectors offers significant advantages for the learning process of the model. By consolidating the coordinate distances (Δx, Δy, Δz) into a single distance magnitude, denoted as r, the distance becomes strictly positive, providing a more intuitive representation for machine learning. Moreover, incorporating the reciprocal of the distance reflects the physical phenomenon where velocity increases as the distance to the vent decreases. However, distances less than 1 mm were standardized to 1mm to avoid numerical instability, effectively bounding the feature within a range of 0 to 1,000 and stabilizing the model during training. Fluid velocity near boundaries tends to be zero. However, the direction of velocity also changes around walls due to the re-circulation of air being deflected. This relationship between fluid velocity and wall distance is, therefore, complex, and the inputs were left as raw distances. Table 1 shows the representation of the models' input features and output, and Figure 1 shows their graphical representation.

In the context of spherical vectors to the inlet and outlet, the polar angle (θ) denotes the angle between the positive z-axis (vertical), while the azimuthal angle (ϕ) signifies the angle on the x-y plane. To align with the natural airflow throughout the room, angles were computed from the inlet to

Feature	Parameters					
Walls	Ceiling	Floor	Right	Left	Front	Back
Inlet	$1/r$	$\theta(0,\ \pi/2)$	$\phi(0, 2\pi)$			
Outlet	$1/r$	$\theta(0,\ \pi/2)$	$\phi(0, 2\pi)$			
Output	r	$\theta(0,\ \pi)$	$\phi(0, 2\pi)$			

Table 1: Table showing the inputs and outputs of the spherical SPM

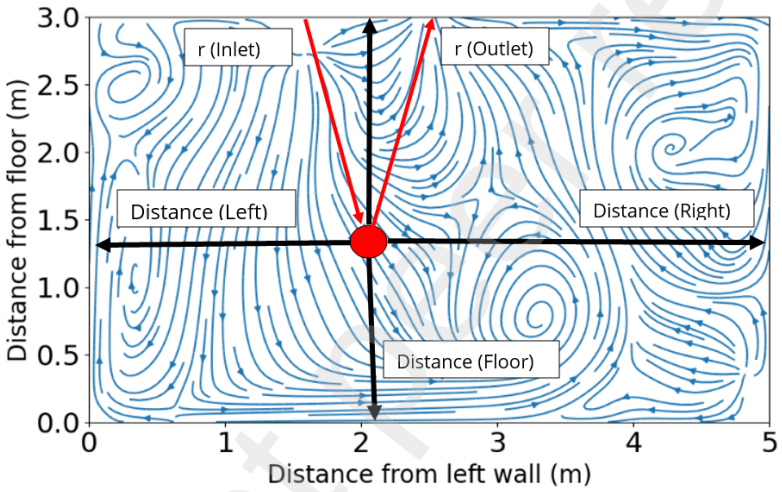

Figure 1: The input features using Cartesian and Spherical coordinate systems

the reference point and from the reference point to the outlet, as depicted in Figure 1. These angles were represented in radians, as degree variations have a much larger magnitude, and the ANN would assign greater significance to the angular input than the velocity magnitude input.

3.3. Loss Function

A key element in supervised learning algorithms is calculating an error measure —often called loss function— through the training process. The error is propagated backward through the network via a process known as backpropagation, in order to adjust the weights and reduce the error over time, iteratively. A common loss function used in ML applications is the Mean Absolute Error (MAE), which quantifies the average magnitude of

errors between predicted values ($\hat{y}i$) and actual values (yi) [38], as shown in Equation 6.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{6}$$

Despite the wide application of MAE, it is not suitable for ML applications, in which the output space involves features at different scales. In such conditions, the Mean Absolute Percentage Error (MAPE) is used, as shown in Equation 7, which assesses the average absolute percentage difference between predicted and actual values [39].

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}|\frac{y_i - \hat{y}_i}{y_i}| \tag{7}$$

Being a relative error metric, it treats all scales equally. However, it yields disproportionately large errors when dealing with very small actual values ($y_i << 1$). This means that small outputs will dominate the training process in physical systems with noise. For example, in the case of CFD simulation, using MAPE for training the SPM is inappropriate since it emphasizes points close to the boundary conditions (e.g., walls) where the airflow velocity is nearly zero. To remedy this limitation, the Mean Absolute Arctangent Error (MAAE) is introduced —shown in Equation 8— as an effective loss function for ML applications by evaluating the arc-tangent of the MAPE [40]. As a result, MAAE is confined in the range of $[0, \frac{\pi}{2}]$.

$$MAAE = \frac{1}{n}\sum_{i=1}^{n} arctan(|\frac{y_i - \hat{y}_i}{y_i + \epsilon}|) \tag{8}$$

The addition of ϵ on the denominator also serves to prevent computational errors when using this loss function. Although less intuitive than MAPE percentages, MAAE is advantageous because it is less sensitive to nearly zero outputs. In this research, MAAE is used as the loss function for training the SPM for CFD simulation due to its advantages over other more commonly used error measures.

4. Model Development Iterations

4.1. Initial Development

In an unoccupied room with no natural or mechanical ventilation, airflow tends to be uniform and stagnant, with zero velocity throughout the space. However, the introduction of ventilation disrupts this uniformity, leading to airflow patterns that eventually settle into a steady-state equilibrium. As a result, the velocity at any point within the room becomes a function of its position relative to the key features within the space (e.g., walls, HVAC inlets, and outlets).

Our initial results confirm that representing the spatial features of the space in the Cartesian coordinate system, as suggested in some earlier studies [41], cannot provide a generalizable SPM. Table 2 presents the input features utilized for developing the initial model.

Walls	Ceiling	Floor	Right	Left	Front	Back
Inlet	Δx	Δy	Δz	$r = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2}$		
Outlet	Δx	Δy	Δz	$r = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2}$		
Output	vx	vy	vz			

Table 2: Input and output features of the SPM defined in the Cartesian system

The positions relative to HVAC inlets and outlets were represented using both coordinate differentials (Δx, Δy, Δz) and absolute distances (r). This dual representation is necessary because HVAC air jets typically exhibit relatively consistent velocities directly beneath them, which then diminish rapidly as one moves away from the source across the room. The SPM outputs comprise the separated x, y, and z components of velocity (vx, vy, vz) as they fully describe the velocity flow field within the room.

Initially, the Mean Absolute Error (MAE), as a common loss function in several ML applications (see Equation6), served as the loss function for training the SPM. However, due to the significantly larger scale of the velocity direction angles as compared to its magnitude, the model tended to prioritize minimizing angle errors. Consequently, this led to velocity magnitude predictions becoming random and inaccurate. To address this issue, the Mean Absolute Percentage Error (MAPE), illustrated by equation 7, was introduced. However, when predicting velocities as low as 1×10^{-7}, the

practically negligible velocity errors may result in percentage errors of several thousand percent and dominate the learning process. As a solution, the Mean Absolute Arctangent Error (MAAE), represented by equation 8, was adopted.

Through the initial model development stage, several architectures of ANN were tested to develop a general SPM for the entire room, and the performance of each model was investigated by developing error heat maps throughout the room. The analysis of error heat maps revealed that velocity vector orientations (θ and ϕ) and magnitudes were highly sensitive to changes in position, which was almost impossible for a singular ANN to predict velocities in the entire room and hence resulted in large errors. To prevent large errors, the entire room space was segmented into four flow regimes, which led to a significant decrease in the average error for all segments. Different physical properties govern different flow regimes; hence an ANN specialized in predicting each regime will yield the best results. Flow regime change can be caused by many factors including Reynolds number, surface roughness of each wall, and geometry changes. However, for simplicity, classification will be based on the Reynolds number using Equation 1. Table 3 summarizes the initial model development results, and the highlights specify the chosen settings for the final SPM.

Number of Hidden Layers	Coordinate system	Learning Rate	Number of ANNs	Loss Function	MAPE (%)
3	Cartesian	0.001	1	MAPE	5489
3	Cartesian	0.001	1	MAAE	187
3	Spherical	0.001	1	MAAE	72
3	Spherical	0.001	4	MAAE	21
7	Spherical	0.001	4	MAAE	12
7	Spherical	0.0017	4	MAAE	1.75

Table 3: Summary of the SPM performance metrics through successful initial iterations

Changing the loss function from MAPE (Mean Absolute Percentage Error) to MAAE (Mean Arctanget Absolute Error) had the largest proportional decrease in percentage error, reducing it from 5486% to 187%. Switching the coordinate system in the data from Cartesian to spherical further reduced the MAPE from 187% to 72%.

4.2. Final Model Architecture

Based on the findings of initial model development efforts, the final architecture of the SPM is developed by integrating coordinate transformation, two-layered ensemble learning (i.e., flow segmentation and multiple velocity predictor DNNs), and the MAAE loss function as presented in Figure 2.

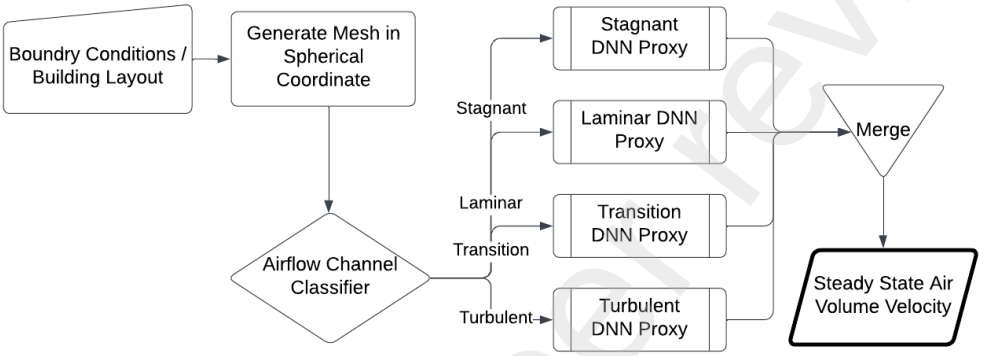

Figure 2: Final SPM Architecture for Predicting CFD Simulation Results

The airflow classifier, which is a simple multi-layer perception ANN, segments the space into four flow regimes; then, the input features for each segment are fed into a deep neural network (DNN) that predicts the air flow velocity.

The architecture of the flow segmentation ANN was optimized using grid-search, resulting in a model with four hidden layers, each with 64, 128, 256, and 512 nodes, respectively, and with a softmax activation function at the output layer. The performance of the flow segmentation ANN is presented in Table 4.

	Turbulent	Transition	Laminar	Stagnant
Percentage of cells (%)	2.53	7.23	88.21	2.03
Min-Velocity (m/s)	0.021	0.01	0.0002	0
Min Reynolds Number	4000	2000	100	0
Prediction Accuracy (%)	99.13	96.41	98.65	86.24

Table 4: Performance metrics of the flow segmentation ANN

The flow segmentation ANN was able to assign flow regimes to each point in the room with a weighted average accuracy of 98.2%.

The velocity predictor DNNs are designed with seven hidden layers with node counts increasing in a pyramidal arrangement: 32, 64, 128, 256, 128, 64, and 32 nodes in each respective layer. This pyramidal architecture enables the extraction of increasingly abstract features at intermediate layers, which are then distilled down to the three output nodes, proving effective for regression tasks [42].

Both models, the flow segmentation ANN and the velocity predictor DNNs, are trained using Adam Optimizer [43]. The optimal training parameters (i.e., learning rate) were acquired by hyperparameter optimization, where adjusting the learning rate from 0.001 to 0.0017 could significantly improve the accuracy of the SPM.

4.3. Model Training and Testing

The training data was generated from two rooms on ANSYS Fluent, and a training, validation, and testing split of the data was created at 70%, 15%, and 15%. The model was trained and then tested on the unseen test set to validate its ability to predict unseen data. The MAAE for each split was recorded and converted into MAPE, which allows a more intuitive analysis of results, as presented in Table 5.

Performance/Flow Segment	**Turbulent**	**Transition**	**Laminar**	**Stagnant**
Training MAAE	0.0250	0.0218	0.0551	0.5881
Training MAPE (%)	2.50	2.18	5.52	66.30
Test MAAE	0.0190	0.0184	0.0458	0.5820
Test MAPE (%)	1.90	1.84	4.58	65.80
Inference Time per room(s)	0.0352	0.0496	0.6414	0.0193

Table 5: The evaluation metrics for Smart Proxy Model

The weighted average MAPE of the SPM on the test data is $5.56 \approx 5\%$, which encompasses both the inaccuracies associated with the airflow segmentation ANN and the velocity predictor DNNs.

5. Results

The flow segmentation ANN classifies the entire room space into four flow regimes. Figure 3(a) shows a cut-plot of the room at the location of

the air-mass flow inlet. This diagram shows the physical location from which each airflow segment is extracted. The turbulent segment is mainly composed of flow below the mass-flow inlet and outlet vents, and the transition region typically outlines any turbulent regions or sharp changes in geometry. Stagnant flow predominantly occurs due to the boundary layer of each wall. Next, the velocity predictor DNNs predict the magnitude and direction of airflow velocity at every given location in the room, as shown in Figure 3(b).

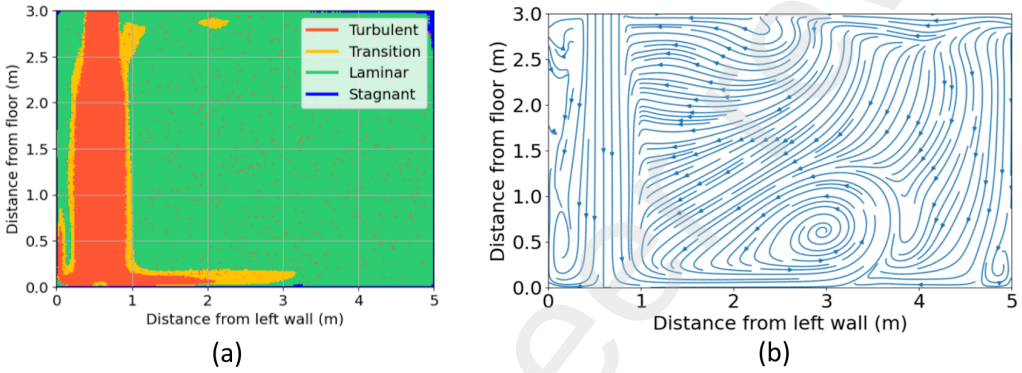

Figure 3: Graphic showing (a) the fluid flow regimes in a cut section of a room with an HVAC system, and (b) airflow velocity predictions by DNNs

The testing results presented in Table 5 illustrate that the SPM predicts turbulent and transitional flow to a test MAPE of less than 2%. The accuracy was very high as turbulent flow is generally clustered in very close proximity to the HVAC inlets and outlets and thus is highly influenced by the input features. These results are very promising as flow from these regimes can have a velocity magnitude of up to 1000 times the magnitude of a velocity in laminar flow and thus will affect the entire velocity flow field much more. On the other hand, the laminar flow model has a relatively higher test error of 6.8%; this discrepancy could be attributed to the dispersed nature of laminar flow throughout the room, leading to more complex flow patterns that are inherently harder to predict accurately. The stagnant flow model performs very poorly, with a MAPE of 68%. At very low airflow velocity, even random variations in conditions can lead to large relative changes in angle and velocity, which are impossible to account for in an SPM. On a positive point, the test MAPE for the stagnant flow model is consistent with the training MAPE, demonstrating that the model performs well with unseen data. Furthermore, since stagnant flow is only 2.03% of the entire room space

and represents nearly zero airflow velocity, it is not a significant output of the SPM expected. Figure 4 presents the heatmap of the error in the results in a combination of standing and bird-eye view angles (a horizontal cross-section of the room).

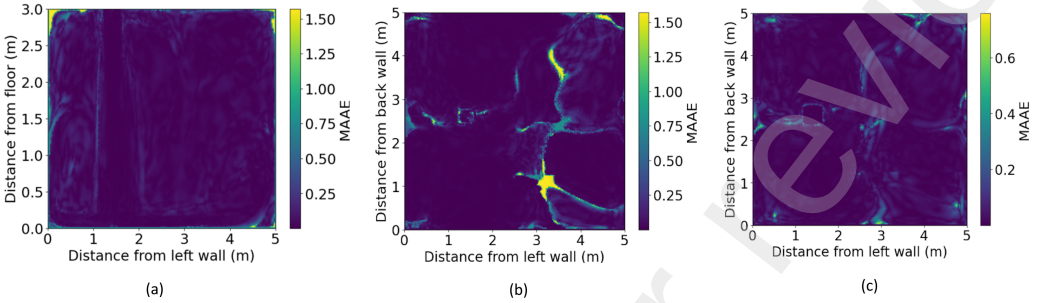

Figure 4: Heath of the SPM results in (a) standing view for the airflow velocity magnitude; (b) bird-eye view for the azimuthal angle; and (c) bird-eye view for the polar angle

The error heatmaps shown in Figure 4 indicate that the stagnant flow regime dominates the errors of the SPM. As Figure 4(a) illustrates in further detail, these errors are predominantly occurring in the proximity of the boundary conditions (i.e., walls and the ceiling), indicating the lower significance in real-world applications of the SPM. Furthermore, Figure 4(b) shows more significant errors in the azimuthal velocity angle around the HVAC outlet, indicating poor model performance when the airflow switches direction sharply.

5.1. Discussion

During the optimization stage of the SPM, the SPM was tested with the same inputs but only tasked with predicting the velocity magnitude, which led to very poor results. This indicates that the neural network weights necessary for forecasting azimuthal and polar angles are crucial in accurately predicting velocity magnitude; this highlights the importance of a complete output basis within SPM models. Indoor airflow is heavily governed by eddies and swirls generated by the deflection of air from walls and edges, and as such, the velocity orientation tells the SPM whether the airflow will be stopped by an obstacle or accelerated. The model was initially trained and tested on one simulation, and then the training set was expanded to include two simulations. As the dataset was expanded, the model training

and testing accuracy increased significantly, suggesting that it scaled well to a more expansive dataset.

A major challenge that the SPM faces is predicting the azimuthal and polar angles in the stagnant flow regime. One potential solution to address this issue is to refine the training data for the stagnant regime within the SPM. By smoothing out the angle changes within the stagnation regime training data, only the predominant flow directions would prevail, which would reduce anomalous errors and stabilize training. Moreover, further discrepancy appears evident between the free stream jet at the HVAC inlet and the surrounding flow, indicating a discontinuity between predictions made by turbulent and transition models. Given that the turbulent model yields the most accurate predictions, the SPM could gradually integrate its predictions from one model to influence the predictions at the boundary of the next model, allowing the SPM to build the flow field iteratively. Such a strategy can potentially mitigate errors across the transition, laminar, and stagnant flow regimes.

A key performance indicator of the SPM, besides the prediction accuracy, is the computational cost of the SPM. The computational cost of the two models is compared based on their simulation run-time. The SPM inference ran on a laptop with 9^{th} gen Intel Core i7 processor and 16GB of RAM, and the CFD simulation ran on Durham Universities Hamilton 8 Supercomputer with 384 CPU cores (2x AMD EPYC 7702) and 256GB RAM [44]. Although the computational power of the two systems is incomparable, we will compare the two models based on the simulation time rather than pure computational cost. The Inference time of the SPM for each flow segment is presented in Table 5. The total inference time for the entire room is $\approx 0.8s$ (including regime classification). In contrast, running equivalent CFD simulation on Hamilton8 Supercomputer [44] takes $\approx 5\,hrs$ to complete, which indicates that the SPM could save 99.996% of the computation time. Such a significant reduction in simulation time can realize new applications of CFD in the building engineering, including real-time control of HVAC systems based on CFD data or the application of heuristic optimization techniques for finding optimal building layouts.

6. Conclusions and Future Developments

This research highlights the effectiveness of a deep learning-based smart proxy model (SPM) in predicting indoor velocity flow fields, replacing CFD

simulation in those problems in which accelerated simulation is needed for engineering applications. A novel SPM with a two-layered, ensemble learning architecture is proposed in this research, which has proven effective in extracting information from the building and replicating the results of CFD simulation at a fraction of the computational cost. The proposed architecture consists of a simple ANN used to segment the airflow field into four flow regimes: turbulent, transition, laminar, and stagnant, which were initially distinguished based on Reynold's number. Four DNNs are placed at the second layer to predict the airflow velocity in every airflow segment. The performance of the proposed SPM, measured by the mean absolute percentage error (MAPE), on the testing (unseen) data 1.9% for turbulent, 1.84% for transitional, 4.58% for laminar, and 65.80% for stagnant. The weighted average MAPE for the entire room is $\approx 5\%$ for the entire room. Crucially, the SPM completes simulations for a typical medium-sized office within less than 0.8 seconds on a laptop PC, contrasting the CFD simulation run for the same office that takes ≈ 5 hours on a supercomputer with 384 CPU cores and 256GB of RAM. As a result, the proposed SPM could determine the airflow velocity in the case study with 95% accuracy at only 0.004% of the time required for numerical simulation. Accelerated yet accurate CFD simulation results can introduce new use cases for CFD in building engineering applications.

Our proposed SPM is still based on a context-dependent modeling architecture, meaning it best performs on the same or similar spaces for which it is trained. For more generalizable SPMs, a diverse range of CFD data must be generated and potentially released as standard CFD training datasets, such as the ones available for computer vision models. This can be achieved in the future using scheme files that can automate the CFD simulation process and repeatedly generate new simulations for different boundary conditions. Furthermore, the generalizability of the SPM could be improved by using physics-informed neural networks, which constrain the predictions of the SPM to those that align with physical principles. These additions may make the SPM slightly more computationally expensive but could also offer substantial increases in their generalizability and accuracy.

References

[1] S. J. Ra, J.-H. Kim, C. S. Park, Real-time model predictive cooling control for an hvac system in a factory building, Energy and Buildings

285 (2023) 112860. doi:10.1016/j.enbuild.2023.112860.

[2] ASHRAE, Ventilation for acceptable indoor air quality, Tech. Rep. ISSN 1041-2336, ASHRAE (2016).

[3] X. Zhang, J. Wu, L. M. Smith, X. Li, O. Yancey, A. Franzblau, J. T. Dvonch, C. Xi, R. L. Neitzel, Monitoring sars-cov-2 in air and on surfaces and estimating infection risk in buildings and buses on a university campus, Journal of Exposure Science & Environmental Epidemiology 32 (5) (2022) 751–758.

[4] A. A. Aliabadi, S. N. Rogak, K. H. Bartlett, S. I. Green, Preventing airborne disease transmission: review of methods for ventilation design in health care facilities, Advances in preventive medicine 2011 (1) (2011) 124064.

[5] G. Calzolari, W. Liu, Deep learning to replace, improve, or aid cfd analysis in built environment applications: A review, Building and Environment 206 (2021) 108315. doi:10.1016/j.buildenv.2021.108315.

[6] B. Birnir, L. Angheluta, The build-up of droplet/aerosols carrying the sars-cov-2 coronavirus, in confined spaces, medRxiv (2020). doi:10.1101/2020.08.11.20173195.

[7] C. Ren, H. Yu, J. Wang, H.-C. Zhu, Z. Feng, S.-J. Cao, Zonal demand-controlled ventilation strategy to minimize infection probability and energy consumption: A coordinated control based on occupant detection, Environmental Pollution 345 (2024) 123550. doi:https://doi.org/10.1016/j.envpol.2024.123550.

[8] Bringing embodied carbon upfront.
URL https://worldgbc.org/advancing-net-zero/embodied-carbon/

[9] W. Wu, B. Wang, A. Malkawi, N. Yoon, Z. Sehovic, B. Yan, A method toward real-time cfd modeling for natural ventilation, Fluids 3 (4) (2018). doi:10.3390/fluids3040101.
URL https://www.mdpi.com/2311-5521/3/4/101

[10] A.-M. Sigounis, C. Vallianos, A. Athienitis, Model predictive control of air-based building integrated pv/t systems for optimal hvac integration, Renewable Energy 212 (2023) 655–668. doi:10.1016/j.renene.2023.05.059.

[11] P. R. Pandey, B. Dong, Prediction of window opening behavior and its impact on hvac energy consumption at a residential dormitory using deep neural network, Energy and Buildings 296 (2023) 113355. doi:10.1016/j.enbuild.2023.113355.

[12] M. Younis, M. T. Kahsay, G. T. Bitsuamlak, Bim-cfd-based thermal analysis for northern buildings on permafrost, Journal of Cold Regions Engineering 37 (4) (2023) 04023019. doi:10.1061/JCRGEI.CRENG-644.

[13] V. Chinde, K. Woldekidan, Model predictive control for optimal dispatch of chillers and thermal energy storage tank in airports, Energy and Buildings 311 (2024) 114120. doi:10.1016/j.enbuild.2024.114120.

[14] P. E. Saarinen, P. Kalliomäki, J. W. Tang, H. Koskela, Large eddy simulation of air escape through a hospital isolation room single hinged doorway—validation by using tracer gases and simulated smoke videos, PloS one 10 (7) (2015) e0130667. doi:10.1371/journal.pone.0130667.

[15] H. Cho, D. Cabrera, S. Sardy, R. Kilchherr, S. Yilmaz, M. K. Patel, Evaluation of performance of energy efficient hybrid ventilation system and analysis of occupants' behavior to control windows, Building and Environment 188 (2021) 107434. doi:10.1016/j.buildenv.2020.107434.

[16] N. Izadyar, W. Miller, B. Rismanchi, V. Garcia-Hansen, Impacts of façade openings' geometry on natural ventilation and occupants' perception: a review, Building and Environment 170 (2020) 106613. doi:10.1016/j.buildenv.2019.106613.

[17] A. Roetzel, A. Tsangrassoulis, U. Dietrich, S. Busching, A review of occupant control on natural ventilation, Renewable and Sustainable Energy Reviews 14 (3) (2010) 1001–1013. doi:10.1016/j.rser.2009.11.005.

[18] G. Vita, D. Woolf, T. Avery-Hickmott, R. Rowsell, A cfd-based framework to assess airborne infection risk in buildings, Building and environment 233 (2023) 110099. doi:10.1016/j.buildenv.2023.110099.

[19] T.-W. Tsang, K.-W. Mui, L.-T. Wong, Computational fluid dynamics (cfd) studies on airborne transmission in hospitals: A review on the research approaches and the challenges, Journal of Building Engineering 63 (2023) 105533. doi:10.1016/j.jobe.2022.105533.

[20] N. Izadyar, W. Miller, Ventilation strategies and design impacts on indoor airborne transmission: A review, Building and Environment 218 (2022) 109158. doi:10.1016/j.buildenv.2022.109158.

[21] J. Hang, Y. Li, W. Ching, J. Wei, R. Jin, L. Liu, X. Xie, Potential airborne transmission between two isolation cubicles through a shared anteroom, Building and Environment 89 (2015) 264–278. doi:10.1016/j.buildenv.2015.03.004.

[22] R. Kumar, C. M. Patel, A. K. Jana, S. R. Gopireddy, Prediction of hopper discharge rate using combined discrete element method and artificial neural network, Advanced Powder Technology 29 (11) (2018) 2822–2834. doi:10.1016/j.apt.2018.08.002.

[23] C. T. Jayasundara, H. Zhu, Predicting liner wear of ball mills using discrete element method and artificial neural network, Chemical Engineering Research and Design 182 (2022) 438–447. doi:10.1016/j.cherd.2022.04.013.

[24] D. Panchigar, K. Kar, S. Shukla, R. M. Mathew, U. Chadha, S. K. Selvaraj, Machine learning-based cfd simulations: a review, models, open threats, and future tactics, Neural Computing and Applications 34 (24) (2022) 21677–21700. doi:10.1007/s00521-022-07838-6.

[25] E. Motaei, T. Ganat, Smart proxy models art and future directions in the oil and gas industry: A review, Geoenergy Science and Engineering 227 (2023) 211918. doi:10.1016/j.geoen.2023.211918.

[26] A. Aboaba, Y. Martinez, S. Mohaghegh, M. Shahnam, C. Guenther, Y. Liu, Smart proxy modeling application of artificial intelligence & machine learning in computational fluid dynamics (7 2020). doi:10.2172/1642460.
URL `https://www.osti.gov/biblio/1642460`

[27] A. Shahkarami, , S. Mohaghegh, Applications of smart proxies for subsurface modeling, Petroleum Exploration and Development 47 (2) (2020) 400–412. doi:https://doi.org/10.1016/S1876-3804(20)60057-X.

[28] C. S. W. Ng, A. Jahanbani Ghahfarokhi, M. Nait Amar, O. Torsæter, Smart proxy modeling of a fractured reservoir model for production optimization: implementation of metaheuristic algorithm and probabilistic application, Natural Resources Research 30 (2021) 2431–2462.

[29] R. Syah, M. Elveny, M. K. Nasution, V. V. Ponkratov, M. Y. Kuznetsova, A. L. Poltarykhin, M. Babanezhad, Numerical investigation of nanofluid flow using cfd and fuzzy-based particle swarm optimization, Scientific Reports 11 (1) (2021) 20973. doi:10.1038/s41598-021-00279-6.

[30] P. Xu, M. Babanezhad, H. Yarmand, A. Marjani, Flow visualization and analysis of thermal distribution for the nanofluid by the integration of fuzzy c-means clustering anfis structure and cfd methods, Journal of Visualization 23 (2020) 97–110. doi:10.1007/s12650-019-00614-0.

[31] S. A. Yousof Haghshenas, Mohammad Emami Niri, R. A. Kolajoobi, Developing grid-based smart proxy model to evaluate various water flooding injection scenarios, Petroleum Science and Technology 38 (17) (2020) 870–881. arXiv:https://doi.org/10.1080/10916466.2020.1796703, doi:10.1080/10916466.2020.1796703.
URL https://doi.org/10.1080/10916466.2020.1796703

[32] X. Dong, Z. Yu, W. Cao, Y. Shi, Q. Ma, A survey on ensemble learning, Frontiers of Computer Science 14 (2020) 241–258. doi:10.1007/s11704-019-8208-z.

[33] M. Bussov, J. Nättilä, Segmentation of turbulent computational fluid dynamics simulations with unsupervised ensemble learning, Signal Processing: Image Communication 99 (2021) 116450. doi:https://doi.org/10.1016/j.image.2021.116450.
URL https://www.sciencedirect.com/science/article/pii/S0923596521002150

[34] T. van Hooff, B. Blocken, 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 (2017) 148–165. doi:10.1016/j.buildenv.2016.12.019.

[35] P. J. LaNasa, E. L. Upp, Fluid flow measurement: A practical guide to accurate flow measurement, Butterworth-Heinemann, 2014.

[36] ASHRAE, Thermal enviornmental conditions for human occupancy, Tech. Rep. ISSN 1041-2336, ASHRAE (2020).
URL https://www.ashrae.org/technical-resources/bookstore/standard-55-thermal-environmental-conditions-for-human-occupancy

[37] E. Motaei, T. Ganat, Smart proxy models art and future directions in the oil and gas industry: A review, Geoenergy Science and Engineering 227 (2023) 211918. doi:https://doi.org/10.1016/j.geoen.2023.211918.
URL https://www.sciencedirect.com/science/article/pii/S2949891023005055

[38] Q. Wang, Y. Ma, K. Zhao, Y. Tian, A comprehensive survey of loss functions in machine learning, Annals of Data Science (2020) 1–26.

[39] A. de Myttenaere, B. Golden, B. Le Grand, F. Rossi, Mean absolute percentage error for regression models, Neurocomputing 192 (2016) 38–48, advances in artificial neural networks, machine learning and computational intelligence. doi:https://doi.org/10.1016/j.neucom.2015.12.114.
URL https://www.sciencedirect.com/science/article/pii/S0925231216003325

[40] S. Kim, H. Kim, A new metric of absolute percentage error for intermittent demand forecasts, International Journal of Forecasting 32 (3) (2016) 669–679. doi:https://doi.org/10.1016/j.ijforecast.2015.12.003.
URL https://www.sciencedirect.com/science/article/pii/S0169207016000121

[41] T. hu Zhang, X. yi You, Applying neural networks to solve the inverse problem of indoor environment, Indoor and Built Environment 23 (8) (2014) 1187–1195. doi:10.1177/1420326X13499596.

[42] S. Koziel, N. Çalık, P. Mahouti, M. A. Belen, Accurate modeling of antenna structures by means of domain confinement and pyramidal deep neural networks, IEEE Transactions on Antennas and Propagation 70 (3) (2021) 2174–2188. doi:10.1109/TAP.2021.3111299.

[43] D. Kingma, Adam: a method for stochastic optimization, arXiv preprint arXiv:1412.6980 (2014). doi:10.48550/arXiv.1412.6980.

[44] Durham University, Advanced research computing, https://www.durham.ac.uk/research/institutes-and-centres/advanced-research-computing/hamilton-supercomputer/systems/.