22nd International Conference on
Harmonisation within Atmospheric Dispersion Modelling for Regulatory Purposes
10-14 June 2024, Pärnu, Estonia

PHYSICS INFORMED NEURAL NETWORKS (PINNS) FOR RAPID CONTAMINATION DISPERSION PREDICTIONS

Zhijing Feng[1], Elisa Y.M. Ang[1], Tay Bee Kiat[2], Koh Wai Heng[2], Peng Cheng Wang[1]

[1]Engineering Cluster, Singapore Institute of Technology, 10 Dover Drive, Singapore 138683 Singapore
[2]DSO National Laboratories, 12 Science Park Drive, Singapore 118225

Abstract: This paper investigates the application of Physics-Informed Neural Networks (PINN) in predicting contamination dispersion, contrasting it with traditional Computational Fluid Dynamics (CFD). Notably, PINN demonstrates advantages when boundary conditions are unknown or frequently changing, a scenario often encountered in emergency response missions. By integrating appropriately positioned sensors, PINN can rapidly estimate dispersion maps. The study utilizes 2D case to analyze the strengths and limitations of sensor-augmented PINN versus traditional grid-based CFD. The findings reveal that for simple 2D flows, the convergence time of PINN is comparable to that of CFD. However, PINN's independence from critical boundary conditions such as the location of the source release and wind conditions renders it particularly attractive for specific applications.

Key words: *contamination dispersion prediction; physics informed neural network; sensors feedback; physics based model*

INTRODUCTION

Contamination dispersion prediction is vital for assessing disease spread, toxic chemical transmission, and indoor air pollutants (Mei, Zeng and Gong, 2021), and traditionally relies on computational fluid dynamics (CFD) (Li, Leong, Xu, Ge, Kang and Lim,2020). However, CFD requires intensive computation and precise initial conditions, limiting its use in rapid response scenarios (Kochkov, Smith, Alieva, Wang, Brenner and Hoyer, 2021). Recent studies have shown that Physics Informed Neural Networks (PINN) have emerged as a promising alternative (Raissi, Perdikaris, and Karniadakis, 2019). PINNs integrate physical laws into their loss functions, allowing for mesh-free solutions to complex equations. Despite the high training costs, recent studies have enhanced PINN's efficiency by combining it with strategic sensor data, improving speed and accuracy without detailed environmental data (Cai, Wang, Wang, Perdikaris, and Karniadakis, 2021). This paper discusses using PINN for rapid contamination prediction using minimal sensor inputs, highlighting its potential over traditional CFD in dynamic situations.

METHODOLOGY

Problem formulation

A basic 2D case was considered in this paper to demonstrate PINN's capability in contamination dispersion when compared to CFD. This scenario involves a simple 2D flow problem with a single release point and a rectangular bluff body placed at the center.

A simple 2D flow was set up where a source was placed at the boundary with a rectangular obstacle within the domain, as illustrated in Figure 1. All measurement shown in Figure 1 are non-dimensionalized with respect to the x dimension of the obstacle (0.2m). The release gas was assumed to have similar properties to air. The domain consists of a 5 m/s freestream air velocity in the x-direction. The problem was set-up

and solved in OpenFOAM: a C++ toolbox for the development of customized numerical solvers, and pre-/post-processing utilities for the solution of continuum mechanics problems, especially for computational fluid dynamics (CFD).

A 2D bluff-body of chord length D is placed in an oncoming freestream providing a Reynolds number Re_D=68,474. The boundaries of the domain span 10D, while an inlet width of 0.1D is positioned far upstream to introduce a contaminant source into the flow encountering the 2D bluff-body. Turbulence is modelled with the realizable k-epsilon model and the steady-state solution is hereby presented. Results focus on the contaminant distribution around the bluff-body. This CFD result from OpenFOAM is regarded as the reference solution which would then be used to evaluate the accuracy of the PINN solution.

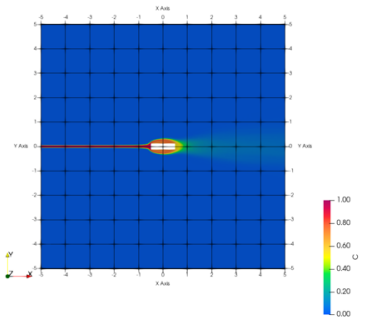

Figure 1. Normalized concentration distribution of the reference 2D flow problem formulation, in dimensionless units $(Re = 68{,}474)$.

PINN design

PINN design involves the construction of the PINN architecture that will be used to solve contamination prediction problems. This requires first the definition of PDEs to be solved. In this case, this will be the time-averaged Navier Stokes plus scalar transport equations, as shown from equation (1) to equation (4) for the two D, steady state case. In equation (1) to equation (4), Re refers to the Reynolds number and Pe refers to Peclet number.

$$u_x + v_y = 0 \tag{1}$$

$$uu_x + vu_y + p_x - \frac{1}{Re}\left(u_{xx} + u_{yy}\right) = 0 \tag{2}$$

$$uv_x + vv_y + p_y - \frac{1}{Re}\left(v_{xx} + v_{yy}\right) = 0 \tag{3}$$

$$uc_x + vc_y - \frac{1}{Pe}\left(c_{xx} + c_{yy}\right) = 0 \tag{4}$$

In typical PINN design, boundary conditions, often unknown, must be specified, leading to ill-posed problems difficult to solve with traditional methods. Recent advances, however, show that fully connected neural networks can address these issues using limited experimental or CFD data. Thus, this study does not specify boundary conditions and relies solely on sensor data as inputs for the PINN.

Figure 2 illustrates the process of optimization of PINN designed for contamination dispersion prediction. The neural network takes in x and y coordinates as inputs, and outputs four parameters x-velocity U, y-velocity V, pressure P and tracer concentration C. Note that these are the dimensionless form of velocities u and v, pressure p and concentration c respectively. The back propagation algorithm for the training of the network parameters depends on the model loss function, which includes the model loss related to the measured sensor values MSE_{sensor} and the loss function related to the governing equation MSE_f. The algorithm seeks to reduce these loss functions to force the neural network to produce outputs that are close to the measured values from the sensors, and at the same time satisfying the governing equations (equation (1) to equation (4)) at the determined points.

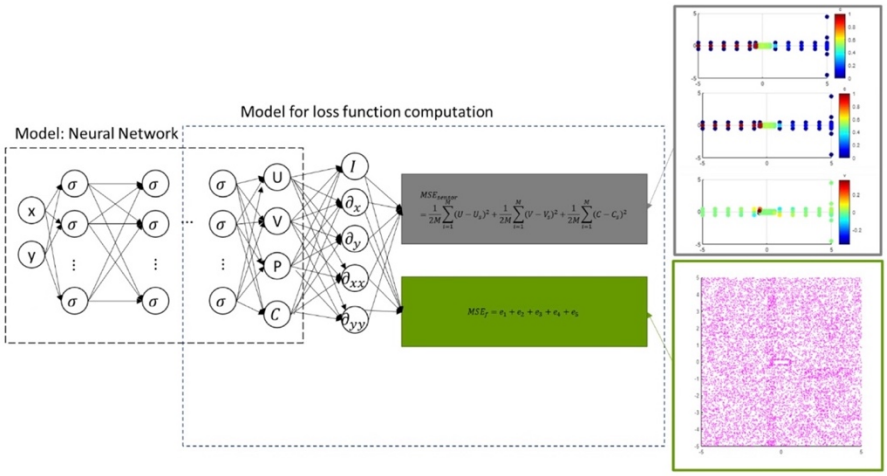

Figure 2. Illustration of the process of optimization of PINN for contamination dispersion prediction

MSE_{sensor} measures the mean square error of the output values from PINN (U, V, C) to that measured by the sensor (U_s, V_s, C_s). The formula to compute MSE_{sensor} is given in equation (5), where M is the number of sensor locations. MSE_f measures how well the output values from PINN (U, V, P, C) satisfy the governing equation written in equation (1) to equation (4) and is defined in equation (6). N is the total number of grid points used to check for consistency with the governing equations.

$$MSE_{sensor} = \frac{1}{2M}\sum_{i=1}^{M}\left(\left(U(x_i,y_i) - U_s(x_i,y_i)\right)^2 + \left(V(x_i,y_i) - V_s(x_i,y_i)\right)^2 + \left(C(x_i,y_i) - C_s(x_i,y_i)\right)^2\right) \tag{5}$$

$$MSE_f = e_1 + e_2 + e_3 + e_4 + e_5, where$$

$$e_1 = \frac{1}{2N}\sum_{i=1}^{N}\left(U_x(x_i,y_i) + V_y(x_i,y_i)\right),$$

$$e_2 = \frac{1}{2N}\sum_{i=1}^{N}\left(U(x_i,y_i)U_x(x_i,y_i) + VU_y(x_i,y_i) + P_x(x_i,y_i) - \frac{1}{Re}\left(U_{xx}(x_i,y_i) + U_{yy}(x_i,y_i)\right)\right),$$

$$e_3 = \frac{1}{2N}\sum_{i=1}^{N}\left(U(x_i,y_i)V_x(x_i,y_i) + VV_y(x_i,y_i) + P_y(x_i,y_i) - \frac{1}{Re}\left(V_{xx}(x_i,y_i) + V_{yy}(x_i,y_i)\right)\right),$$

$$e_4 = \frac{1}{2N}\sum_{i=1}^{N}U(x_i,y_i)C_x(x_i,y_i) + V(x_i,y_i)C_y(x_i,y_i) - \frac{1}{Pe}\left(C_{xx}(x_i,y_i) + C_{yy}(x_i,y_i)\right) \tag{6}$$

RESULTS

For the defined 2D flow problem, diffusion paths are primarily within the -1 < y < 1 range. During model training, 10k data points from a comprehensive CFD dataset were used, including many with negligible concentration values outside this range. For performance assessment, only data points within this range were used, featuring higher concentration and flow values, enhancing prediction accuracy.

Three sensor setups were examined: sensor configurations 1, 2, and 3. **Figure 3** illustrates the distribution of the basic 2D case in terms of the dimensions c, u, and v generated by OpenFOAM. **Figure 4** displays the corresponding results for three sensor cases. In all three sensor cases, a long strip sensor is placed on the far right, and sensors are placed around obstacles in the center. Sensor configuration 1 has only a few

sensors placed around the obstacle, while sensor configuration 2 and 3 have a larger number of sensors. Compared to sensor configuration 2, sensor configuration 3 also has sensors placed along the path of contamination diffusion.

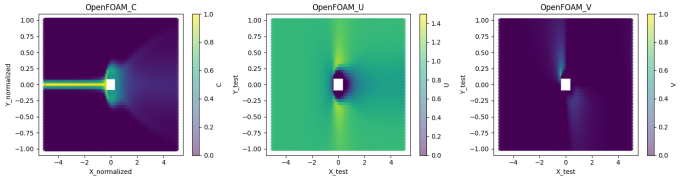

Figure 3. basic 2D case in terms of the dimensions c, u, and v generated by OpenFOAM

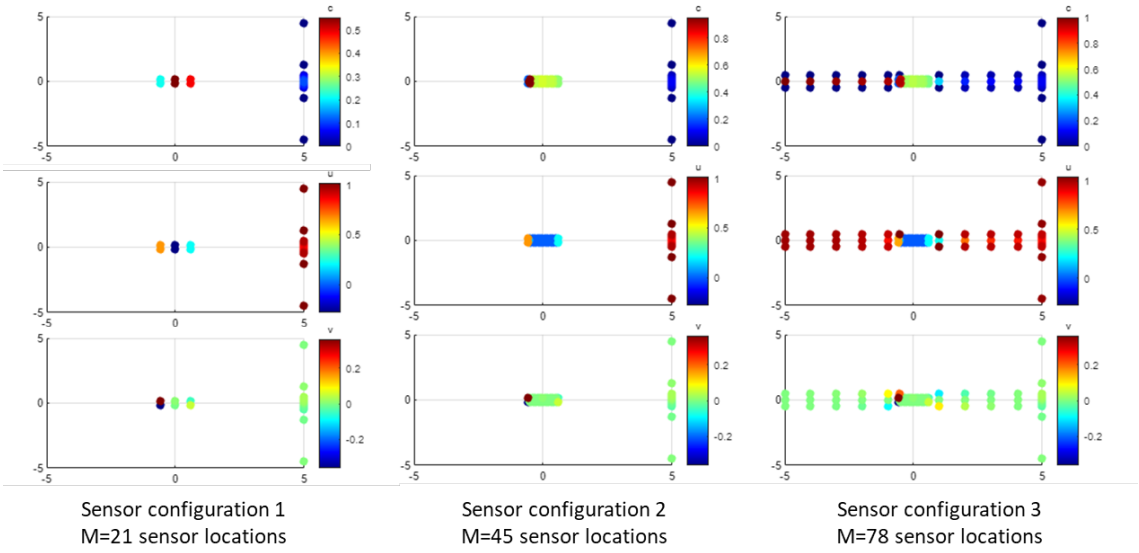

Figure 4. three fundamental sensor cases, three rows correspond to the distributions of c, u, and v, respectively

For evaluating the model's performance, it was discovered that a new metric needs to be developed. Initially, we utilized the conventional Mean Squared Error (MSE) for assessment. However, we observed its inadequacy in certain scenarios, where despite small MSE values, substantial discrepancies persisted between predicted and original images.

Hence, we opted to transform the images into a new colour space, LAB76, for comparative analysis. L, a, and b represent the 3 parameters which are used to separate out colours. L is for lightness, it goes from 0 to 100. A is red to green, the negative axis is green and the positive is red. B goes from yellow to blue, blue lies on the negative side and yellow on the positive one. Subsequently, we defined the formula: $LAB = \frac{1}{n}\sum_{i=1}^{n}\left(\sqrt[2]{\left(L_i - \widehat{L_i}\right)^2 + (a_i - \widehat{a_i})^2 + \left(b_i - \widehat{b_i}\right)^2}\right)$, and utilized "LAB" as the evaluation criterion.

Through extensive experimentation, we observed faster convergence in the velocity component u than in the concentration component c. Increasing training epochs can improve the accuracy for c but may cause overfitting, affecting the accuracy of u. A strategic approach is needed to enhance both metrics without overfitting.

Given the importance of concentration c in this context, we prioritized its accuracy over velocity u. The model was stopped after 2,000 epochs, achieving satisfactory predictions for c, u, and v without excessive training. Using sensor configuration 3, the model's loss showed minimal reduction after 2,000 epochs.

After establishing the foundational model, in pursuit of optimizing model performance, we adjusted the values of the following parameters: initialLearnRate, decayRate, numNeurons and numLayers. We

conducted combination of the selected parameters. For each parameter combination, models were trained for an equal number of epochs. We compared the final LABc, LABu, LABv, and LAB_sum of each model. The majority of parameter combinations yield similar results, but there are a few combinations that perform significantly worse. This discrepancy can lead to inaccurate model predictions. Therefore, we need to select the parameter combination that minimizes the value of "LAB". In this case, we selected the parameter combination resulting in the minimum LABc.

With the optimized parameters, the model ran with sensor configurations 1 to 3 and the results are shown in Figure 5 while sensor configuration 3 has the best prediction.

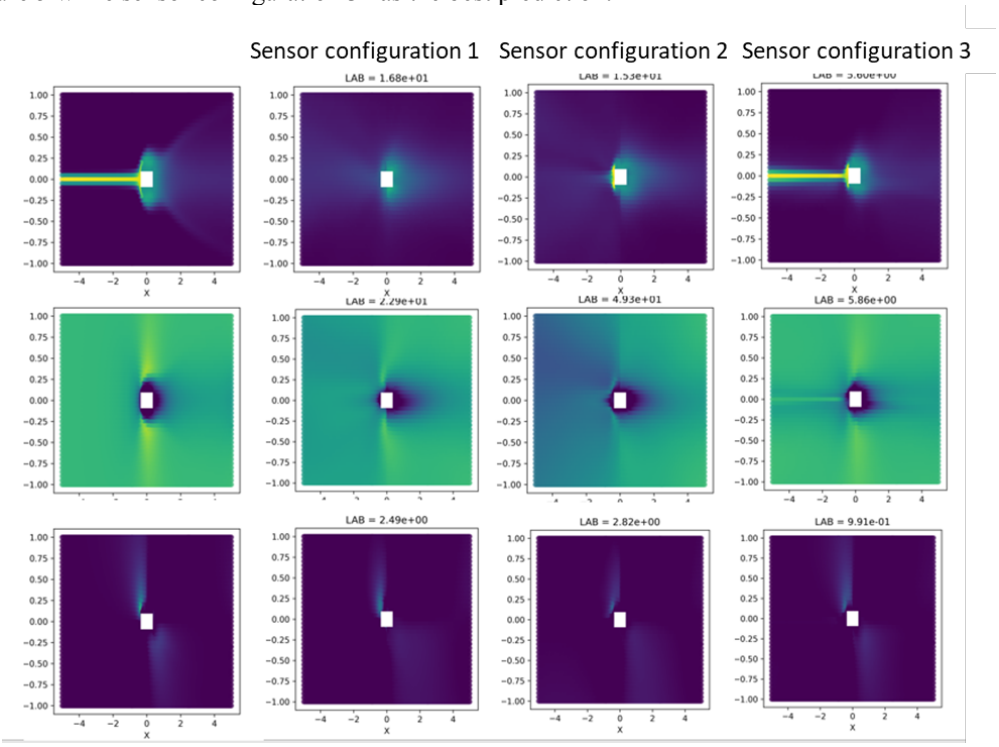

Figure 5. The first column depicts images from OpenFOAM, while columns 2 to 4 depict images predicted from PINN. Each row corresponds to the solution for c, u, and v, respectively

CONCLUSION

This contribution demonstrates the use of PINN for contamination dispersion prediction as an alternative for traditional CFD solvers. While CFD solvers require the specification of source release location and boundary conditions such as current windspeed and direction, PINN only requires feedback from strategic sensors in the domain. A careful investigation was carried out in 2D basic case as a feasibility study in this paper. Our results show that with strategically placed sensors, PINN can reproduce the 2D dispersion map to a good degree of accuracy when compared to that generated by CFD simulation. We have also conducted additional investigations based on basic 2D cases, such as scenarios involving the rotation and/or translation of obstacles. Due to space constraints, these results are not presented herein. Nevertheless, through judicious placement of sensors, favorable outcomes were attained for all these 2D cases when employing PINN as the modeling framework.

WORK TO BE CARRIED OUT IN THE NEXT PHASE

For the development of PINN approach, the following work would be the focus in the next phase. First, work has commenced to optimize sensor placement pattern that works for different 2D scenarios. Second, attempt to reduce the number of sensors to see how few can be used while still achieving good results from the model. Third, work on extending this methodology to both 3D (current work is limited to 2D domain), and also time-dependent solvers will be performed. Lastly, instead of point sensors, sensors that generated from the 2D integrated map can be considered.

REFERENCES

X. Mei, C. Zeng, and G. Gong, Building Simulation 2021 15:7, vol. 15, no. 7, pp. 1243–1258, Nov. 2021.
H. Li, F. Y. Leong, G. Xu, Z. Ge, C. W. Kang, and K. H. Lim, Physics of Fluids, vol. 32, no. 11, pp. 1–11, 2020.
D. Kochkov, J. A. Smith, A. Alieva, Q. Wang, M. P. Brenner, and S. Hoyer, Proc Natl Acad Sci U S A, vol. 118, no. 21, May 2021.
M. Raissi, P. Perdikaris, and G. E. Karniadakis, J Comput Phys, vol. 378, no. Part I, pp. 686–707, 2019.
S. Cai, Z. Wang, S. Wang, P. Perdikaris, and G. E. Karniadakis, J Heat Transfer, vol. 143, no. 6, 2021.