2021 NDIA GROUND VEHICLE SYSTEMS ENGINEERING AND TECHNOLOGY SYMPOSIUM
MODELING SIMULATION & SOFTWARE TECHNICAL SESSION
AUGUST 10-12, 2021 - NOVI, MICHIGAN

MACHINE LEARNING-BASED THERMAL AND FLOW SIMULATION ON HETEROGENEOUS PLATFORM FOR SIGNATURE PREDICTION

Seong Hyeon Hong[1], Alec House[2], Andrew L. Kaminsky[2], Nathan Tison[3], Yeefeng Ruan[3], Vamshi Korivi[3], Yi Wang[1*], Kapil Pant[2]

[1]Mechanical Engineering, University of South Carolina, Columbia, SC
[2]CFD Research Corporation, Huntsville, AL
[3]US Army DEVCOM Ground Vehicle Systems Center, Warren, MI

ABSTRACT

The data-driven machine learning (ML) method is developed to rapidly evaluate the thermal and flow fields of a ground vehicle and its neighboring environment at various conditions. The artificial neural network (ANN) is implemented as the ML model to evaluate the fields, while achieving equivalent accuracy as the CFD simulations. In order for ANN to precisely map a relationship between the simulation parameters and the solution field, the proper orthogonal decomposition (POD) technique is applied to reduce the dimension of the field variables. Consequently, the compressed data (i.e. modal coefficients) is selected as the target for the ANN. Once trained, POD reconstruction is performed on the ANN predicted modal coefficients to recover the CFD solution. The developed framework is tested at diverse sample sites, and the maximum mean absolute errors are found to be 0.41 K and 0.019 m/s for thermal and flow simulations, respectively, verifying the outstanding prediction performance.

Citation: S.H. Hong, A. House, A.L. Kaminsky, N. Tison, Y. Ruan, V. Korivi, Y. Wang, K. Pant, "Machine Learning-based Thermal and Flow Simulation on Heterogeneous Platform for Signature Prediction", In *Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium* (GVSETS), NDIA, Novi, MI, Aug. 10-12, 2021.

1. INTRODUCTION

Computational fluid dynamics (CFD) is a great tool for simulating heat transfer and turbulent flow to predict accurate thermal signatures of vehicles. Despite its high fidelity and accuracy, the computational costs can be prohibitive for various DoD applications, such as real-time simulation, thermal control development, and design optimization. As a consequence, machine learning (ML) for data dimension reduction and multi-physics field prediction has garnered significant research interest recently.

The most popular data dimension reduction methods are proper orthogonal decomposition (POD) and deep autoencoder (DAE). The POD method finds a set of orthogonal basis for representing a data set in an L^2-optimal sense to find the lower-dimensional approximations [1]. POD-based dimension reduction applications can be found in numerous literatures including miscible

DISTRIBUTION A. Approved for public release; distribution unlimited. OPSEC #: 5433
*Corresponding Author: Associate Professor, University of South Carolina, yiwang@cec.sc.edu

viscous fingering [2], mode-locked lasers [3], hyperelasticity and viscoelasticity material behaviors [4], bearingless motor [5], and fluid transport [6]. DAE is an unsupervised learning technique for artificial neural networks (ANNs) that reduces data dimensions by learning efficient data representations in the underlying nonlinear manifold. DAE is comprised of two parts: encoder and decoder, which are responsible for compression and reconstruction, respectively. Similar to the POD method, DAE has been used extensively in various engineering applications such as, induction motor [7], cyber security [8], power transmission line [9], wind turbine [10], bearing [11], and building energy [12]. Both methods have their own merits and disadvantages. POD is L^2-optimal and autonomous method and is easy to compute. However, it is a linear transformation, which requires a large number of modes to explain data variance. Otherwise, significant information loss could be induced due to POD modal truncation. On the other hand, DAE suffers from difficulties in implementation since it often requires large amount of training data and hyperparameter optimization. Therefore, either of the method must be carefully selected to meet the purpose of the application.

Reducing the data dimension offers a huge advantage in ML and data-driven modeling since it becomes easier to learn the relation between the data in the reduced subspace or manifold by removing unnecessary information that hinders model training, and alleviate the demanding requirement of data for training. Wang et al. [13] proposed a deep learning-based reduced order model (ROM) method that uses DAE to reduce the dimension of spatiotemporal data and then model the low-dimensional representation by either a linear autoregressive model or a nonlinear random vector functional-link (RVFL). Chen and Li [14] used DAE to extract the multisensor features to train the fault classification model by fine tuning the identified encoder weights. Wang et al. [15] presented a ROM method that combines the deep learning and the POD technique for transient ocean gyre and flow past a cylinder. The key idea was to use POD to reduce the data of field variables, such as velocity and pressure into the POD modal subspace, and then a long short-term memory network was developed to predict the modal coordinates at the next time step using those at the previous time steps.

Based on the existing efforts on dimension reduction and ML modeling, we propose a data-driven method to rapidly evaluate thermofluidic fields of a ground vehicle and its surroundings at various simulation conditions. The ML model, specifically ANN, can be trained to run rapidly while achieving almost equivalent accuracy as the CFD. To address the challenge associated with the large CFD meshes that preclude the ANN from learning the relationship between field variables and simulation parameters, a POD data reduction technique is implemented. POD modes are able to capture coherent physics underlying CFD simulation data and significantly reduce the dimension of CFD data. Thus, the ANN is able to predict the modal coefficients in the reduced domain. The predicted modal coefficients and the POD modes then can be used to reconstruct the CFD solution in the full computational domain. It is worth mentioning that POD is selected over DAE for two reasons: First, POD is an L^2-optimal method and can take in the input data of an extremely high dimensions (e.g., $\sim >10^6$ [16]), which, however, is intractable for DAE. Second, CFD data generation is computationally demanding and its availability can be limited. POD essentially reduces the data dimension through either singular value decomposition or eigenvalue decomposition, which is not a learning-based method, and hence, more suited for such a circumstance.

There are several advantages of the proposed method to combine the POD and ANN. First, through POD, the input-output mapping relationship is established between the simulation parameters and the modal coefficient (instead of the full field variables), and is much easier to model.

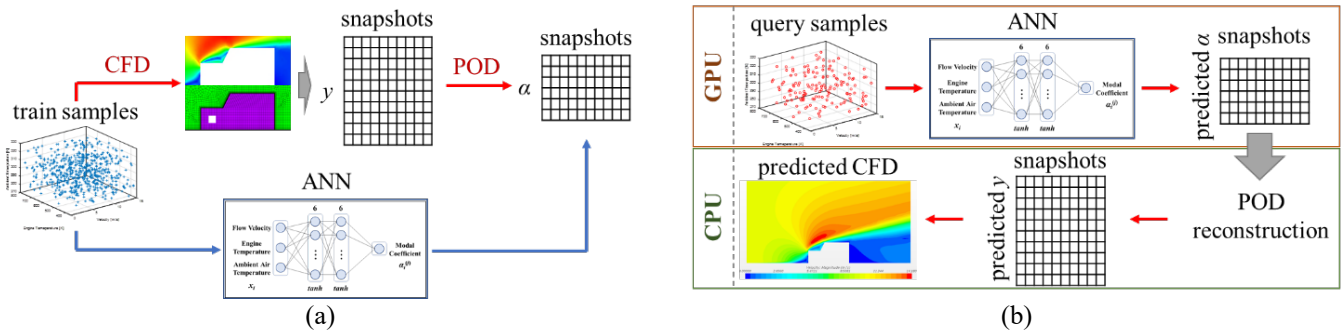


Figure 1: Flowchart of (a) ANN training, and (b) CFD prediction

As a result, the ANN-ROM can be trained with salient prediction accuracy and generality. Second, a lower dimension of outputs (i.e., modal coefficients) allows a concise ANN structure, reducing both the training and the inference time. Third, because of these two features, the training process is more robust to training failures or complications, such as overfitting and bias issues. Last, since there are a smaller number of weight parameters to train, the volume of the training data and corresponding CFD simulations can be reduced, and the computational load and resource usage can be alleviated.

In addition to the proposed ANN-ROM method, heterogeneous computing platform that incorporates both CPU and GPU is also developed for parallelized inference to enhance prediction throughput and for field applications.

The proposed research will enable ultrafast and accurate flow and thermal modeling and simulation, and enhance DoD's capabilities in concept evaluation, optimization design, and signature analysis and control of weapon systems.

The contributions of this research is summarized as follows: (1) the thermal analysis framework is integrated within one fully coupled CFD solver to generate required data for model training; (2) the POD, a mathematically rigorous data reduction technique, is applied to reduce the dimension of CFD data; (3) the ANN is trained to predict the modal coefficients in the reduced domain; and (4) the efficient heterogenous computing that utilizes both CPU and GPU is implemented for a rapid and large scale prediction.

2. METHODOLOGY

The detailed flowchart of ML model generation and prediction is depicted in Figure 1. At the model generation stage as shown in Figure 1a, samples are first created with the input parameter space, where CFD simulation is conducted to generate data for model training. For this work, the boundary conditions of the CFD analysis are selected as the input parameters, and the goal is to find the data-driven model that approximates the CFD simulation and is parameterized by the same boundary conditions used in the CFD model. The size of the CFD data is dictated by the number of cells in the CFD model. The solution field of CFD at one sample/simulation condition is vectorized into a single column vector y, where the components of a vector represent the field variable such as the temperature and velocity at a cell. The solution fields of all samples are horizontally concatenated in a matrix form, viz., a snapshot matrix, and thus, the rows and columns correspond to the field variables at CFD mesh cells and the solution snapshots of different samples (simulation parameters). In order to perform accurate CFD simulations, densely structured mesh is required, resulting in a large number of cells. In other words, often the CFD simulation yield a massive number of output parameters (corresponding to the solution variables at all mesh cells), which makes ML models difficult to predict, especially with a small

amount of data. To circumvent this issue, POD is directly performed on the solution snapshot matrix to find the modal representations of the output parameters denoted as α with a much lower dimension in the reduced domain than y. The ANN model is trained to establish a mapping between the CFD boundary conditions, viz, simulation parameters and α, the modal representation of the solution field.

The prediction stage is shown in Figure 1b. Once trained, the ANN can be used to predict α for different simulation parameters (i.e., boundary conditions). To increase the throughput, ANN prediction is performed on GPU. All the query samples and the ANN models are uploaded to the GPU memory. Once all the necessary data is transferred, modal coefficients are estimated concurrently. One caveat is that if the query size exceeds the capacity of GPU, not all the ANN inference could be conducted simultaneously. The predicted modal coefficients are transferred back to CPU as a matrix form. Within CPU, POD reconstruction is performed to actually predict the CFD solution fields. Once all the values of y is reconstructed, CFD solution is plotted by matching the output components to the corresponding cells. For ANN simulation in this research, a workstation with Intel(R) Core(TM) i9-9820X CPU @ 3.30 GHz and NVIDIA GeForce RTX 2080 Ti GPU is used. The details of POD and ANN applied in this work are elucidated in the following subsections.

2.1. Proper Orthogonal Decomposition

POD is a method to extract the dominant modes (i.e., basis vectors) and coefficients, denoise the data, and reduce the data dimension of discrete thermal flow fields from the training simulation. POD modes can be obtained by either the eigenvalue decomposition (EVD) or singular value decomposition (SVD) method. In this work, SVD is implemented as shown in equation (1).

$$\begin{gathered} Y = U\Sigma V^{\mathrm{T}} \\ U = [u_1 \cdots u_N] \\ V = [v_1 \cdots v_M] \end{gathered} \tag{1}$$

$$\Sigma_{ii} = \sigma_i$$

where $Y \in \mathbb{R}^{N\times M}$ is the snapshot matrix; $U \in \mathbb{R}^{N\times N}$ and $V \in \mathbb{R}^{M\times M}$ are orthogonal matrices; $\Sigma \in \mathbb{R}^{N\times M}$ is the matrix that has diagonal entries of singular values in a descending order and satisfies:

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{\min(N,M)} \geq 0. \tag{2}$$

Particularly in our work, N is equal to the size of CFD field (i.e., number of CFD cells) and M is equal to the number of train samples. By retaining the first k singular values and the corresponding U and V matrices, Y can be approximated as follows:

$$\begin{gathered} Y \approx \hat{Y} = U_r \Sigma_r V_r^{\mathrm{T}} = \sum_{i=1}^{k} \sigma_i u_i v_i^{\mathrm{T}} \\ U_r = [u_1 \cdots u_k] \\ \Sigma_r = diag(\sigma_1 \cdots \sigma_k) \\ V_r = [v_1 \cdots v_k] \\ k < M \end{gathered} \tag{3}$$

where k represents the dimension of the reduced domain. The matrix of modal coefficients A, is defined by:

$$A \in \mathbb{R}^{k\times M} = \Sigma_r V_r^{\mathrm{T}} = [\alpha_1 \cdots \alpha_M] \tag{4}$$

The matrix A is used as the target for ANN training instead of Y. Eq. (3) can also be used to reconstruct the CFD solution using the predicted modal coefficients. The reduced POD modes can be quantified by the ratio defined as:

$$E_{POD} = \frac{\sum_{i=1}^{k} \sigma_i^2}{\sum_{i=1}^{r} \sigma_i^2}. \tag{5}$$

where E_{POD} is the relative energy captured by the first k POD basis. k is selected as the minimum integer that satisfies the following inequality:

$$\begin{gathered} 1 - E_{POD} \leq \varepsilon. \\ 0 < \varepsilon < 1 \end{gathered} \tag{6}$$

For this work ε ranges between $1e^{-4}$ and $1e^{-6}$.

2.2. Artificial Neural Network

ANN is a ML technique that uses interconnected nodes (often denoted as neurons) from multiple layers to identify the system, purely based on the data. It has emerged as an effective and popular tool in identifying complex and nonlinear systems [17]. Various ANN-based system modeling applications can be found in the following literatures [18],[19],[20].

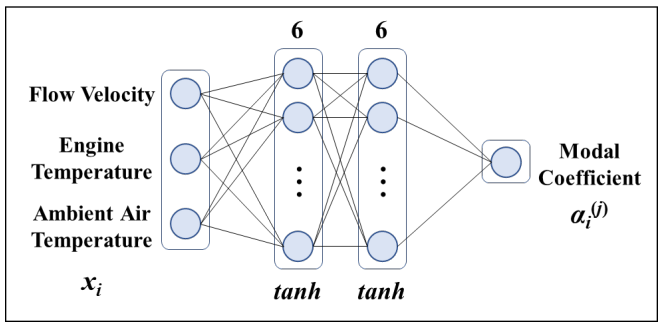


Figure 2: ANN architecture

The structure of the ANN model implemented in the present study is exhibited in Figure 2. There are two hidden layers and one output layer assigned to the network. In the figure, x_i and α_i represent boundary conditions (simulation parameters) and modal coefficients of CFD simulation, respectively. Subscript i denotes the sample number. Importantly, a separate ANN model is constructed and trained for each of the modal coefficient in the reduced domain. In other words, instead of building one multi-output model, which outputs all the components of α_i, multiple single-output models are trained, each predicting only one component of α_i. All ANN models share the same structure consisting of 6 hidden neurons for both hidden layers and hyperbolic tangent activation function.

The ANN model can be mathematically expressed as

$$\alpha_i^{(j)} = W_3 tanh(W_2 tanh(W_1 x_i + b_1) + b_2) + b_3, \quad (7)$$

where (W_1,W_2,W_3) and (b_1,b_2,b_3) are the weight parameters of the model; and j denotes the index of the modal coefficient targeted for prediction. The Levenberg-Marquardt optimization algorithm and mean squared error loss function are used in the training process. The maximum epoch for training process is set to be 500, and to prevent from overfitting, the training is halted when the validation error stops decreasing for 8 consecutive epochs. Due to the random initialization of the network weights, the entire training processes for all ANN models are repeated 5 times and the model with the highest test accuracy is selected for prediction/inference. Likewise, 600 and 150 data points are used for training and testing, respectively. Among the training set, 90% is used for training and 10% for validation.

3. CASE STUDIES

For demonstrating our computational tool, a 2-dimensional (2D) mock-up vehicle geometry is considered, which contains two domains: the solid vehicle (in gray) and the fluid flow domain (in blank but within the bounding box) as shown in Figure 3a. The far field is a rectangular shape with a length of 20 m and a height of 10 m. The specific dimensions of the vehicle are shown in Figure 3b. The square region of the size (0.5×0.5 m) within the vehicle represents the engine box. Equivalent CFD model is constructed to perform the thermal and flow simulation. Given the geometry, a 2D computational domain is developed. The mesh is

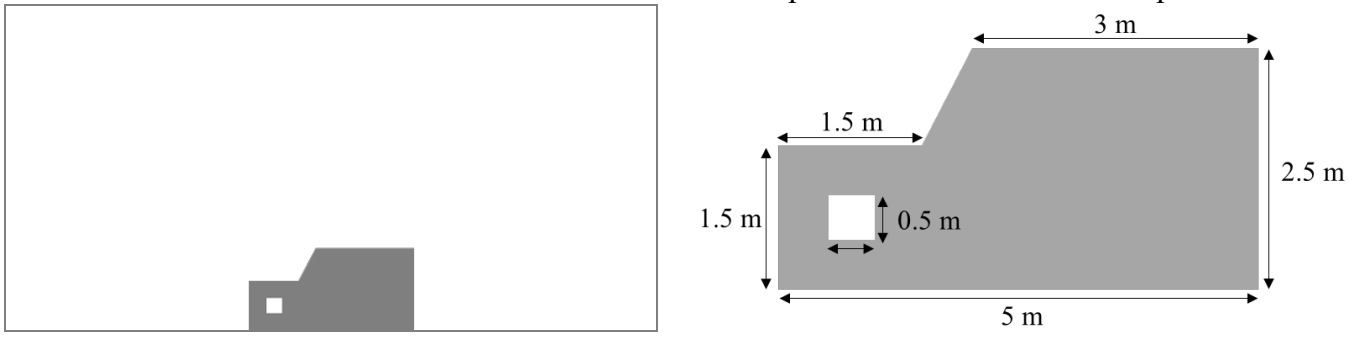
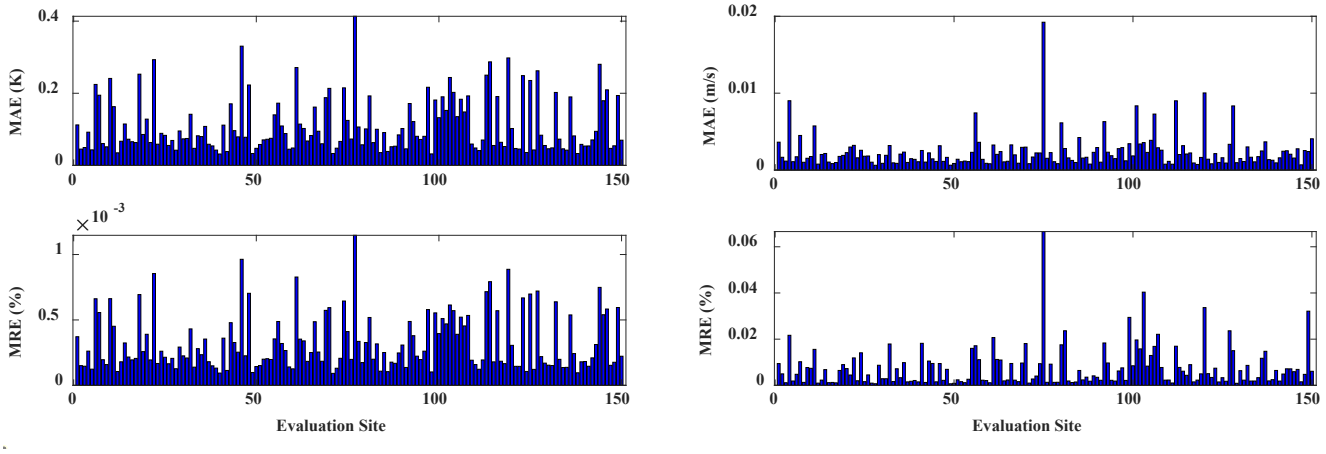


Figure 3: Geometry of CFD simulation: (a) entire field, and (b) vehicle dimensions



(a) (b)

Figure 4: CFD field prediction errors: (a) temperature and (b) magnitude of the flow velocity

constructed as an unstructured grid and made up of fluid and solid regions with 8,988 and 1,877 cells, respectively. A fine grid is applied at the boundary of the ground and the vehicle.

Three CFD boundary conditions: wind velocity, engine temperature, and ambient air temperature are selected as simulation parameters for this study; and their ranges are chosen as, 1-15 m/s, 350-800 K, and 273-323 K, respectively. The Latin hypercube sampling (LHS) method is employed to generate a large number of near-random samples that represent various combinations of the simulation parameters. As indicated in Section 2.3, two sets of LHS are generated for model training and verification/testing, and their sample sizes are, respectively, 600 and 150. In total 750 coupled CFD and conjugate heat transfer simulation (with different thermal and flow boundary conditions) are carried out, aiming to capture the effects of the simulation parameters on thermal and flow field variations within and around a vehicle.

In this work, OpenFOAM is employed to generate the simulation data. It is the free and open source CFD software and includes flow analysis solvers for various simulation scenarios, such as transient and steady state, compressible and incompressible, buoyancy, heat transfer, laminar and turbulent, and others. The advantage of OpenFOAM is its potential for fast running, automated simulations if a user develops a Linux script, such as Shell, Bash, TCL, and others. The combination of the script-based method and the high-performance computing (HPC) is applied to run multiple CFD simulations simultaneously, generating a library of CFD data for ML model training. The HPC node used in this research is Intel(R) Xeon(R) Platinum 8260 Processors @ 2.4 GHz with 192 GB of RAM and 48 cores. In this study, 2 nodes (96 cores) are utilized.

Conjugate heat transfer CFD simulations involving both thermal and flow are performed to observe the thermal field of the vehicle and its vicinity. The thermal solution is present in both the fluid and solid domains. Conversely, flow solution is only available in the flow domain. Through CFD, fluid velocity is computed in two orthogonal directions: horizontal and vertical, such that the sum of these two velocity vectors indicate the resultant flow velocity and direction. In summary, CFD solutions computed for given boundary conditions are divided into four groups: fluid temperature, solid temperature, horizontal velocity, and vertical velocity. POD is performed for each of the solution groups and the number of modal coefficients used is listed in

Table **1**. It is observed that for this particular problem, thermal data can be compressed more than the fluid data while preserving the similar

relative energy ratio. In total, 21 and 110 modal coefficients must be predicted to reconstruct thermal and flow solutions, respectively. Consequently, 131 different ANN models are trained.

Table 1: Number of modal coefficients identified for each CFD solution

		number of modal coefficients
thermal	fluid temperature	13
	solid temperature	8
flow	horizontal velocity	62
	vertical velocity	48

4. RESULTS WITH DISCUSSION

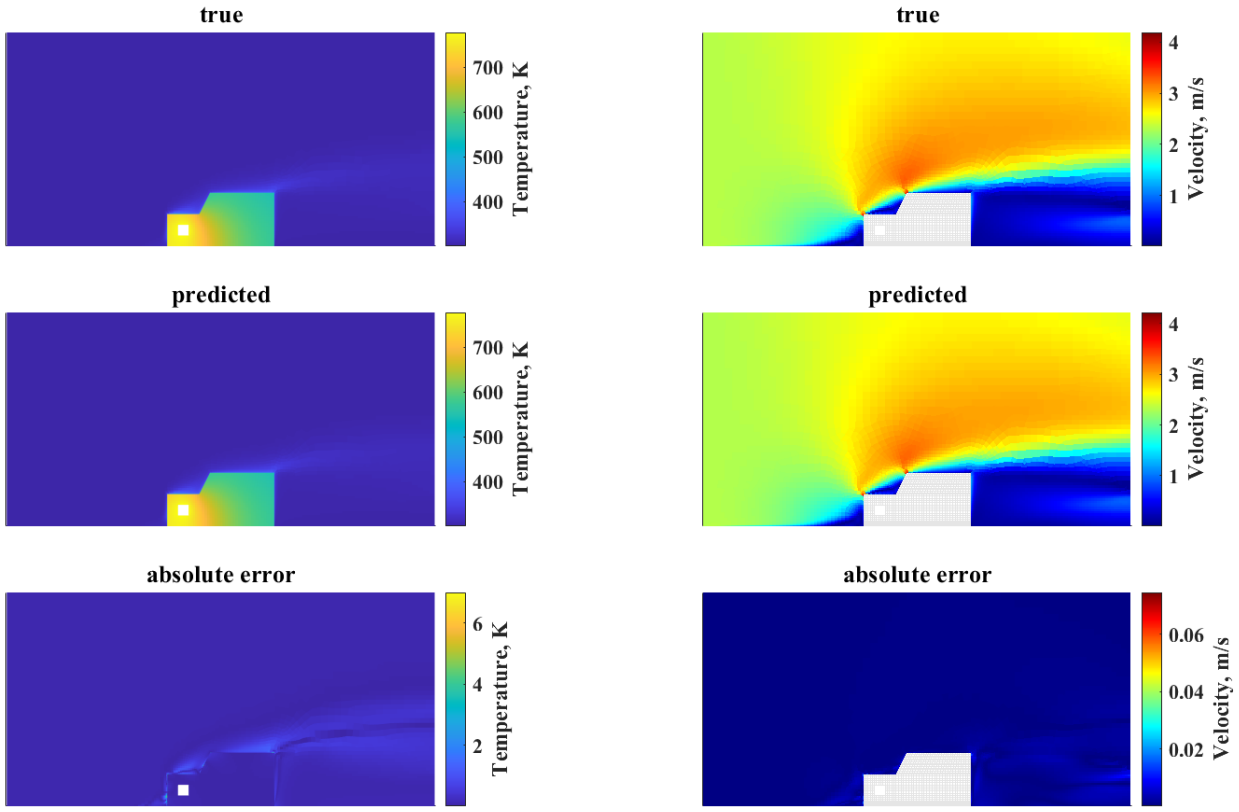
The mean absolute error (MAE) and mean relative error (MRE) of ML-based thermal and flow CFD field predictions are illustrated in Figure 4. All 150 test samples generated at different conditions are evaluated. The maximum mean absolute error (MAE) of the thermal signature prediction is found to be 0.41 K. This indicates that the average temperature error of the entire domain is less than or equal to 0.41 K, verifying the accuracy of the prediction. Since the temperature range varies based on the boundary conditions, it is equally important to check the mean relative errors (MREs). It is clear from Figure 4a that all MREs at different evaluation sites are strictly less than 0.0012%, indicating salient accuracy of the thermal prediction. Likewise, as seen from Figure 4b, the maximum MAE and MRE of the flow field prediction is 0.019 m/s and 0.067%, respectively, again indicating excellent accuracy of the proposed ML method. Furthermore, the average MAEs for all evaluation sites are only 0.11 K and 0.0024 m/s. It is important to note that the large difference exist between the average and the maximum errors. It has been confirmed that the sharp errors exist near the boundaries of the parameter space, where the number of training samples is appreciably smaller.

For more elaborate analysis of ML-based predictions, temperature and velocity contour plots are generated for two distinct evaluation sites: the 10^{th} and the 150^{th}. The boundary conditions of the two evaluation sites are presented in Table 2. The sample from evaluation site No. 10 exhibits large temperature difference between the engine and the ambient air, but low fluid velocity. On the other hand, the sample from evaluation site No. 150 possesses high fluid velocity but small temperature difference. The MAEs at 10 and 150 evaluations sites are computed to be 0.24 K and 0.07 K, respectively, for the temperature field, and 0.0018 m/s and 0.0041 m/s, respectively, for the velocity field.

Table 2: Boundary conditions of evaluation sites: 10 and 150

evaluation site	10	150
flow velocity	2.32 m/s	14.12 m/s
engine temperature	781.28 K	326.28 K
ambient air temperature	300.05 K	307.30 K

The true and predicted thermal and flow CFD contours at evaluation sites No. 10 and No. 150 are depicted in Figure 5 and Figure 6, respectively. For both figures, (a) and (b) display thermal and flow field solutions. In general, all contour plots exhibit extremely accurate predictions over the entire domain, and the difference between the true and predicted plots cannot be observed visually and without specifically extracting the difference between the two. From the error plots, it can be seen that most of the errors are close to zero for the entire domain. However, relatively larger errors exist near the boundaries of the vehicle. For the thermal predictions, since two different domains are comprised of different materials with disparate temperatures, the prediction becomes difficult at the boundary. Flow predictions also experience larger errors near the vehicle boundaries due to more complicated flow motions near the solid surfaces that are difficult to capture. The largest absolute errors for the temperature contours at evaluation sites No. 10 and No. 150 are less than 7 K and 2 K, respectively. Similarly, the largest



(a) (b)

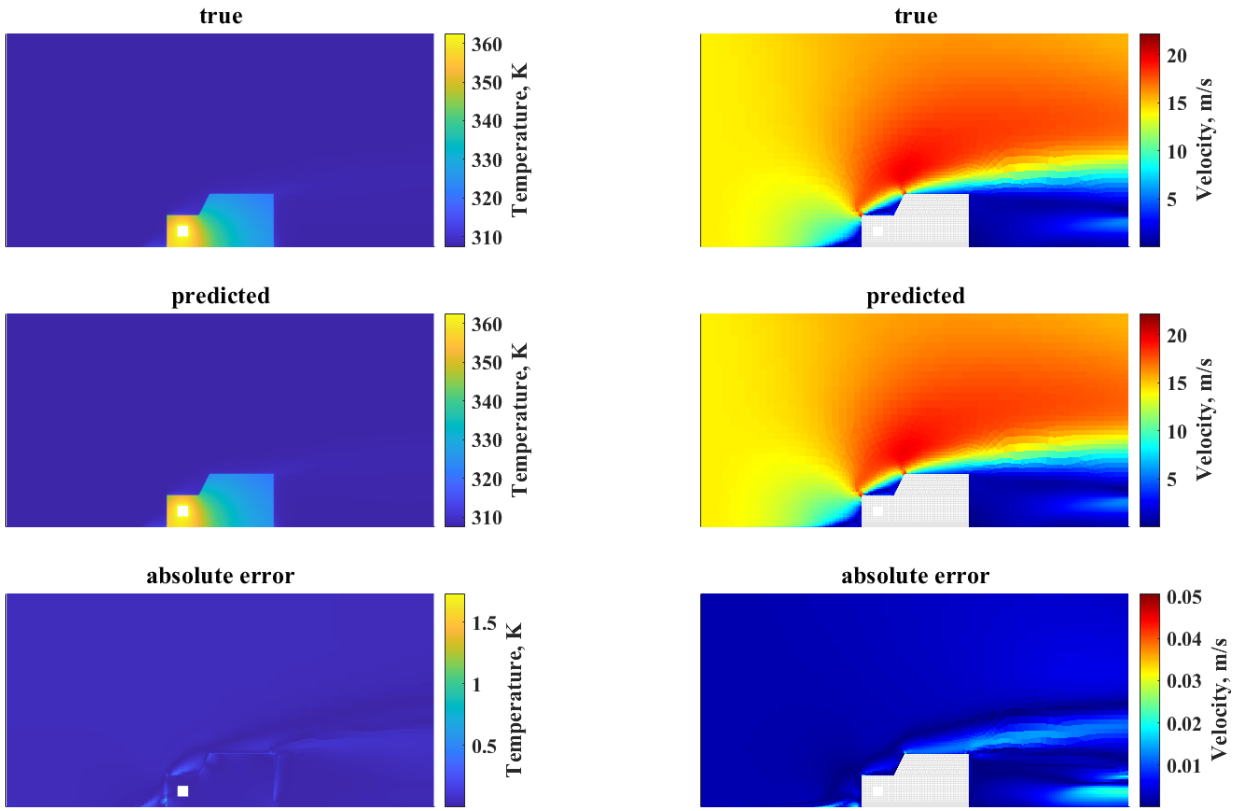
Figure 5: CFD contours at evaluation site 10: (a) temperature and (b) flow velocity

absolute errors for both velocity contours are less than 0.08 m/s. The contour plots further confirm the outstanding performance of the proposed ML model.

5. CONCLUSION

A ML method to predict the thermal and flow simulation is presented in this work. For demonstration, the computational framework is applied to the 2D mock-up vehicle geometry to observe the thermal and flow behaviors around and within the vehicle. The CFD simulation using the OpenFOAM software is utilized to generate the training data. The unstructured mesh is constructed in both fluid and solid regions. Due to a large number of computational cells generated to cover the entire simulation domain, POD method is applied to reduce the data dimension. For the thermal field, the total of 10,865 cells are compressed to 21 modal coefficients. Similarly, the data dimension for the flow field is reduced from 8,988 to 110. The ANN models are trained to predict the modal coefficients instead of the cell values in the full domain. This data reduction allowed ANN models to learn the relation between the CFD simulation parameters and the modal coefficients in the reduced domain even with only 600 train samples. Moreover, each ANN model is trained to predict a single modal coefficient to further increase the accuracy of the prediction.

The ANN predictions on all 150 test samples are completed within 1 second by the heterogeneous computing framework developed in this work, whereas the OpenFOAM software takes approximately 500 seconds for a single CFD simulation by a single CPU core.



(a) (b)

Figure 6: CFD contours at evaluation site 150: (a) temperature and (b) flow velocity

The predictions based on the proposed ML method showed excellent performance for all the query samples. The maximum MAEs for thermal and flow field predictions are computed to be 0.41 K and 0.019 m/s, corresponding to less than 0.1% relative error. In other words, the trained ANN models are able to predict the modal coefficients of the CFD solution data accurately, and also verifies that the POD method can be used effectively to represent the CFD data of the high dimension while preserving the information. For further analysis, temperature and velocity contour plots predicted by the CFD and the ANN models at two distinct sample locations are compared. All predicted contours resemble the true CFD solutions, and their difference is almost visually indistinguishable.

ACKNOWLEDGEMENT

This research was sponsored by the US Army DEVCOM Ground Vehicle Systems Center (GVSC) under the contract number W56HZV-20-0086.

REFERENCES

[1] Z. Luo and G. Chen, “Proper orthogonal decomposition methods for partial differential equations”. Elsevier Science, San Diego, 2018.

[2] S. Chaturantabut, and D. C. Sorensen, “Application of POD and DEIM on dimension reduction of non-linear miscible viscous fingering in porous media”, Mathematical and Computer Modelling of Dynamical Systems, vol 17, issue 4, pages 337-353, 2011.

[3] E. Shlizerman, E. Ding, M. O. Williams, and J. N. Kutz, “The proper orthogonal decomposition for dimensionality reduction in mode-locked lasers and optical systems”, International Journal of Optics, 2012

[4] A. Radermacher, and S. Reese, "POD-based model reduction with empirical interpolation applied to nonlinear elasticity", International Journal for Numerical Methods in Engineering, vol 107, issue 6, pages 477-495, 2016.

[5] V. Mukherjee, M. F. Far, F. Martin, and A. Belahcen, "Constrained algorithm for the selection of uneven snapshots in model order reduction of a bearingless motor", IEEE Transactions on Magnetics, vol 53, issue 6, pages 1-4, 2017.

[6] J. Reiss, P. Schulze, J. Sesterhenn, and V. Mehrmann, "The shifted proper orthogonal decomposition: A mode decomposition for multiple transport phenomena", SIAM Journal on Scientific Computing, vol 40, issue 3, pages A1322-A1344, 2018.

[7] W. Sun, S. Shao, R. Zhao, R. Yan, X. Zhang, and X. Chen, "A sparse auto-encoder-based deep neural network approach for induction motor faults classification", Measurement, vol 89, pages 171-178, 2016.

[8] M. Yousefi-Azar, V. Varadharajan, L. Hamey, and U. Tupakula, "Autoencoder-based feature learning for cyber security applications", In 2017 International joint conference on neural networks, pages 3854-3861, IEEE, May 2017.

[9] K. Chen, J. Hu, and J. He, "Detection and classification of transmission line faults based on unsupervised feature learning and convolutional sparse autoencoder", In 2017 IEEE Power & Energy Society General Meeting, pages 1-1, IEEE, July 2017.

[10] G. Jiang, P. Xie, H. He, and J. Yan, "Wind turbine fault detection using a denoising autoencoder with temporal information", IEEE/Asme transactions on mechatronics, vol 23, issue 1, pages 89-100, 2017.

[11] H. Liu, J. Zhou, Y. Zheng, W. Jiang, and Y. Zhang, "Fault diagnosis of rolling bearings with recurrent neural network-based autoencoders", ISA transactions, vol 77, pages 167-178, 2018.

[12] C. Fan, F. Xiao, Y. Zhao, and J. Wang, "Analytical investigation of autoencoder-based methods for unsupervised anomaly detection in building energy data", Applied energy, vol 211, pages 1123-1135, 2018.

[13] M. Wang, H. X. Li, X. Chen, and Y. Chen, "Deep learning-based model reduction for distributed parameter systems", IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol 46, issue 12, pages 1664-1674, 2016.

[14] Z. Chen, and W. Li, "Multisensor feature fusion for bearing fault diagnosis using sparse autoencoder and deep belief network", IEEE Transactions on Instrumentation and Measurement, vol 66, issue 7, pages 1693-1702, 2017.

[15] Z. Wang, D. Xiao, F. Fang, R. Govindan, C. C. Pain, and Y. Guo, "Model identification of reduced order fluid dynamics systems using deep learning", International Journal for Numerical Methods in Fluids, vol 86, issue 4, pages 255-268, 2018.

[16] E. D. Robertson, Y. Wang, K. Pant, M. J. Grismer, J. A. Cameros, "A Flow Feature Detection Framework for Large-scale Computational Data Based on Incremental Proper Orthogonal Decomposition and Data Mining", International Journal of Computational Fluid Dynamics, vol 32, issue 6-7, pages 261-277, 2018

[17] S. H. Hong, J. Cornelius, Y. Wang, and K. Pant, "Fault compensation by online updating of genetic algorithm-selected neural network model for model predictive control", SN Applied Sciences, vol 1, issue 11, pages 1-16, 2019.

[18] S. H. Hong, H. Yang, and Y. Wang, "Inverse design of microfluidic concentration gradient generator using deep learning and physics-based component model", Microfluidics and Nanofluidics, vol 24, pages 1-20, 2020.

[19] S. H. Hong, J. Cornelius, Y. Wang, and K. Pant, "Optimized Artificial Neural Network Model and Compensator in Model Predictive Control for Anomaly Mitigation", Journal of Dynamic Systems, Measurement, and Control, vol 143, issue 5, 051005, 2021.

[20] S. H. Hong, J. I. Shu, Y. Wang, and O. Baysal, "Automated optimization of double heater convective polymerase chain reaction devices based on CFD simulation database and artificial neural network model", Biomedical Microdevices, vol 23, issue 2, pages 1-14, 2021.