2024 NDIA MICHIGAN CHAPTER
GROUND VEHICLE SYSTEMS ENGINEERING
AND TECHNOLOGY SYMPOSIUM
MODELING, SIMULATION, PROTOTYPING & VALIDATION (MSPV) TECHNICAL SESSION
AUGUST 13-15, 2024 - NOVI, MICHIGAN

DEVELOPMENT OF MACHINE LEARNING MODELS FOR PREDICTING WIND FIELDS AROUND A MILITARY GROUND VEHICLE

Roy Koomullil[1], Emmanuel Ramogi[1], Feroz Mohamed Iqbal[1], Peter Rynes[2], Vladimir Vantsevich[3], Vamshi Korivi[4], and Nathan Tison[4]

[1]Mechanical and Materials Engineering, University of Alabama at Birmingham, Birmingham, AL
[2]ThermoAnalytics, Inc., Calumet, MI
[3]Department of Mechanical and Materials Engineering, Worcester Polytechnic Institute, Worcester, MA
[4]US Army Combat Capabilities Development Command -- Ground Vehicle Systems Center, Warren, MI

ABSTRACT

During multi-day missions, military vehicles face different environmental conditions. Calculating high-fidelity flow fields for these varying conditions in real-time is an impossible task due to the significant computational time required. This paper discusses a machine learning (ML) based approach to predict the flow fields faster than real-time. The testcase for this ML model is taken as the FED-Alpha vehicle geometry, and the training data for the ML model is taken to be the high-fidelity simulation data from computational fluid dynamics studies involving various wind directions using Ansys/Fluent. The surface temperature of the vehicle is calculated based on the operating conditions of the vehicle using the software TAITherm from ThermoAnalytics, Inc. Three different ML models were tested to estimate the accuracy of the predictions and time requirements.

Citation: R. Koomullil, E. Ramogi, F. Iqbal, P. Rynes, V. Vantsevich, V. Korivi, and N. Tison, "Development of Machine Learning Models for Predicting Wind Fields Around a Military Ground Vehicle," In *Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium* (GVSETS), NDIA, Novi, MI, Aug. 13-15, 2024.

1 INTRODUCTION

The efficiency of modern computers has increased in the past decade, and so has the amount of data that can be generated and stored. Moreover, the use of data-driven machine learning (ML) in research is emerging as a fast, effective, and popular tool in the identification of complex and non-linear systems [1]. These advances have been driven by several factors such as extensive increase in data, investment by industry leading to numerous open-source software, scalability of algorithms from statistics and applied mathematics, advancements in data storage and transfer, and improvement in high performance computing [2].

The advantage of data-driven machine learning is confirmed in research done in various disciplines such as aerospace and aircraft design analysis [3], [4], fluid flow prediction [5], etc. With the use of ML, data can be used to build neural network models for predicting flow fields. Building these models involves four major stages: identifying the objective; collecting and curating training data; determining the model structure and parametrization; and choosing an optimization strategy to identify parameters of the model from the data [2]. Different types of neural networks are currently in use. These include Convoluted Neural Network (CNN), Physics Informed Neural Networks (PINN), Deep Neural Networks, etc.

Regardless of the advancements in the use of machine learning, a few difficulties persist. A particular one involves the situation in which ML models built by large data sets require long processing times and large amounts of computational processing power. This can cause delays or downtime, especially when results are needed in real time. It is thus essential to reduce the training data to a smaller subset to save time and computational power.

Hong et al. used popular data dimension reduction methods such as proper orthogonal decomposition (POD) and deep autoencoder on fluid flow analysis data [1]. The authors performed computational fluid dynamics (CFD) simulation on a 2D mock-up vehicle geometry to generate temperature and flow field data. The CFD simulation provided a large data set that was analyzed for flow prediction. This high dimension data set was then reduced by use of the POD method. Furthermore, the reduced data set was used to train an artificial neural network (ANN) model to predict the mode coefficients of the CFD data solution. The predictions based on the model showed excellent performance, with relative errors of less than 0.1%.

Compared to a simple 2D flow field, 3D flow fields are unconstrained, and the complexity of its data can lead to excessive processing times. Moreover, geometric models with intricate patterns add to the complexity of generated data. The computational effort required for such an analysis is substantial, given the complexities of the geometry involved. Consequently, the undertaking demands extensive computing power and time to accurately simulate air flow around complex military vehicles. It is therefore crucial to reduce the large data set to a smaller subset, essentially a sliced representation of the whole data. This can be done using ROM methods such as dynamic mode decomposition, SPOD, and non-uniform dynamic mode decomposition.

Based on the existing methods of ROM, this paper presents a data-driven method to accelerate the evaluation of a flow and temperature field around FED-Alpha ground vehicles. Vehicles such as these are used by the army during missions and can be subjected to different environmental conditions. It is therefore crucial to understand and predict the flow and temperature field patterns around the ground vehicle. This will assist in mission efficiency since real-time analysis can be performed by use of pre-built ML models.

The flow field data generated from the CFD solution is reduced using singular value decomposition (SVD). Using the reduced data domain, the ML model is trained to predict the mode coefficients as outputs while having wind direction as the input. These predicted mode coefficients can be used in conjunction with the mode shapes to predict and reconstruct the flow field. In this paper, three ML model approaches are compared. Predicted mode coefficients from all methods are used to predict and reconstruct the flow fields. The generated flow fields are contrasted and the errors among them are compared. Their differences in accuracy are discussed in section 3.4.

It may be noted that the CFD analysis model used was validated by the often-used testcase in the field of automotive aerodynamics – the Ahmed body, introduced by Ahmed et al. [6] This was used as the testcase because of its geometric simplicity.

This paper has been organized into 3 parts: the methodology, results and discussion, and the conclusion. The methodology part is divided into 3 sections. Section 2.1 discusses the geometry preparation of the FED-Alpha vehicle for computational analysis. Section 2.2 outlines how data was generated, and the dimensionality reduction approach used on the data. Finally, section 2.3 explains the architecture of the machine learning model used for flow field prediction.

The results and discussion section is divided into four groups. Section 3.1 outlines how validation of the CFD model was accomplished. Section 3.2 discusses the thermal analysis of the FED-Alpha vehicle and the analysis software used. Section 3.3 considers the wind field around the vehicle. Lastly, section 3.4 compares the 3 ML models in terms of accuracy and computational resources required.

2 METHODOLOGY

2.1 Geometry Preparation

Calculation of flow field around complex military vehicles for varying wind direction is computationally expensive, which prevents prediction of the wind field in real-time. Therefore, a ML model approach is used for the predition of wind fields faster than real-time. The ML models are trained using high-fidelity simulation results for various wind directions. The test vehicle used for this analysis is FED-Alpha, a test vehicle for the Next Generation NATO Reference Mobility Model [7]. The computer-aided geometry (CAD) definition of the original FED-Alpha geometry contains details of the interior drive trains, gears, suspension, etc., as shown in Figure 1. High-level details are not needed for the CFD simulations. Therefore, SpaceClaim, a CAD preparation software in Ansys [8] is used to simplify the geometry by removing the fine-scale details and to make the geometry watertight. The final geometry used for the computational simulation is shown in Figure 2.

The geometric ratios of the FED-Alpha model to the refined region and the computational domain are as follows: 1) The radius of the computational domain is 16 times the length of the FED Alpha. 2) The height of the computational domain is 8 times the height of the FED Alpha.

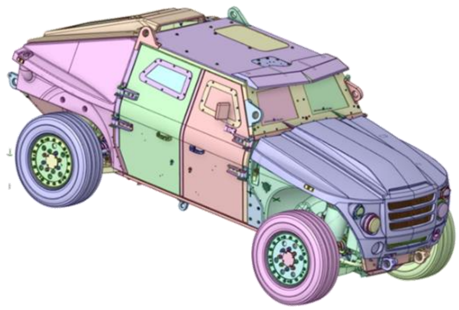


Figure 1: Original detailed geometry of the FED-Alpha vehicle.

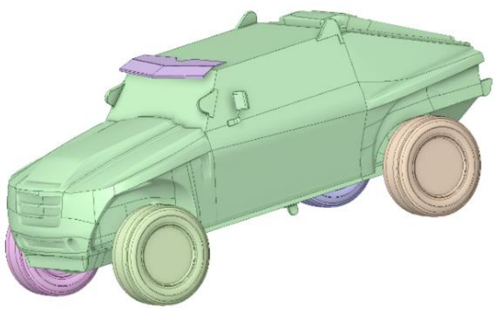


Figure 2: Simplified FED-Alpha geometry used for the simulation.

2.2 Data Generation and Proper Orthogonal Decomposition

Computational simulations were carried out for different wind directions using Ansys/Fluent [9] and the velocity and temperature at the nodal points were extracted to train the ML model. In these calculations, the flow is modeled using incompressible flow assumptions, and the SST k-omega turbulence model is used to estimate the eddy viscosity. The boundary conditions were set as follows: ground was slip boundary, sky was slip wall while freestream temperature was set at 300K. The angular velocity of the wheels was considered and set as 39.8 rad/s which corresponds to a vehicle speed of 45 mph.

The number of nodes in the mesh that is needed to get an accurate result for the wind field around the FED-Alpha vehicle is of the order of millions. Using these nodal flow variables for training the ML model is extremely time-consuming. Therefore, the POD using SVD is used as the dimensionality reduction approach [10]. SVD can be written as

$$A = U\Sigma V^T \tag{1}$$

where A is the snapshot matrix, U is the matrix with each column as the eigenvector of the matrix AA^T, Σ is a diagonal matrix with diagonal elements as the square of the eigenvalues of AA^T or A^TA in the descending order, and V^T is the matrix with each row as the eigenvector of the matrix A^TA. The nodal values of the flow variable of interest for each given wind direction is taken as a column in the snapshot matrix for SVD. In POD, U represents mode shapes and ΣV^T represents mode coefficients.

2.3 ML Model Architecture

The ML model employed in this study is a dense neural network (DNN) characterized by multiple hidden layers, with each neuron serving as an input for all neurons in the subsequent layer. Throughout the training process, the model dynamically adjusts weights and biases to enhance its overall performance, learning from the training dataset to make accurate predictions of the desired output. Weights correspond to the connections between neurons, while biases are parameters within each neuron.

The specific DNN configuration used in this study is comprised of six hidden layers, each with 32 neurons. The activation function chosen for the model is Leaky-ReLU where the mean square error (MSE) serves as the loss function. The training process is facilitated by the Adam optimizer, utilizing a batch size of four. The input layer consists of a single neuron representing the wind angle, while the output layer represents the mode coefficients for different mode shapes from POD. Based on the number of neurons in the output layer, three different ML models were considered in this study: 1) For Approach 1, the number of ML models used for the prediction of the flow field is the same as the total number of mode coefficients present in POD. In this case, the number of neurons in the output layer is one, corresponding to the mode coefficient under consideration. 2) For Approach 2, only one ML model is used for the prediction of the flow field. The number of neurons in the output layer for this ML model is the same as the total number of modes present in POD. This will optimize the weights and biases for the neural network to reduce the overall error for all mode coefficients. 3) For Approach 3, only one ML model is used for the prediction of the flow field. The difference between the ML models for Approach 2 and 3 is the number of neurons in the output layer. In Approach 3, the number of neurons in the output layer is the same as the number of modes that are used for the reconstruction of the flow field using the ML model. This number will change based on the cumulative energy of modes from POD that are used for the prediction of the flow field, as described in Section 3.4.

The trained ML model is used to predict the mode coefficients for any arbitrary wind direction. These predicted mode coefficients are multiplied by the corresponding mode shapes and summed together to predict the flow field for any given wind direction. The predicted wind field is compared with the CFD simulation results from Ansys/Fluent to estimate the error in the prediction.

3 RESULTS AND DISCUSSIONS

3.1 CFD Data Validation

Validation of the CFD model used is an essential step in computational simulation. This is done by comparing the numerical solutions with the available experimental data or analytical results for the same class of problems. One of the commonly used benchmark testcases for automotive application is the experimental data for the Ahmed body [6][11][12]. An Ahmed body geometry with a rear slant angle of 35 degrees is shown in Figure 3, and was used for the current validation study. The region around the Ahmed body is refined to capture the complex flow field around the body, which can be seen in the cross-sectional view of the mesh in Figure 4.The geometric ratios of the Ahmed body to the computational domain are as follows. 1) The radius of the computational domain is 18 times the length of the Ahmed Body. 2) The height of the computational domain is 4 times the height of the Ahmed Body.

The final mesh used for the simulation had approximately ten million elements. The element size for the computational domain was 0.86391m.

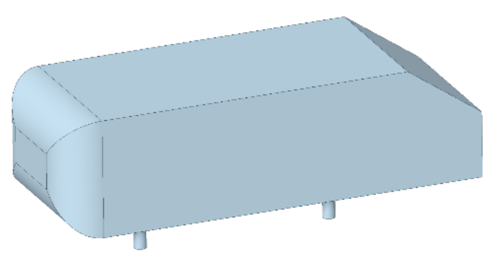


Figure 3: Geometry of Ahmed Body [11]

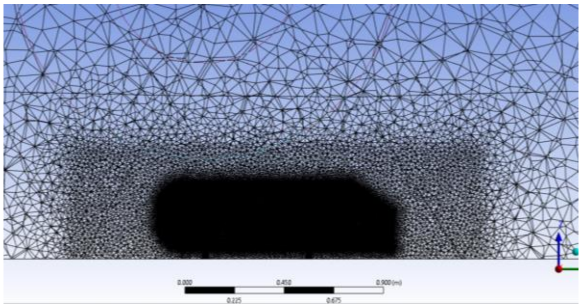


Figure 4: Mesh of Ahmed Body

For this simulation, the freestream velocity is taken to be 20m/s and the freestream temperature is taken to be 300K. Based on the low flow velocity, the incompressible flow assumption is used for the simulation. SST k-omega was used as the turbulence model.

Table 1 compares the calculated value of coefficient of drag (C_d) with the experimental data and other published work. It can be seen from the table that the computational results are in good agreement with the experimental data. This validated model is used for the simulation of air flow around the FED-Alpha geometry in the following sections.

	Drag Coefficient
Experimental Data	0.260
Our simulation	0.258
Results from Ref. [12]	0.3133

Table 1: Comparison of predicted drag coefficient with experimental data

3.2 Surface Temperature Prediction

For the purposes of thermal modeling and analysis presented in the paper, the TAITherm software from ThermoAnalytics is employed [13]. TAITherm is a highly advanced tool designed for thermal analysis, with a key focus on modeling heat distribution patterns across complex components of systems. Its strength lies in its simulation of heat behavior, both transient and steady-state, across detailed surface geometries. The software takes a holistic approach, accounting for three-dimensional heat transfer processes such as conduction, convection, and multi-bounce radiation. TAITherm's main deliverable is a comprehensive thermal map, which provides a detailed representation of temperature fluctuations within the system.

For the thermal analysis of the system, a few additional components are added to the geometry that is used for the CFD simulations. These components include: 1) A simplified model of the engine, 2) an exhaust passage, and 3) air space for cabin, engine, and trunk. Also, the geometry of the vehicle is divided into different groups such as body, windows, mirrors, engine, exhaust, wheel, etc. to specify appropriate material properties such as thermal conductivities, convective heat transfer coefficients, and radiation emissivities and absorptivities. A quad-dominant mesh is used for the discretization of the geometry model.

For the current simulation, based on the assumed vehicle speed of 45 mph, the coolant heat rejection was set as 45kW and the exhaust mass flow rate was taken to be 5 kg/min. The specific timeframe selected for the simulation spanned from 06:00 to 14:00 (6 AM to 2 PM) on July 19, 1984, covering an 8-hour period. To optimize the simulation process, the step size was configured to be 30 minutes, ensuring a detailed analysis over the selected duration. The predicted surface temperature of the vehicle at the end of the simulation is plotted in Figure 5. It can be seen from the figure that the surface gets heated up because of the heat rejection from the engine, with cooler temperatures on the window due to its transmissibility with respect to solar radiation.

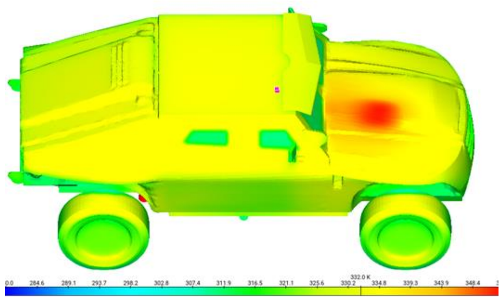


Figure 5: Predicted surface temperature by TAITherm on FED-Alpha geometry.

TAITherm provides functionality to output surface temperature data into a profile file, which is particularly advantageous for subsequent CFD analyses. This export capability ensures a streamlined workflow for CFD coupling with Ansys/Fluent, allowing for an accurate specification of thermal boundary conditions derived from TAITherm. This approach is used to specify the temperature distribution on the surface of FED-Alpha geometry in Ansys/Fluent

3.3 Wind Field Around FED-Alpha Geometry

The wind field around FED-Alpha vehicle could be very complex because of chances of massive flow separation in the wake region. This is particularly true when the wind is coming from the sides of the vehicle. The results from one of these scenarios is plotted in Figure 6, which shows the large, separated flow regions around the vehicle. In this case, the wind is assumed to be coming from 55 degrees to the front of the vehicle.

Figure 6: Visualization of flow over FED-Alpha for a wind angle of 55 degrees; streamline pattern showing separated region.

3.4 ML Model Comparison

In this study, three ML models discussed in Section 2.3 were compared for accuracy and computational resource requirements. All these models have a single neuron in the input layer representing the wind angle. The difference in these ML models is the number of neurons in the output layer representing mode coefficients and the number of ML models used in the reconstruction of the field. Flow simulations were conducted for 91 different wind directions, varying from 0 to 180 degrees in steps of 2 degrees, and taken as the dataset for ML model training and testing. The mesh used for the simulation had 4,463,607 nodes, which resulted in 4,463,607 x 91 as the size of the snapshot matrix for POD for each of the flow variables. The POD of the snapshot matrix using SVD results in 91 mode shapes and the corresponding mode coefficients. SVD predicts the modes in the order of decreasing significance, indicating the first mode contains the most energy of the system. The cumulative energies of the modes are traditionally used to determine how many modes need to be considered for getting good accuracy while reducing the dimensionality. The cumulative energy of the modes for the x-component of the velocity field is plotted in Figure 7. In this graph, the horizontal axis represents the mode number, and the vertical axis represents the cumulative energy in percentage. The cumulative energy for a particular mode is defined as the ratio of the cumulative sum of the eigenvalues up to the mode under consideration and sum of all eigenvalues of the system.

For training the ML models, data from 90 wind directions were used, and for testing the trained ML models one wind direction (of 56 degrees) was used. In this study, NumPy version 1.22.3 and TensorFlow version 2.9.0 were used for the calculations and ML model setup. To estimate the accuracy of the predicted wind field, different numbers of modes corresponding to different levels of cumulative energy were used and the root mean square (RMS) error was calculated. For the velocity components, cumulative energies of 10, 20, 30, 40, 50, 60, 70, 80, 90, and 95 percentage were used. However, for the temperature field, only the 90 and 95 percentages were used, since the first mode itself contained over 90 percent of the energy, indicating not much of a change in temperature for the flow field. This is because the temperature variation in the entire flow field is not as significant.

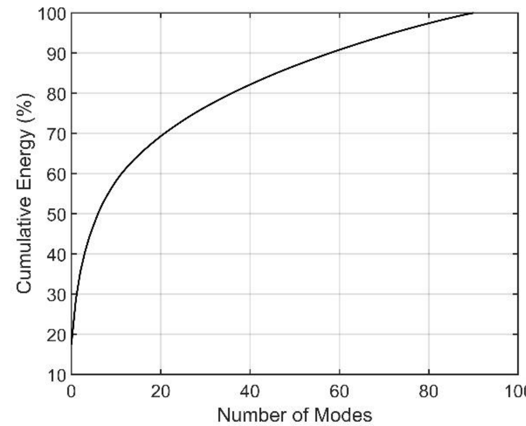
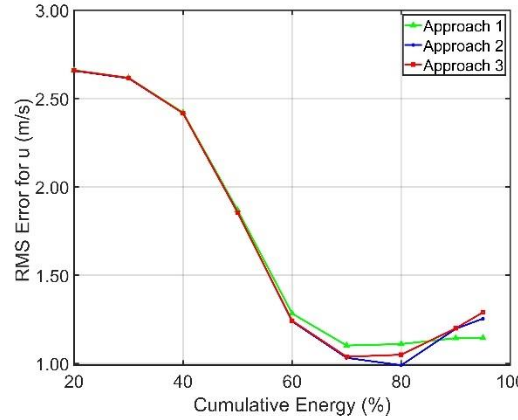


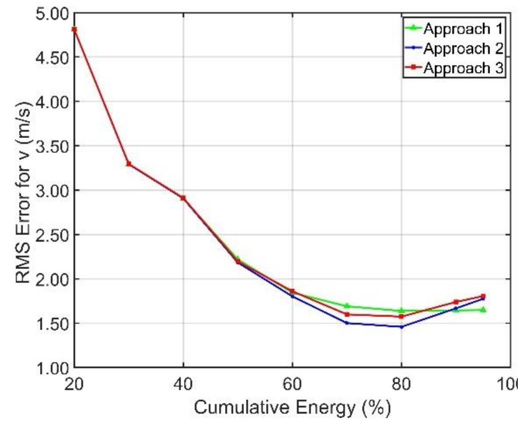
Figure 7: Cumulative energy of different modes from POD of the x-component of the velocity.

The predicted mode coefficients from different machine learning models were multiplied by the mode shapes, and the contributions from different modes were added together to get the predicted wind field. The predicted field results were compared with the CFD results for the test wind angle to calculate the RMS error. For the x-component of the velocity, the number of modes used for 20, 30, 40, 50, 60, 70, 80, 90, and 95 percentages of cumulative energy were 2, 3, 4, 7, 13, 22, 37, 59, and 73 respectively. For example, this indicates that 70% of the cumulative energy is contained in the first 22 modes of POD. Therefore, to estimate the error in flow field prediction using 70% cumulative energy of POD, the following ML models were considered. 1) For Approach 1, 22 different ML models were taken with one neuron each in the output layer corresponding to each of the first 22 mode coefficients. 2) For Approach 2, one ML model was trained using 91 neurons in the output layer corresponding to all modes from POD. From the prediction, the first 22 mode coefficients were used for predicting the flow field. 3) For Approach 3, one ML model was trained using 22 neurons in the output layer and the output from these 22 neurons were used for the prediction of the flow field.

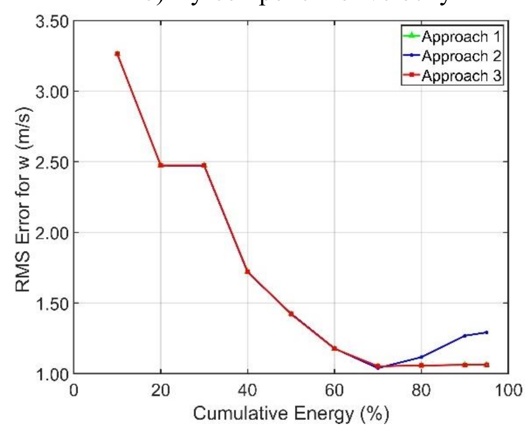
A comparison of the RMS error in the predicted flow field for x-, y-, and z-components of the velocity, and temperature are shown in Figure 8. From the figures it can be observed that: 1) There is near consistency in the trend lines for all approaches. 2) The RMS error for the 3 components of velocity significantly drop from 10-60% of the cumulative energy. This is because the velocity field has large fluctuations, and a larger number of modes are required to capture all these variations. Therefore, using more modes translates to better model accuracy and decreased error. 3) The lowest error is achieved at 80% of the cumulative energy. Therefore, the number of mode coefficients with respect to 80% cumulative energy is the optimum for predicting the flow field. It is observed that there is a slight increase in the error after including modes for cumulative energy greater than 80% for Approaches 2 and 3. There are two reasons for this behavior. Firstly, the number of neurons in the output layer increases as the energy content increases. Therefore, the weights and biases of the network are optimized to minimize the overall error from all neurons in the output layer. Even though the contributions to the predicted flow field from the low energy modes are small, those were also considered for the optimization, which could influence predicted mode coefficients for higher energy modes. Secondly, the mode coefficients for low energy modes are very close to zero. Any slight error in the predicted mode coefficients for the low energy modes by the ML model will increase the error in the predicted flow field. Therefore, it is advisable not to include low energy modes in the ML model.



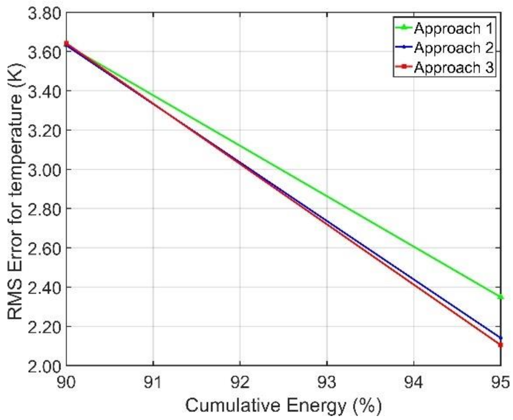
a) x-component of velocity



b) y-component of velocity



c) z-component of velocity



d) Temperature

Figure 8: RMS errors in prediction using three different approaches for velocity components and temperature.

Another way to visualize the error in the predicted flow field is the iso-surface of the error. The iso-surface corresponding to an absolute maximum error of 5 m/s for the x-component of velocity for 70% cumulative energy levels is shown in Figure 9. This iso-surface indicates the region in the flow field where the error greater than 5 m/s is confined. This plot was made from the predicted x-component of the velocity using the ML model in approach 3. It can be seen from the figure that the region where the error is greater than 5 m/s is confined in a small region of separated flow in the computational domain. It was observed from the calculations that the region where the error is greater than 5 m/s progressively decreases as the cumulative energy increases for all flow variables.

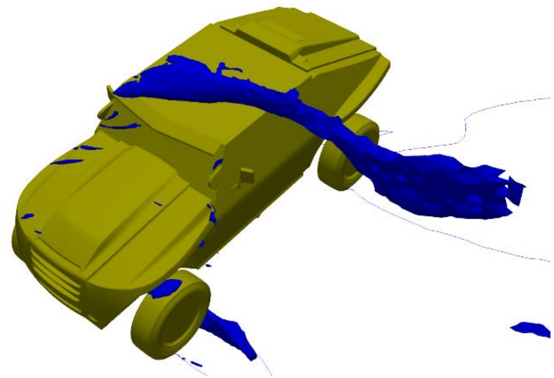


Figure 9: Iso-surface of error corresponding to 5 m/s for x-component of velocity when using 70% of cumulative energies from POD.

The amount of CPU time used for different components of the machine learning model calculations for the x-component of the velocity is summarized in Table 2. These computations were carried out in serial mode using a 2.4 GHz Intel Xeon E5-2680 processor. The amount of memory used was 21 GB. The CPU time requirements for other flow variables were also comparable to the values given in the table. As seen in Table 2, Approach 1 takes 11486 seconds for training due to the use of multiple ML models. However, it takes shorter training time for both Approaches 2 and 3 since they both have only one ML model.

	Approach 1	Approach 2	Approach 3
SVD	1.08	1.10	0.81
Training	11486 (91 ML models)	117.05	128.56
Prediction	10.8605 (91 ML models)	0.15	0.16
Reconstruction	0.18	0.16	0.20

Table 2: CPU timings for x-component of velocity.

4 CONCLUSION

Three different machine learning models were implemented and tested to predict the flow field faster than real-time for the FED-Alpha geometry. High-fidelity wind field data generated using CFD was used to train these ML models. The CFD model was validated using benchmark data for the Ahmed body. POD using SVD was used for dimensionality reduction. Different cumulative energies from POD were used to estimate the error in the predictions, and it was concluded that only the modes involving up to 70 – 80% cumulative energy are needed to get accurate results for the flow field predictions. It was noted that the computational time required for training different ML models for different mode coefficients takes significant computational time, without much gain in the accuracy in the prediction. It was also concluded that a single ML model with neurons in the output layer for all mode coefficients performed the best in terms of accuracy, computational resource requirements, and ease of implementation.

5 ACKNOWLEDGEMENT

This work has been conducted under sponsorship of the U.S. Department of Defense, Contract no. W56HZV1920001, Subaward no. SUBK00019350. The authors would like to acknowledge Dr. Jeffrey D. Naber and Mr. Yashodeep Lonari of Michigan Technological University for their help in proving heat rejection and exhaust mass flow rates for the FED-Alpha vehicle. Also, the authors would like to acknowledge Ms. Lauren Tetzloff of ThermoAnalytics, Inc for her help with preparation of simulation model for TAITherm.

6 REFERENCES

[1] S. H. Hong, A. House, A. L. Kaminsky, N. Tison, Y. Ruan, V. Korivi, Y. Wang, and K. Pant, "Machine Learning-based Thermal and Flow Simulation on Heterogeneous Platform for Signature Prediction," in *GVSETS*., Novi, MI, USA, August 10-12, 2021, pp.1-3.

[2] S. L. Brunton, J. N. Kutz, K. Manohar, A. Y. Aravkin, K. Morgansen, J. Klemisch, N. Goebel, J. Buttrick, J. Poskin, A. W. Blom-Schieber, T. Hogan, and D. McDonald, "Data-Driven Aerospace Engineering: Reframing the Industry with Machine Learning," *AIAA*., vol 59, pp.2820-2822, Aug. 2021, doi: 10.2514/1.J060131

[3] J. Li, M. A. Bouhlel, and J. R. R. A. Martins, "Data-Based Approach for Fast Airfoil Analysis and Optimization," *AIAA*., vol 57, pp. 581-596, Feb. 2019, doi:10.2514/1.J057129

[4] J. Li, M. Zhang, C. M. J. Tay, N. Liu, Y. Cui, S. C. Chew, and B. C. Khoo, "Low-Reynolds-number Airfoil Design Optimization using Deep-learning-based Tailored Airfoil Modes," *Aerospace Science and Technology*, vol. 121, 2022, doi:10.1016/j.ast.2021.107309

[5] L. Sun, H. Gao, S. Pan, and J. Wang, "Surrogate modelling for fluid flows based on physics-constrained deep learning without simulation data," *Comput. Methods Appl. Mech. Engrg,* vol 361, 2020, doi: 10.1016/j.cma.2019.112732

[6] S. R. Ahmed, G. Ramm, and G. Faltin, "Some salient features of the time-averaged ground vehicle wake," *SAE Transactions*, Vol. 93, Section 2: 840222-840402, pp. 473-503, 1984

[7] "Cooperative Demonstration of Technology (CDT) for Next-Generation NATO Reference Mobility Model (NG-NRMM)", STO Technical Memorandum,

TM-AVT-308, Edited by Michael Letherwood, Paramsothy Jayakumar, Richard Gerth, and Jean Dasch

[8] "Ansys SpaceClaim: 3D Modeling Software", https://www.ansys.com/products/3d-design/ansys-spaceclaim (accessed on Feb. 10, 2024).

[9] "Ansys Fluent: Fluid Simulation Sooftware", https://www.ansys.com/products/fluids/ansys-fluent (accessed on Feb. 10, 2024).

[10] Steven L. Brunton and J. Nathan Kutz, "Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control", 1st Edition, Cambridge University Press, ISBN-13: 978-1108422093

[11] P. Y. Patel, T. Jannoi, W. Zou, V. Vantsevich, and R. Koomullil, "Aerodynamic analysis of the utility truck with the morphing boom equipment," in *FEDSM*, Toronto, ON, Canada, Aug. 3-5, 2022, doi: 10.1115/FEDSM2022-88368

[12] E. Guilmineau, "Computational study of flow around a simplified car body," J Wind Eng Ind Aerodyn vol. 96, pp. 1207-1217, 2008, doi: 10.1016/j.jweia.2007.06.041

[13] "TAITherm: 3D Thermal Analysis Software", https://www.thermoanalytics.com/taitherm (accessed on Feb. 10, 2024).