Chemical Mixing Simulations with Integrated AI Accelerator

Krzysztof Rojek[1][0000−0002−2635−7345], Roman Wyrzykowski[1][0000−0003−1724−1786], and Pawel Gepner[2][0000−0003−0004−1729]

[1] Institute of Computer and Information Sciences, Częstochowa University of Technology, Poland
https://pcz.pl
[2] Faculty of Mechanical and Industrial Engineering, Warsaw University of Technology, Poland
https://www.pw.edu.pl

Abstract. In this work, we develop a method for integrating an AI model with a CFD solver to predict chemical mixing simulations' output. The proposed AI model is based on a deep neural network with a variational autoencoder that is managed by our AI supervisor. We demonstrate that the developed method allows us to accurately accelerate the steady-state simulations of chemical reactions performed with the MixIT solver from Tridiagonal solutions.
In this paper, we investigate the accuracy and performance of AI-accelerated simulations, considering three different scenarios: i) prediction in cases with the same geometry of mesh as used during training the model, ii) with a modified geometry of tube in which the ingredients are mixed, iii) with a modified geometry of impeller used to mix the ingredients.
Our AI model is trained on a dataset containing 1500 samples of simulated scenarios and can accurately predict the process of chemical mixing under various conditions. We demonstrate that the proposed method achieves accuracy exceeding 90% and reduces the execution time up to 9 times.

Keywords: CFD · chemical mixing · artificial intelligence · machine learning · DNN · HPC

1 Introduction

Artificial intelligence (AI) and machine learning (ML) are rapidly growing fields revolutionizing many areas of science and technology [15], including high-performance computing (HPC) simulations. HPC simulations involve using supercomputers and other advanced computing systems to perform complex calculations and simulations in physics, engineering, biology, etc. AI techniques can be efficiently used by enabling computing platforms to learn from datasets and make accurate extrapolations of simulations to reduce the execution time of intensive solver computations significantly. In particular, machine learning algorithms can be used to analyze and interpret simulation data, identify patterns and trends,

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36021-3_50

and predict outcomes. AI can also be used to optimize simulation parameters and directly predict the results. By leveraging the power of AI/ML, HPC simulations can help researchers and scientists gain new insights and make more informed decisions in a wide range of fields. In this paper, we develop a method for integrating a proposed AI model with the MixIT tool based on the OpenFOAM (Open Field Operation and Manipulation) [9] solver to accelerate chemical mixing simulations.

Chemical mixing simulations are computer-based models used to predict chemical mixtures' behavior under various conditions [15]. These simulations can be used in a variety of industries, including pharmaceuticals, petrochemicals, and food and beverages, to help optimize the production of chemical products and reduce the cost and environmental impact of manufacturing. Overall, chemical mixing simulations are a powerful tool that can help improve chemical manufacturing processes' efficiency while reducing the cost and environmental impact of these processes.

OpenFOAM [9] is a widely used open-source software platform for simulating and analyzing fluid flow and heat transfer. It has been used in a variety of industries, including aerospace, automotive, and chemical processing. One of the main features of OpenFOAM is its highly modular and flexible design, which allows users to easily customize and extend the software to meet their specific needs. OpenFOAM includes a range of solvers and libraries for simulating different types of fluid flow, as well as tools for meshing, visualization, and post-processing of simulation results.

In this paper, we extend the method proposed in our previous papers [15, 14] and explore new techniques and models. The previously proposed method has limitations that restrict its applicability. It can not handle modified geometries of the simulated phenomenon with the required accuracy, being able to solve problems within the close family of scenarios used during training. To overcome this limitation, in this work, we develop new methods which enable us to handle a broader range of scenarios. The contributions of our paper are outlined below:

- We develop a method for integrating the AI model with the computational fluid dynamics (CFD) solver that leverages the power of machine learning to improve the accuracy and efficiency of fluid flow predictions.
- We propose an AI model based on the variational autoencoder (VAE) architecture to predict key quantities in CFD simulations, including pressure and velocity, as well as an ML algorithm detecting the steady state and making a decision to stop the simulation.
- The efficiency of our method is demonstrated through a series of experiments, comparing the results of our AI-accelerated simulations to those obtained using traditional CFD methods.
- It is shown that our AI-accelerated simulations can produce accurate results with different tube and impeller geometries in simulated phenomena.
- The performance and accuracy of the proposed solution are investigated for 3D cases with meshes exceeding 1 million cells.

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36021-3_50

2 Related work

A significant amount of research has focused on the use of AI techniques in CFD simulations in recent years. These techniques can potentially improve the efficiency and accuracy of CFD simulations and enable the simulation of more complex and realistic problems [2, 24].

A common approach is to employ machine learning algorithms to model and predict the behavior of fluids. For example, neural networks were used to predict the aeroelastic response of the coupled system [20], or approximate computational fluid dynamics for modeling turbulent flows [4].

Other researchers have explored using AI techniques to optimize the parameters and settings of CFD simulations [11]. Genetic algorithms and other optimization methods were used to identify the optimal mesh size and solver settings for a given simulation and to adjust these parameters based on the simulation results automatically [1].

The use of AI techniques to analyze and interpret the results of CFD simulations was also a subject of research [27]. For example, clustering algorithms were exploited to group similar flow patterns [16], and classification algorithms were used to identify and classify different types of flow.

There are several ways in which AI techniques are incorporated into HPC simulations. Some common approaches include:

- Machine learning-based models: Machine learning algorithms can be used to model and predict the behavior of systems being simulated by HPC systems [14, 2, 26]. These models are trained on large amounts of data and used to make accurate and efficient predictions [7], which can be incorporated into HPC simulation.
- Optimization of simulation parameters: AI techniques, such as deep learning and machine learning, are used to optimize the parameters and settings of HPC simulations [13, 25]. These techniques can search through a large space of possible parameter values and identify the optimal configurations for a given simulation.
- Data analysis and approximation: AI techniques, such as interpolation with deep learning algorithms, can be used to create generative models able to accurately approximate the training data set of HPC simulations [3]. These techniques identify patterns and trends in the data and provide insights that may not be immediately apparent from the raw data.
- Real-time control: In some cases, AI techniques are used to control HPC simulations in real-time [5]. For example, an AI system is used to adjust the parameters of a simulation based on the current state of the system being simulated.

There has been a growing interest in using AI techniques in weather forecasting simulations in recent years [23]. These techniques can potentially improve the accuracy and reliability of weather forecasts and enable the simulation of more complex and realistic weather scenarios. One common approach is to use machine learning algorithms to model and predict the atmosphere's behavior.

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36021-3_50

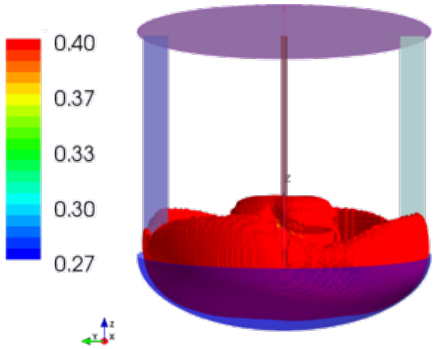


Fig. 1: Tube equipped with a single impeller during the simulation of the magnitude of U [m/s].

For example, neural networks were exploited to model and forecast the evolution of weather systems, such as storms and hurricanes [19]. Other researchers used machine learning techniques to predict the likelihood of extreme weather events, such as floods and droughts. AI techniques have also been used to optimize the parameters and settings of weather forecasting simulations [18]. For example, genetic algorithms and other optimization methods were employed to identify the optimal initial conditions and model configurations for a given simulation [12].

Overall, integrating AI with HPC simulations can greatly improve these simulations' efficiency and accuracy and enable them for more complex and realistic systems.

3 Chemical mixing simulations

One of the main benefits of chemical mixing simulations is that they can be used to test different scenarios and variables without the need for expensive and time-consuming physical experiments. This allows researchers and engineers to quickly evaluate the effects of different factors on the resulting mixture, including temperature, pressure, and mixing speed.

Our simulations are performed with a MixIT tool [6], which is based on the OpenFOAM platform. MixIT is the next-generation collaborative mixing analysis tool designed to facilitate comprehensive stirred tank analysis using laboratory and plant data, empirical correlations, and advanced 3D CFD models.

The chemical mixing simulations are based on the standard k-epsilon model [10]. The goal is to compute the converged state of the liquid mixture in a tank equipped with a single impeller and a set of baffles. Based on different settings of the input parameters, we simulate a set of quantities, including the velocity vector field U, pressure scalar field p, turbulent kinetic energy k of the substance, turbulent dynamic viscosity mut, and turbulent kinetic energy dissipation rate

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36021-3_50

ϵ. This paper focuses on predicting the most important quantities, including U and p. The basic geometry used to train our model is shown in Fig. 1, where we use a cylindrical tube and a single flat impeller. Here we focus on simulations with a mesh of size 1 million cells. The traditional CFD simulation used in our scenario requires 5000 iterations.

4 AI accelerator for CFD simulations

4.1 Basic scheme of AI-accelerated simulations

Fig. 2b presents the basic scheme of the AI-accelerated simulation versus the conventional non-AI simulation illustrated in Fig. 2a. This scheme includes the initial iterations computed by the CFD solver and the AI-accelerated prediction module. The CFD solver produces results sequentially, iteration by iteration. In the basic scheme, which was used in our previous works [14, 15], the results of initial iterations computed by the solver are sent as input to the AI module, which generates the final results of the simulation.

4.2 New method of integrating AI prediction with a CFD solver: AI supervisor

This work proposes a new method for incorporating AI predictions into CFD simulations. Besides the conventional CFD solver, this method involves two other parts. The first one - AI supervisor, is designed to switch between traditional CFD simulation executed by a CFD solver and AI predictions. The second one is the AI accelerator module, responsible for AI predictions. It provides an extrapolation of the simulation to achieve results faster. The overall idea is presented in Fig. 2c.

In this scheme, a traditional CFD simulation is first executed for a specified number of iterations to generate a set of initial data points. These initial data points are then used by a machine learning model, which can accurately predict the fluid flow dynamics in subsequent iterations. Once the machine learning model generates the output, the AI supervisor invokes the traditional CFD solver to resume the simulation on the predicted data. The supervisor continues to switch between CFD and AI parts until a convergence state of the simulation is achieved. The number of iterations required to achieve convergence depends on the complexity of the simulated flow dynamics and the training data quality.

The AI supervisor recognizes the data pattern in simulation and decides if the steady state is achieved. It analyses the output of the CFD simulation and decides whether to invoke the AI accelerator or stop the simulation.

The AI supervisor uses One-Class Support Vector Machine (SVM) [8] model to detect achieving the converged state. This is a type of machine learning algorithm used for anomaly detection. Anomaly detection is the process of identifying observations that deviate significantly from most of the data points, which are considered normal or expected. This algorithm is trained by the data containing

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36021-3_50

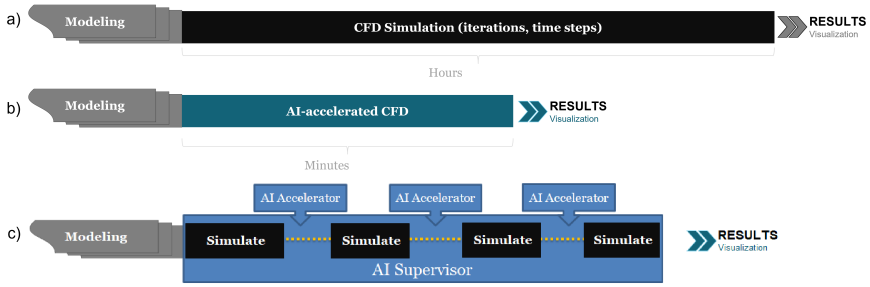


Fig. 2: Comparison of the traditional workflow of CFD simulation (a), the basic scheme of AI-accelerated simulation (b), and the proposed method of AI-accelerated simulation with AI supervisor (c).

the standard deviation of differences between elements of vectors corresponding to boundary iterations of the last 75 iterations of the CFD simulations. Such data are assumed to contain normal (expected) samples. The algorithm then attempts to build a boundary around these normal examples so that new, unseen data points that fall outside the boundary are classified as anomalies.

The supervisor will stop the simulation and return the results if the predicted output is sufficiently close to the converged state. If the predicted output is not close enough to the converged state, the supervisor will call the AI accelerator to make a prediction and then executes 100 more iterations of a CFD solver. The simulation flow is described by the Algorithm 1.

The proposed method allows us to improve the performance of the simulations compared to the traditional CFD solver while ensuring that the simulation results are reliable and consistent. Using the converged state as a stopping criterion, we can reduce the number of iterations of the simulations.

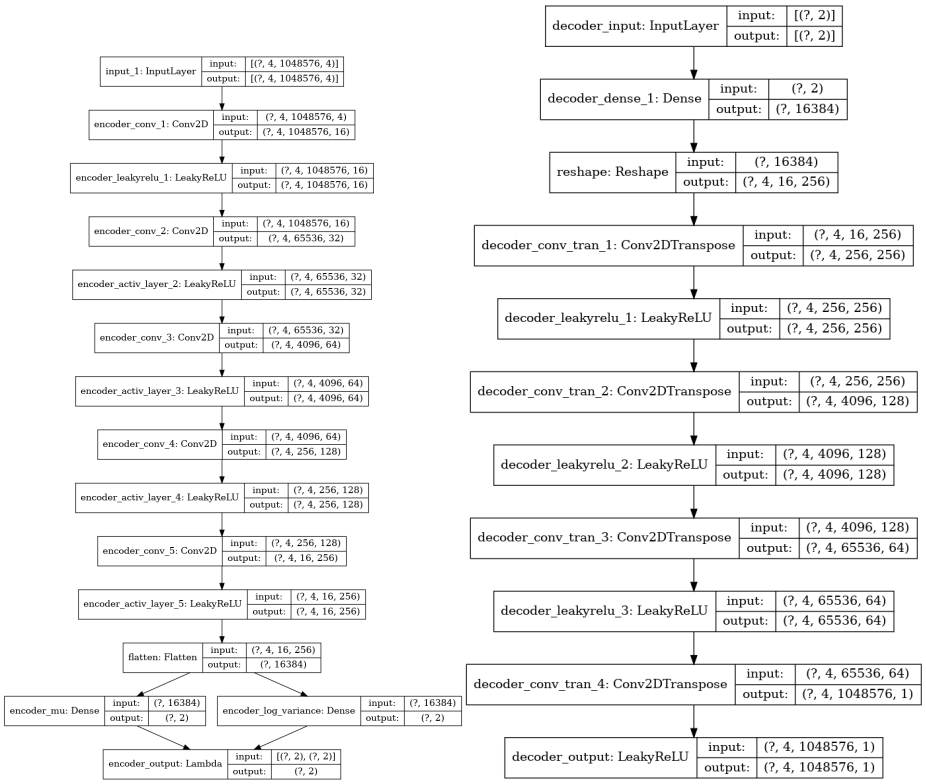
4.3 AI model for CFD acceleration

Our AI model for predicting iterations results is based on a variational autoencoder (VAE) architecture [17]. It is a powerful and flexible neural network architecture well-suited for CFD simulations for two main reasons.

Algorithm 1 Managing a CFD simulation by the AI supervisor

Require: 100 *iterations of CFD solver*
Ensure: $iter_{25}, iter_{50}, iter_{75}, iter_{100}$
  while $OneClassSVM(std(iter_{100} - iter_{25}))$ **do** ▷ True if anomaly detected
    $iter_{pred} \leftarrow model.predict(iter_{25}, iter_{50}, iter_{75}, iter_{100})$ ▷ Predict steady-state
    $iter_{25}, iter_{50}, iter_{75}, iter_{100} \leftarrow CFD_Solver(iter_{pred})$ ▷ Smooth data with solver
  end while
  Return $iter_{100}$

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36021-3_50



(a) Encoder (b) Decoder

Fig. 3: Architecture of encoder and decoder models used to create VAE.

First, the VAE architecture is highly efficient and scalable, allowing us to process large amounts of data quickly and accurately. This is critical in CFD simulations, where large datasets and complex computations are commonly encountered. Second, this architecture allows us to learn the underlying structure and patterns in the data without explicit supervision, making it a powerful tool for understanding complex systems.

In our experiments, we use a machine learning pipeline based on the VAE architecture to predict key quantities in CFD simulations, including pressure and velocity. Specifically, we take four iterations of the CFD simulation as input to the model and return a single output. Inputs and outputs are represented by the quantities corresponding to 1 million cells and are used as a separate row of the input/output arrays. As a result, we have four input rows (three for a 3D velocity vector field and one for a scalar pressure field). This approach allows us to accurately predict the system's behavior over time while accounting for the complex and dynamic nature of fluid flows in CFD simulations. Fig. 3 shows our encoder and decoder, while the full VAE model is presented in Fig. 4.

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36021-3_50

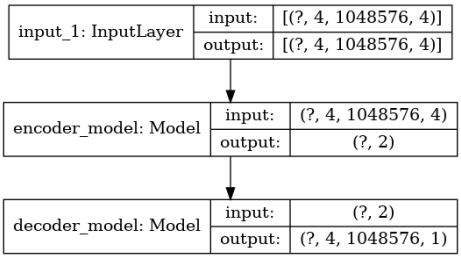


Fig. 4: Full VAE model.

4.4 Dataset

In our simulations, a 3D mesh with 1 million cells is utilized. The mesh was designed to represent the physical characteristics of the system being simulated accurately and to capture the complex fluid flow dynamics in detail.

To feed our network, we use a dataset consisting of results of 50 CFD simulations, each with a different set of parameters and input data, and each requiring 5000 iterations to achieve a steady state. To train our ML model, we create 30 samples for each simulation, resulting in 1500 samples in the dataset. Each sample includes four iterations as input and one iteration as output. Samples generated from a single simulation contain iterations $25 + 100 * i$, $50 + 100 * i$, $75 + 100 * i$, $100 + 100 * i$ (where $i = 0, 1, \ldots, 29$) as input data, and the final 5000-th iteration as output data. All the simulations have the same geometry with a single impeller and cylindrical tube. The differences between the simulations are the results of different liquid levels, rotates per minute (RPM) of the impeller, and viscosity of the mixed substance. The range of RPM varies from 110 to 450, indicating a moderate to the high-speed range. The viscosity range is from 1 to 5000 centipoise (cP) (millipascal x seconds). Finally, the liquid level contains 15 different levels that are linearly distributed across the top half of the tube.

We divided our dataset into two parts for training and validation, with 90% of the data used for training and 10% for validation. This allows us to train our model on a large and diverse set of data while still ensuring that the models can generalize well to new data.

5 Experimental results

5.1 Testing Platform

The testing platform used to evaluate the performance and accuracy of our AI-accelerated simulations consists of the Intel Xeon Gold 6148 CPU equipped with the NVIDIA V100 GPU dedicated to training the machine learning algorithms. The CPU is used for executing a CFD solver and making AI predictions, while

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36021-3_50

the GPU is dedicated to training the model. During model training and inferencing of the model, the half-precision and single-precision formats are used, respectively. Such a mixed-precision approach [22] is beneficial in our case since half-precision reduces memory usage and increases training speed, while single-precision provides more accurate results during inferencing.

The Intel Xeon Gold 6148 CPU is a server-grade processor that provides 20 cores and 40 threads, with a base clock speed of 2.4 GHz and a turbo frequency of up to 3.7 GHz. The NVIDIA V100 GPU, on the other hand, is a graphics processing unit designed specifically for machine learning and other HPC applications. It includes 5120 CUDA cores and 640 Tensor Cores, with a peak performance of up to 7.8 Tflops for double-precision calculations and 15.7 Tflops for single-precision. It also features 16 GB of HBM2 memory with a memory bandwidth of up to 900 GB/s.

In our experiments, a significant amount of memory is required to handle the large dataset and complex computations required for our simulations. Specifically, 400GB of DRAM memory is used to store and process the data used in our simulations.

5.2 Accuracy results

Verifying accuracy includes both visualizations and numerical metrics. Visualizations such as contour plots provide qualitative verification of the flow field. We also use statistical metrics [21] such as the root-mean-square error (RMSE) and Pearson and Spearman coefficients to assess the accuracy of our predictions quantitatively.

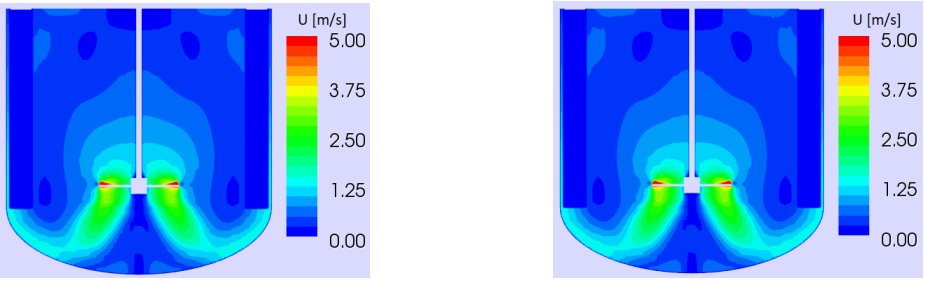
The experiments in this section are based on three newly created cases. First, we consider the case with the same mesh geometry as used while training the model. Second, we study the case with a modified geometry of the tube in which the ingredients are mixed. Finally, a modified geometry of the impeller used to mix the ingredients is considered.

Fig. 5 shows contour plots of the velocity magnitude (U). In relation to cases used during training, this case contains the same geometry of tube and impeller but a different combination of values of RPM, viscosity, and liquid level. For these experiments, we use the parameters from the middle of their respective ranges. This means that the RPM is set to 280, while the viscosity is set to 2500 cP. However, for the liquid level, we use a fully filled tube. By using these parameters, we can expect to see a wide range of fluid properties and accurately measure their behavior in this controlled environment.

Fig. 6 presents the simulation results when a different-shaped tube is used. Such a rectangular geometry of the tube was not used during training. The shape of the tube significantly affects the velocity profile of the fluid.

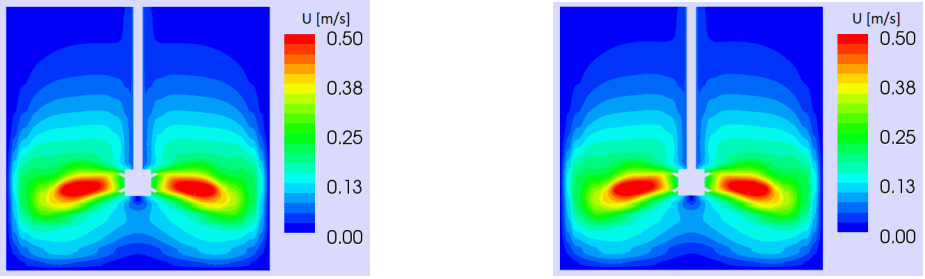
Next, we examine the pressure (p) field in a rectangular tube with flat agitator blades (Fig. 7), which was used during training, and wider agitator blades (Fig. 8), which was not used during training. A wider blade configuration produces a different pressure distribution than a flat blade configuration due to the

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36021-3_50



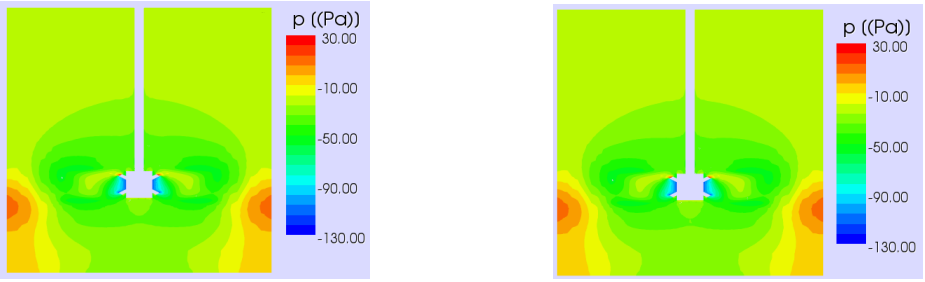
(a) with AI. (b) CFD solver.

Fig. 5: Contour plot of the velocity magnitude (U) in the cylindrical tube.



(a) with AI (b) CFD solver

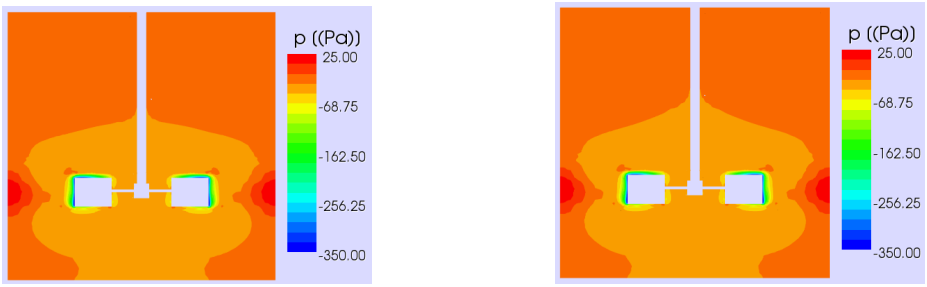
Fig. 6: Contour plot of the velocity magnitude (U) in the rectangular tube.



(a) with AI (b) CFD solver

Fig. 7: Contour plot of the pressure (p) field in the rectangular tube with flat agitator blades.

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36021-3_50



(a) with AI (b) CFD solver

Fig. 8: Contour plot of the pressure (p) field in the rectangular tube with wider agitator blades.

Table 1: Accuracy results for different cases

Case	No modifications		Impeller modif.		Tube modif.	
Quantity	U	p	U	p	U	p
Pearson coef.	0.918	0.984	0.909	0.975	0.902	0.967
Spearman coef.	0.911	0.966	0.902	0.956	0.895	0.949
RMSE	0.012	0.003	0.012	0.003	0.012	0.003
Histogram equal. [%]	92	95	91	94	91	94

increased surface area of the blade that comes into contact with the fluid. Comparing the AI-accelerated CFD results for contour plots we conclude that the quality of predictions is acceptable.

Table 1 contains the accuracy results for all three cases: (i) with the cylindrical tube and flat agitator blades (no modifications in relation to the case used for training), (ii) with modified impeller, and finally, (iii) with the modified tube. The results show a high Pearson correlation, indicating a strong linear relationship between the prediction and accurate results. A high value of the Spearman correlation indicates a strong monotonic relationship between the two variables, meaning that as one variable increases, the other tends to increase or decrease as well. A small RMSE (below 0.02 for all cases) indicates that the predictions made by the model are very close to the accurate values. Finally, the correlation between the two histograms exceeds 90%. It shows that the predicted values are strongly associated with corresponding changes in the accurate values.

5.3 Performance results

The achieved performance results (Table 2) show that AI-accelerated simulations are generally faster than traditional CFD simulations, although the speedup depends on the specific case being simulated. It is difficult to predict precisely the speedup that can be achieved, as it depends on several factors, such as the

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36021-3_50

complexity of the flow dynamics, the amount and quality of available training data, and the performance of the machine learning algorithms. In some cases, the speedup is up to about 10 times, while in others, it is about 2.4 times. The number of iterations executed by the CFD solver is reduced by 90% for the case with no modifications in the geometry of the mesh, by 76% for the case with the impeller modified, and by 57% for the case with the tube modified. The differences in performance between various cases are illustrated in Fig. 9.

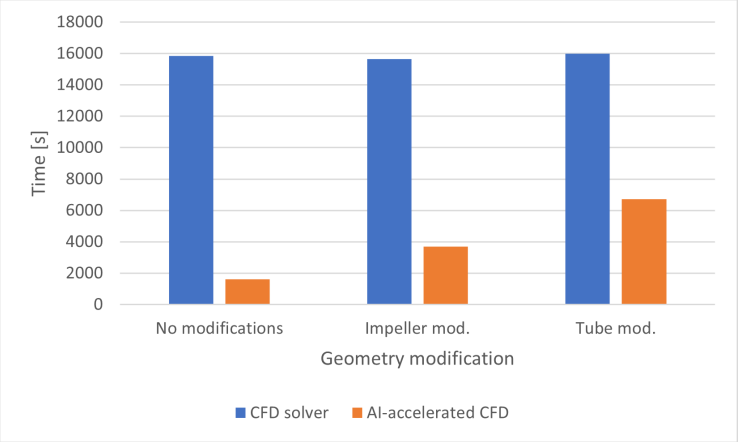


Fig. 9: Execution time comparison between different cases.

The first case with the same geometry as during training is more accelerated than others. Changing the geometry has a significant impact on the speedup of the proposed AI module. This is because the AI model can better learn and make predictions based on patterns that it has encountered before, so simulations that are similar to the ones used during training can be more accurately and faster performed. On the other side, harder-to-predict cases are less accelerated due to the AI supervisor, which executes more iterations to achieve a converged state than in the first case.

Table 2: Performance results for different cases

Case	CFD solver [s]	AI-acc. sim. [s]	Speedup	Sim. reduction [%]
No modifications	15851	1615	9.81	90
Impeller modified	15634	3692	4.23	76
Tube modified	15984	6713	2.38	57

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36021-3_50

6 Conclusion

The proposed machine learning model for predicting chemical mixing simulations demonstrates the potential for AI techniques to improve the performance and keep the accuracy of these simulations. The proposed model shows accuracy exceeding 90% and allows us to achieve a speedup of up to nine times compared to a traditional CFD solver. The proposed supervised learning techniques used for the model's training can accurately predict the evolution of chemical reactions in the simulations and generalize to new scenarios.

Overall, our AI-accelerated algorithm permits us to significantly reduce the time required to simulate fluid flow dynamics while maintaining high accuracy. By combining traditional CFD simulations' strengths with AI-accelerated simulations' speed and efficiency, we optimize the design of industrial processes and improve the efficiency of the whole fluid mixing modeling.

Further research will explore AI's potential in simulating more complex mixing scenarios and fine-tuning the model's performance on different hardware platforms.

Acknowledgements This research was supported by project no. 020/RID/2018 /19 financed within the program of the Polish Ministry of Science and Higher Education "Regional Initiative of Excellence" (years 2019 – 2023, the amount of financing 12 000 000 PLN). This research was partly supported by the PLGrid infrastructure at ACK Cyfronet AGH in Krakow.

References

1. Huang, K., Krügener, M., Brown, A., Menhorn, F., Bungartz, H.J., Hartmann, D.: Machine Learning-Based Optimal Mesh Generation in Computational Fluid Dynamics. ArXiv pp. 1–22 (2021), http://arxiv.org/abs/2102.12923
2. Iserte, S., Macías, A., Martínez-Cuenca, R., Chiva, S., Paredes, R., Quintana-Ortí, E.S.: Accelerating Urban Scale Simulations Leveraging Local Spatial 3D Structure. Journal of Computational Science **62**, 101741 (2022)
3. Kim, B., Azevedo, V.C., Thuerey, N., Kim, T., Gross, M., Solenthaler, B.: Deep fluids: A generative network for parameterized fluid simulations. Computer Graphics Forum **38**(2), 59–70 (2019)
4. Kochkov, D., Smith, J.A., Alieva, A., Wang, Q., Brenner, M.P., Hoyer, S.: Machine Learning-accelerated Computational Fluid Dynamics. Proceedings of the National Academy of Sciences **118**(21), e2101784118 (2021). https://doi.org/10.1073/pnas.2101784118, https://www.pnas.org/doi/abs/10.1073/pnas.2101784118
5. Kurz, M., Offenhäuser, P., Viola, D., Shcherbakov, O., Resch, M., Beck, A.: Deep reinforcement learning for computational fluid dynamics on HPC systems. Journal of Computational Science **65**, 101884 (2022). https://doi.org/https://doi.org/10.1016/j.jocs.2022.101884, https://www.sciencedirect.com/science/article/pii/S1877750322002435
6. MixIT: the enterprise mixing analysis tool. https://mixing-solution.com/. Last accessed 19 April 2023

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36021-3_50

7. Obiols-Sales, O., Vishnu, A., Malaya, N., Chandramowliswharan, A.: CFDNet: A deep learning-based accelerator for fluid simulations. Proceedings of the International Conference on Supercomputing (2020). https://doi.org/10.1145/3392717.3392772
8. One-Class Support Vector Machine (SVM) for Anomaly Detection. https://grabngoinfo.com/one-class-support-vector-machine-svm-for-anomaly detection/. 26 Sept. 2021
9. OpenFOAM. https://www.openfoam.com, [Last accessed 19 April 2023]
10. OpenFOAM: User Guide: k-epsilon. https://www.openfoam.com/ documentation /guides/latest/doc/guide-turbulence-ras-k-epsilon.html, [Last accessed 28 Feb. 2023]
11. Panchigar, D., Kar, K., Shukla, S., Mathew, R.M., Chadha, U., Selvaraj, S.K.: Machine Learning-Based CFD Simulations: A Review, Models, Open Threats, and Future Tactics. Neural Comput. Appl. **34**(24), 21677–21700 (dec 2022). https://doi.org/10.1007/s00521-022-07838-6, https://doi.org/10.1007/s00521-022-07838-6
12. Roh, S., Song, H.J.: Evaluation of Neural Network Emulations for Radiation Parameterization in Cloud Resolving Model. Geophysical Research Letters **47**(21), e2020GL089444 (2020)
13. Rojek, K.: Machine learning method for energy reduction by utilizing dynamic mixed precision on GPU-based supercomputers. Concurrency and Computation: Practice and Experience **31**(6), e4644 (2019). https://doi.org/https://doi.org/10.1002/cpe.4644, https://onlinelibrary.wiley.com/doi/abs/10.1002/cpe.4644, e4644 cpe.4644
14. Rojek, K., Wyrzykowski, R.: Performance and scalability analysis of AI-accelerated CFD simulations across various computing platforms. In: Euro-Par 2022: Parallel Processing Workshops. vol. 13835. Springer Int. Publishing (2023)
15. Rojek, K., Wyrzykowski, R., Gepner, P.: AI-Accelerated CFD Simulation Based on OpenFOAM and CPU/GPU Computing. In: Paszynski, M., Kranzlmüller, D., Krzhizhanovskaya, V.V., Dongarra, J.J., Sloot, P.M.A. (eds.) Computational Science – ICCS 2021. pp. 373–385. Springer Int. Publishing, Cham (2021)
16. Savarese, M., Cuoci, A., De Paepe, W., Parente, A.: Machine learning clustering algorithms for the automatic generation of chemical reactor networks from CFD simulations. Fuel **343**, 127945 (2023). https://doi.org/https://doi.org/10.1016/j.fuel.2023.127945, https://www.sciencedirect.com/science/article/pii/S0016236123005586
17. Schannen, M., Bachem, O., Lucic, M.: Recent advances in autoencoder-based representation learning. arXiv preprint arXiv:1812.05069 (2018)
18. Song, H.J., Kim, P.S.: Effects of Cloud Microphysics on the Universal Performance of Neural Network Radiation Scheme. Geophysical Research Letters **49**(9), e2022GL098601 (2022)
19. Song, H.J., Roh, S., Lee, J., Nam, G., Yun, E., Yoon, J., Kim, P.S.: Benefits of Stochastic Weight Averaging in Developing Neural Network Radiation Scheme for Numerical Weather Prediction. Journal of Advances in Modeling Earth Systems **14**(10), e2021MS002921 (2022)
20. Srivastava, S., Damodaran, M., Khoo, B.C.: Machine Learning Surrogates for Predicting Response of an Aero-structural-sloshing System. arXiv preprint (2019)
21. Statistics How To. https://www.statisticshowto.com/probability-and-statistics/calculus-based-statistics/, [Last accessed 28 Feb. 2023]
22. TensorFlow Core: Mixed precision. https://www.tensorflow.org/guide/mixed_ precision, [Last accessed 19 April 2023]

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36021-3_50

23. Tompson, J., Schlachter, K., Sprechmann, P., Perlin, K.: Accelerating eulerian fluid simulation with convolutional networks. In: Proc. 34th Int. Conf. Machine Learning, ICML'17 - Vol. 70. p. 3424–3433 (2017)
24. Vinuesa, R., Brunton, S.L.: The Potential of Machine Learning to Enhance Computational Fluid Dynamics. arXiv preprint arXiv:2110.02085 (2021)
25. Wyatt, M.R., Yamamoto, V., Tosi, Z., Karlin, I., Essen, B.V.: Is Disaggregation possible for HPC Cognitive Simulation? In: 2021 IEEE/ACM Workshop on Machine Learning in High Performance Computing Environments (MLHPC). pp. 94–105 (2021). https://doi.org/10.1109/MLHPC54614.2021.00014
26. Xiao, D., Heaney, C.E., Mottet, L., Fang, F., Lin, W., Navon, I.M., Guo, Y., Matar, O.K., Robins, A.G., Pain, C.C.: A Reduced Order Model for Turbulent Flows in the Urban Environment Using Machine Learning. Building and Environment **148**, 323–337 (2019)
27. Yu, W., Zhao, F., Yang, W., Xu, H.: Integrated analysis of CFD simulation data with K-means clustering algorithm for soot formation under varied combustion conditions. Applied Thermal Engineering **153**, 299–305 (2019). https://doi.org/https://doi.org/10.1016/j.applthermaleng.2019.03.011, https://www.sciencedirect.com/science/article/pii/S1359431118349172

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36021-3_50