UDC 004.032.26

Valentyn Kuzmych, Mykhailo Novotarskyi.

APPLICATION OF MACHINE LEARNING IN THE MODELING OF PHYSICAL PROCESSES

The article gives an overview of modern approaches to the application of machine learning in the modeling of physical processes on the example of fluid motion in space. The structure and capabilities of artificial neural networks in the modeling of complex physical processes are described.

Key words: modeling, artificial neural networks, lattice Boltzmann model, machine learning, computational fluid dynamics

Fig.: 6. Tabl.: 1. Bibl.: 5

Target setting. Computational Fluid Dynamics is one of the most demandable field of mechanics, a hugely important subject with applications in almost every engineering field, however, fluid simulations are extremely computationally and memory demanding. Due this reason, decreasing of the computation time and memory usage of fluid simulations has become crucial in recent years.

Actual scientific researches and issues analysis. Various approaches have been proposed to solve this issue. Oliver [1] proposed artificial neural network, which consists of three parts and allows to generate considerably larger simulations. Yang etc. [2] use a neural network to solve the Poisson equation in order to accelerate Eulerian fluid simulations. Guo et.al [3] proposed a general and flexible approximation model for real-time prediction of non-uniform steady laminar flow in a 2D or 3D domain based on convolutional neural networks (CNNs).

The research objective. The purpose of this paper is to investigate the application various types of artificial neural networks to fluid flow simulation and compare those applications by following metrics: accuracy, performance, scalability, applicability for other fields of physics.

The statement of basic materials. The common formulation of the task of the simulation of physical processes is the process of mathematical modelling, performed on a computer, which is designed to predict the behavior of or the outcome of a real physical process. If process can be described by differential equations, simulation performs by sequential numerical solving of the set of partial differential equations. The novel approach by using machine learning replace explicit numerical solving on trained model, which converts previous state of simulation into next.

This section describes three different artificial networks which can be applied to fluid flow problems.

1) Convolutional neural network for CFD simulations

This approach was proposed by Guo et.al [3]. It predicts non-uniform steady laminar flow in a 2D or 3D domain on given boundary conditions.

It consists of three key components: adopted signed distance functions (SDF) as a flexible and general geometric representation for convolutional neural networks; multiple convolutional encoding layers for extracting abstract and high-level geometric representations; multiple convolutional decoding layers that map the abstract geometric representations into the computational fluid dynamics velocity field.

Fig. 1 shows general structure of proposed model which was presented in [3]. Main advantage of this approach is that CNN prediction of non-uniform steady laminar flow is considerably faster than traditional LBM [4] solvers. The authors speedup results show that GPU accelerated CNN model achieves up to 12K speedup compared to traditional LBM solvers running on a single CPU core, the CNN model achieves up to 292 speedups compared to GPU-accelerated

LBM solver.

Fig. 2 shows examples of visualization of two-dimensional prediction results that were first considered in [3] First column demonstrates simulation results of traditional LBM solver as a ground truth, second column shows the magnitude of the CNN prediction, third - error magnitude.

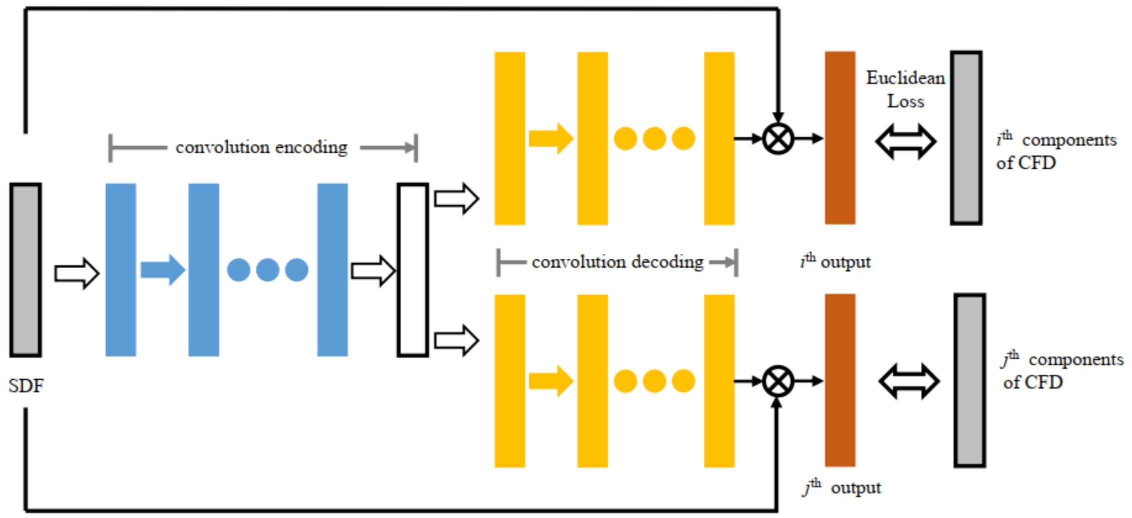

Fig. 1. CNN based CFD surrogate model architecture

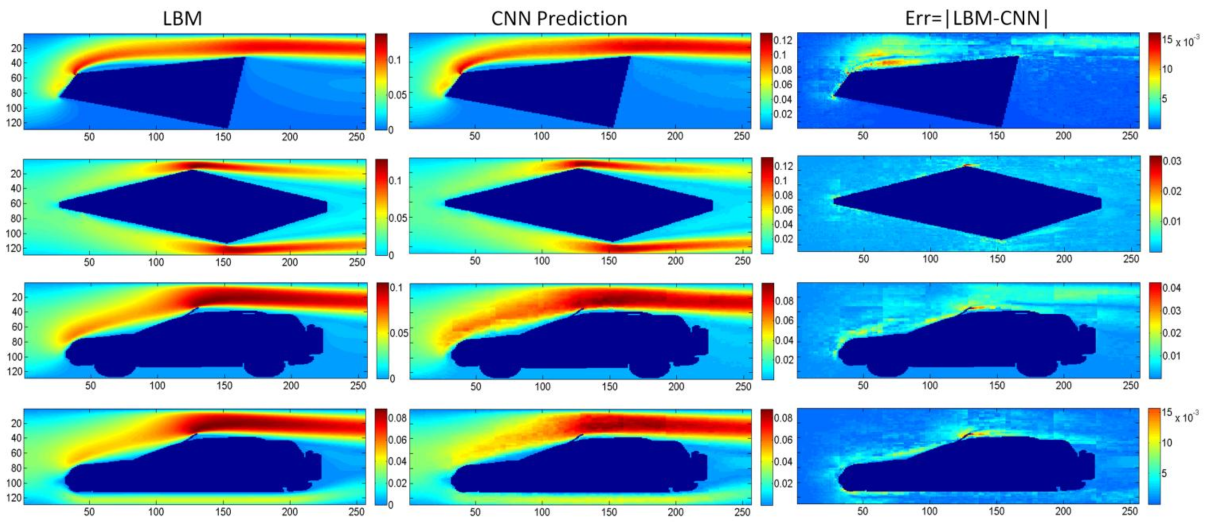

Fig. 2. Prediction visualization

2) Lat-Net

This artificial neural network architecture, Lat-Net, was proposed by Hennigh [1]. It decreases both the computation time and memory usage of Lattice Boltzmann flow simulations. As the author says, that once Lat-Net is trained, model can generalize different grid sizes and various geometries while maintaining accuracy.

Fig. 3 shows the general structure of the Lat-Net architecture and the simulation process first proposed and described in [1]. Simulation process starts by conducting tensor f_t, which represents flow state, with shape $(n_x, n_y, 9)$for the 2D case and $(n_x, n_y, n_z, 15)$ for the 3D case, and tensor b, which represents boundaries, with shape $(n_x, n_y, 1)$or$(n_x, n_y, n_z, 1)$.If the cell is solid, value is 1, and 0 otherwise. Two separate networks, $\phi_{enc}$and ϕ'_{enc}compresses $f_t$and b respectively into tensors g_t, $b_{mul}$and $b_{add}$with equal shapes. After that, model applies the compressed boundary to the compressed state every time-step in following way:

$$g_t = (g_t \odot b_{mul}) + b_{add}$$

Next network ϕ_{comp}emulates the dynamics: $g_t \to g_{t+1}$. In the end, decoder network ϕ_{dec}generates simulation state.

According to the results presented by the author, Lat-Net achieved 3.4x efficiency gain in working memory usage, and about 9x speed increase, in comparison of traditional LBM solver.

Also, that method can be applied to electromagnetic simulations, that shows generality of such approach. For example, fig 4 from [1] demonstrates predicted values for different grid sizes. We can see that the authors of the work were able to obtain results that closely resemble real physical processes

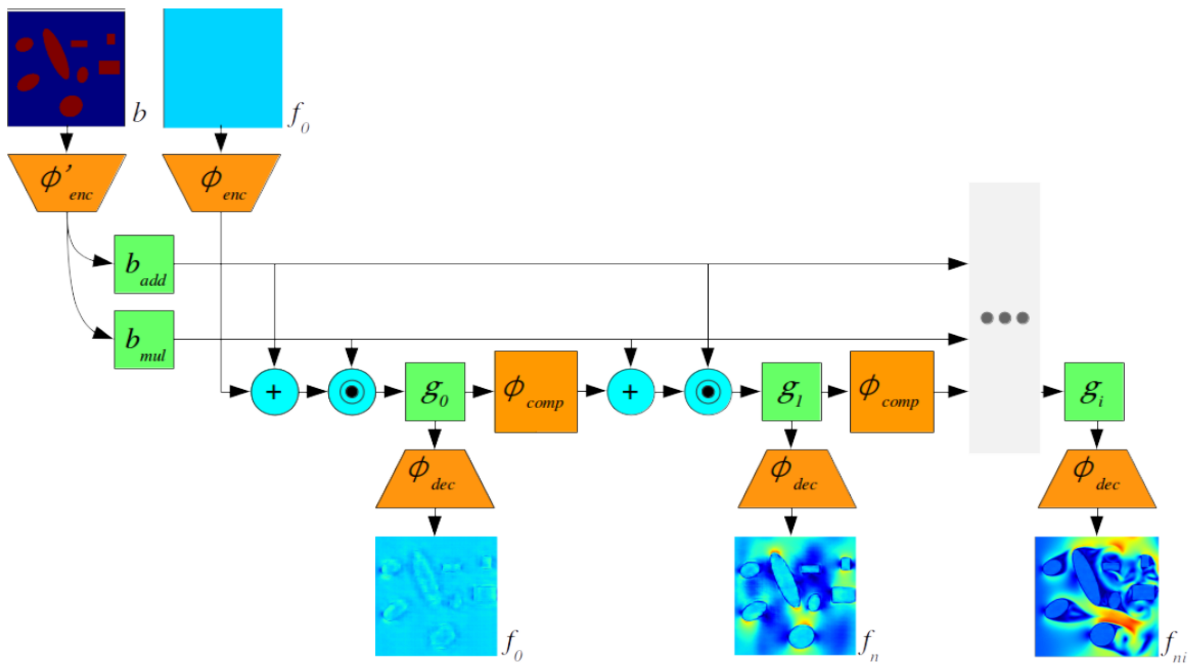

Fig. 3. Structure of the Lat-Net simulation process

3) Data-driven projection method

The projection step is most time-consuming step in numerical calculation to solve Navier – Stokes equations in the grid-based simulations.

Yang et.al [2] proposed a novel data-driven projection method using an artificial neural network for solving the Poisson equation in the projection step. Fig. 5 demonstrates some steps data-driven projection method in grid-based fluid simulation framework. These steps are described in detail in [2].

The author defines Ras the box $[x_{min}, x_{max}] \times [y_{min}, y_{max}] \times [z_{min}, z_{max}]$, each grid at position $x_{i,j,k} \in R$; $Q$is the matrix of the pointwise property $q$per grid: $q\left(x_{i,j,k}\right) \in$ $\in Q(R), x_{i,j,k} \in R$. The function $o(x)$represents the boundary condition, $p_n(x)$ - pressure in grid at frame n, $u_n(x)$- velocity in grid at frame n. PCG [5] function is regarded as projection step in the solving process. Then, pressure values in the grid at the next frame is:

$$P_n(R) = PCG(P_{n-1}(R), \nabla U_{n-1}(R), O_n(R))$$

According to previous equation, a feature vector was defined:

$$\begin{aligned}\beta_n(x_{i,j,k}) = [&p_{n-1}(x_{i,j,k}), p_{n-1}(x_{i-1,j,k}), \nabla u_n(x_{i-1,j,k}), o_{n-1}(x_{i-1,j,k}),\\ &p_{n-1}(x_{i+1,j,k}), \nabla u_n(x_{i+1,j,k}), o_{n-1}(x_{i+1,j,k}),\\ &p_{n-1}(x_{i,j-1,k}), \nabla u_n(x_{i,j-1,k}), o_{n-1}(x_{i,j-1,k}),\\ &p_{n-1}(x_{i,j+1,k}), \nabla u_n(x_{i,j+1,k}), o_{n-1}(x_{i,j+1,k}),\\ &p_{n-1}(x_{i,j,k-1}), \nabla u_n(x_{i,j,k-1}), o_{n-1}(x_{i,j,k-1}),\\ &p_{n-1}(x_{i,j,k+1}), \nabla u_n(x_{i,j,k+1}), o_{n-1}(x_{i,j,k+1})]\end{aligned}$$

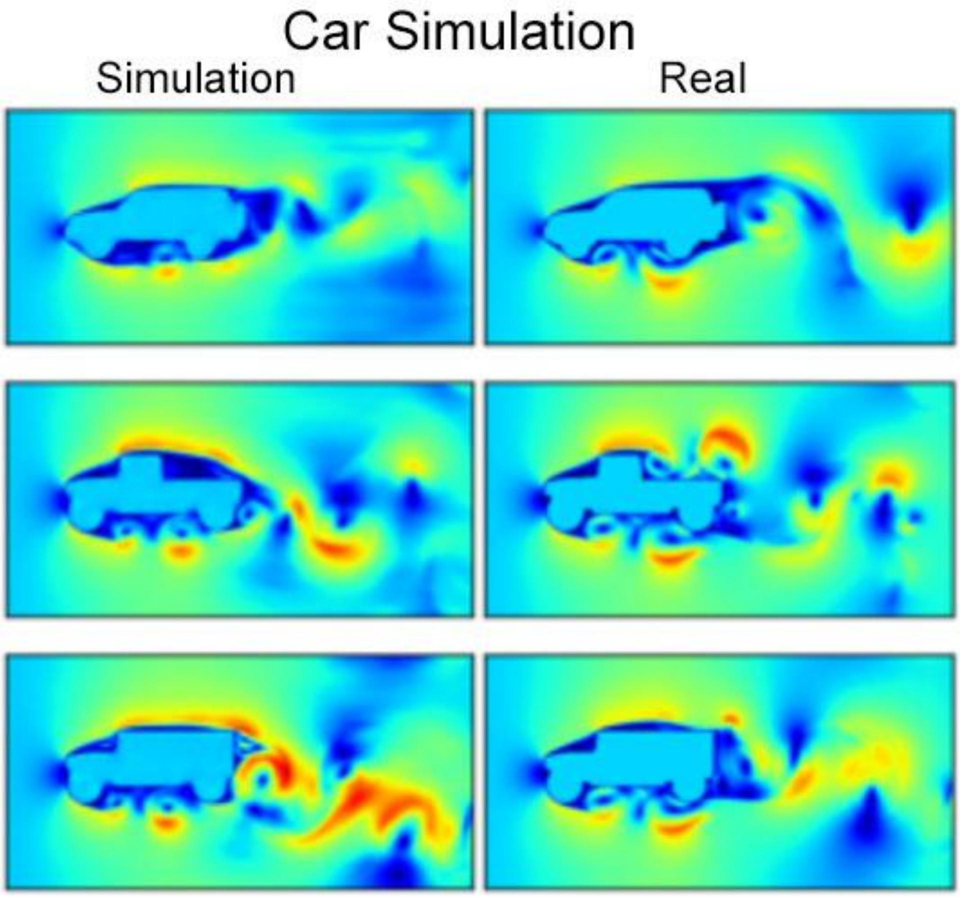

Fig. 4. Visualization of the Lat-Net prediction

Trained artificial neural network receives this vector as input data, and predicts pressure p after projection step. neural network has 20 input neurons (including the bias neuron) in input layer. And the sole neuron in the output layer is the pressure in the next frame.

Fig. 6 shows visualization of simulation results of both data-driven method and PCG in different grid resolutions. These results were summarized in [2].

According to the results presented by the author, method obtained 4.4x speedup in comparison of PCG method at resolution $48 \times 64 \times 48$, and 14.9x speedup at resolution $384 \times 512 \times 384$.

4) Comparison of current approaches from this review

In order to compare mentioned methods, Table 1 shows main features of each method. These features are following: ability to simulate in 2D and 3D space, type of neural network (convolutional or shallow), replaced method (LBM or PCG), value of method's speedup, ability to simulate steady and non-steady flow and ability to be applied in problems of simulation of processes from other fields of physics.

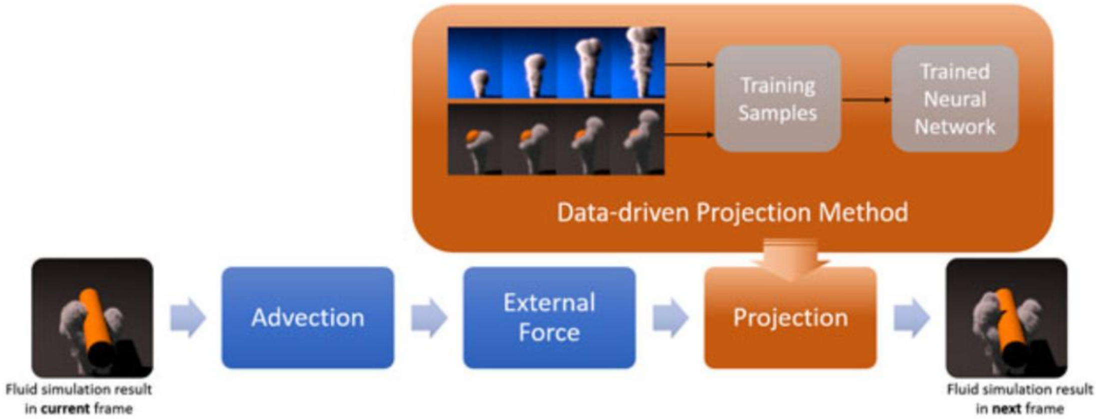

Fig. 5. Framework of data-driven projection method

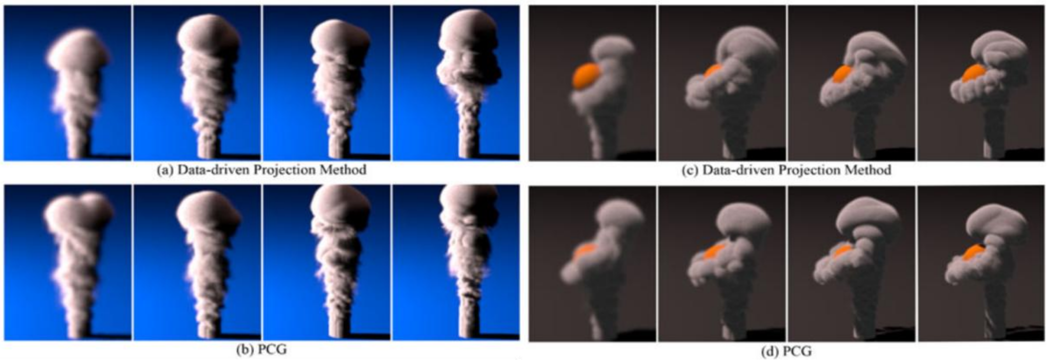

Fig. 6. Visualization of PCG and data-driven projection method

Table 1

Comparison of methods

Feature	CONVOLUTIONAL NEURAL NETWORK SURROGATE MODELS FOR CFD	Lat-Net	Data-driven projection method
2D	+	+	-
3D	+	+	+
Type of NN	Convolutional	Convolutional	Shallow
Replaced method	LBM	LBM	PCG
Speedup	12K(CPU), 292(GPU)	9	4.4-14.9
Ability to simulate steady flow	+	+	-
Ability to simulate non-steady flow	-	+	+
Can be applicable for other fields of physics	-	+	-

Conclusions. The paper has demonstrated and compared novel machine learning methods for fluid flow simulation problem, which achieved crucial decreasing in computation time and memory usage in comparison of traditional methods - LBM and PCG.

Also, one of them, Lat-Net considers by us as promising, in order to it scalability, performance, ability to be applied in both 2D and 3D cases and even possibility to be applied in another field of physics, like electromagnetism. Lat-Net provides wide range of possibilities design exploration and optimization.

References

1. Hennigh O. (2017) Lat-Net: Compressing Lattice Boltzmann Flow Simulations using Deep Neural Networks [Електронний ресурс] // arXiv.org. URL: https://arxiv.org/abs/cond-mat/0106542 (дата звернення: 24.04.2020).
2. Yang C., Yang X., Data-driven projection method in fluid simulation [Електронний ресурс] // Wiley Online Library. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/cav.1695 (дата звернення: 25.04.2020).
3. Guo, Xiaoxiao & Li, Wei & Iorio, Francesco (2016). Convolutional Neural Networks for Steady Flow Approximation [Електронний ресурс] //ResearchGate URL: https://www.researchgate.net/publication/305997840_Convolutional_Neural_Networks_for_Steady_Flow_Approximation (дата звернення: 25.04.2020).
4. G. R. McNamara and G. Zanetti, Use of the boltzmann equation to simulate lattice-gas automata. Physical review letters, vol. 61, no. 20, p. 2332, 1988.
5. Foster N, Fedkiw R. Practical animation of liquids. In Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques. ACM, New York, USA, 2001; 23–30

AUTHORS

Valentyn Kuzmych – PhD student, Department of Computer Engineering, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”.

E-mail: valentine.kuzmich@gmail.com

Mykhailo Novotarskyi (supervisor) – full professor, Department of Computer Engineering, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”.