ParCFD'2020
32nd International Conference on Parallel Computational Fluid Dynamics
May 11-13 2020, Nice FRANCE

GPU-BASED ARTIFICIAL NEURAL NETWORKS FOR CFD

OLIVER HENNIGH[*], SANJAY CHOUDHRY[*] AND STAN POSEY[*]

[*]NVIDIA Corporation
Santa Clara, CA, USA 95050
e-mail: sposey@nvidia.com, web page: https://www.nvidia.com

Key words: GPU, AI, Computational fluid dynamics, Physics-informed neural network.

Summary. Current trends in computational fluid dynamics (CFD) include the use of graphics processing units (GPUs) as parallel co-processors to CPUs in order to accelerate numerical operations and algorithms common to CFD solvers. GPUs are also applied in the novel CFD methods in artificial intelligence (AI) capable of encoding the Navier-Stokes equations into physics-informed neural networks (PINNs) while being agnostic to geometry, or initial and boundary conditions. Examples will include NVIDIA use of such techniques applied in electronics thermal design of heat sinks.

1 INTRODUCTION

Efficient use of computational resources and simulation turn-around times are critically important factors behind engineering decisions to expand CFD technology to support more product design. Recent developments in GPU-based high performance computing (HPC) and AI have improved computational speeds by orders of magnitude for a broad range of CFD simulations relevant to engineering practice. This topic will examine an inverse method to the solution of CFD through implementation of artificial neural networks.

The CFD solution starts with imaging data and discovers the underlying physics and/or properties of a fluid flow configuration directly from the partial differential equations (PDEs) that describe the flow behavior. This inverse approach can used for predicting flow velocities and pressures for an entire transient range, without the need for direct measurement of such quantities and promises advances for a wide range of fluid flow applications.

2 PHYSICS-INFORMED NEURAL NETWORKS

AI research[1] has given rise to applications of physics-informed neural networks (PINNs) that leverage the underlying laws of physics, often described in the form of PDEs, to solve forward, inverse, and model discovery problems. Advantages over traditional methods of solving PDEs relevant to CFD include (i) usability: not requiring arduous meshing, (ii) speed: ability to solve multiple geometries simultaneously, (iii) scalability: embarrassingly parallel across clusters of GPUs, and (iv) expertise: ability to leverage training experience.

NVIDIA has applied PINNs for problems requiring either an inverse approach or a forward solution similar to those available from conventional numerical CFD solvers, for use cases such as the design and optimization of heat sinks for the company's DGX systems powered by the V100 GPU. These networks require no data, can work with single or parameterized geometries and solve single or multiphysics problems. At the backend, GPUs are used for computation of the network training and inference procedures with the cuDNN accelerated TensorFlow deep learning framework.

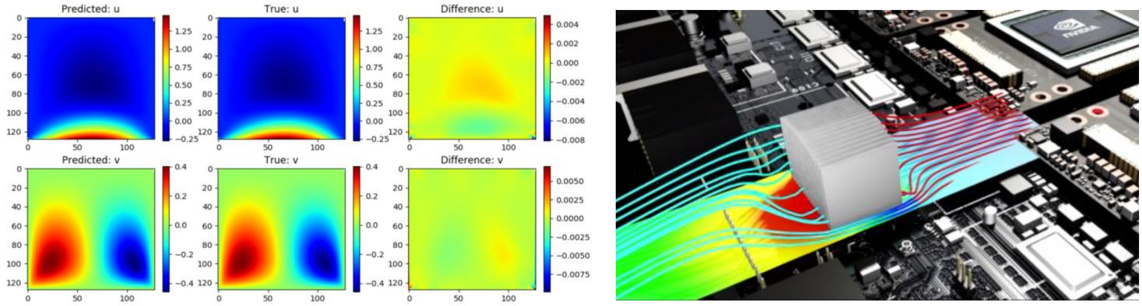

Figure 1: Validation of PINN Predicted results vs. True by an open source CFD solver, for a lid driven cavity, and PINN Predicted thermal results of a typical NVIDIA GPU heat sink design candidate.

The concept behind a neural network solver is to approximate the solution to a given PDE and boundary conditions. This is accomplished by formulating a loss for how well a neural network solution satisfies these conditions. For validation of this idea, a lid driven cavity example was investigated as shown in Figure 1. Results show a forward solution of the PINN approach for the lid driven cavity compares very well with a conventional CFD solver, with errors in the u and v components of velocity being 0.2% and 0.4% respectively.

3 MULTIPHYSICS HEAT SINK

The PINN inverse method described was used to improve the design and effectiveness of heat sink candidates where thousands of configurations could be analyzed within hours as opposed to weeks using conventional CFD simulations. Results are provided in Figure 2. that apply the neural networks solver to heat sink configurations for a multiphysics simulation of both fluid and heat equations with one-way coupling. The left configuration is a simple design with only 3 fins used to benchmark and compare solvers, and the left image shows results for a heat sink design candidate under consideration for next generation DGX servers.

The key physical quantities of interest from the multiphysics simulations are pressure drop and peak temperature. Results for these quantities were calculated using the PINN solver as well as conventional CFD solvers that typically deploy conjugate heat transfer solutions to the multiphysics solution. Table 1. provides results that compare PINN solver, the OpenFOAM solver (www.openfoam.com/) using the pimpleFoam option, and a commercial CFD solver for the 3 fin heat sink, and Table 2. provides comparisons for the design candidate heat sink. For both configurations, the PINN solver shows good agreement and under predicts results in the ~5% range for the 3 pin configuration, and in the ~10% range for the design configuration.

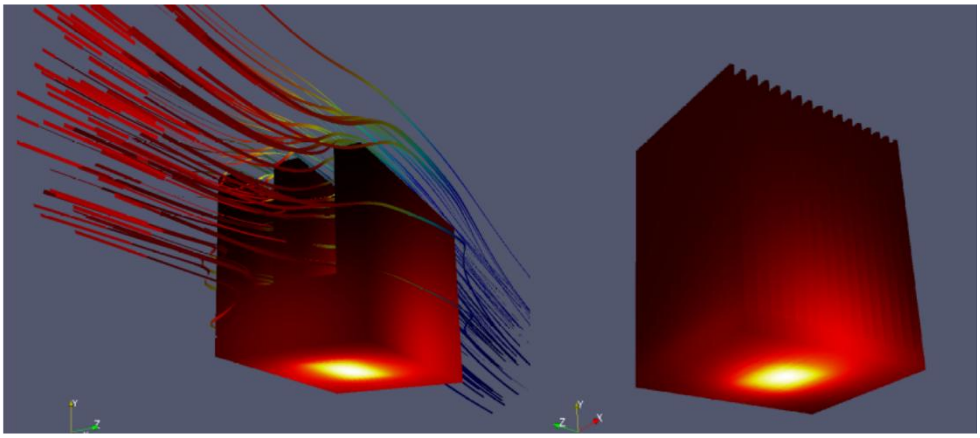

Figure 2: Heat sink configurations, 3 pin (left) and full design candidate (right)

Physical Quantity	Neural Network Solver	OpenFOAM (pimpleFoam)	Commercial CFD Solver
Pressure Drop (Pa)	6.9	7.3	7.3
Peak Temperature (C)	79.8	81.3	80.2

Table 1 : Results of numerical consistency for 3 pin heat sink

Physical Quantity	Neural Network Solver	OpenFOAM (pimpleFoam)	Commercial CFD Solver
Pressure Drop (Pa)	25.6		28.4
Peak Temperature (C)	78.8		84.9

Table 2 : Results of numerical consistency for design heat sink

4 HPC CONSIDERATIONS

The number of potential design evaluations for HPC-driven design optimization is most often restricted by the HPC workload that can fit within overall design cycle times. As an example, investigation of 50 dimensional variations for each of just two design parameters, edge fin height and center fin height, result in 2500 design evaluations. The required HPC resources and computational time using conventional CFD simulations is intractable, however the PINN approach offers the possibility to analyze a full spectrum of design candidates in a fraction of the time.

Comparisons of HPC resource requirements between the PINN approach and conventional CFD solutions are shown in Table 3. The neural network requires 5 days of computational time for the training on 1 x V100 GPU for the entire design space. Once completed, a single inference run is only 3 sec for a single evaluation vs. 1 hour for the commercial CFD solver or

HPC Quantity	Neural Network Solver	Commercial CFD Solver	Factor
Total compute time for 2500 design evaluations	2 hrs*: 3s each on 1 x NVIDIA V100 GPU *Inference time only	104 days: 1h each on 1 x Intel Gold 6128 (SKL) 12 cores @ 3.4 GHz	1200x
Memory capacity (each eval)	0.216 GB	64 GB	296x
Output file size (each eval)	0.5 GB	2 GB	4x

Table 3 : HPC resource requirements of PINN solver and commercial CFD solver

a factor of 1200x that represents 3 orders of magnitude. There are also dramatic reductions in memory capacity requirements by 296x, and a 4x reduction in output file size to help improve file handling and I/O times for post-processing procedures.

5 CONCLUSIONS

As CFD simulation demands increase and motivate the need for more transients, higher-resolutions, and multiscale, multiphysics simulations, GPUs will become an essential HPC technology. The AI approach presented has demonstrated substantial benefits to dramatic reductions in CFD turn-around time and other HPC resource requirements that lead the way towards practical HPC-driven design methods. Based on current trends, GPU-based HPC combined with novel AI techniques will enable a level of applied CFD that can grow as a common practice to support engineering design and optimization procedures.

REFERENCES

[1] M. Raissi, P. Perdikaris, and G. E. Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," Journal of Computational Physics 378 (2019): 686-707.

[2] O. Hennigh, Sukirt, M. Raissi, S. Choudhry, S. Narasimhan, C. Lamb. “Multiphysics Neural Network Solver,” NVIDIA GPU Technology Conference, March, 2020, San Jose, CA, USA.