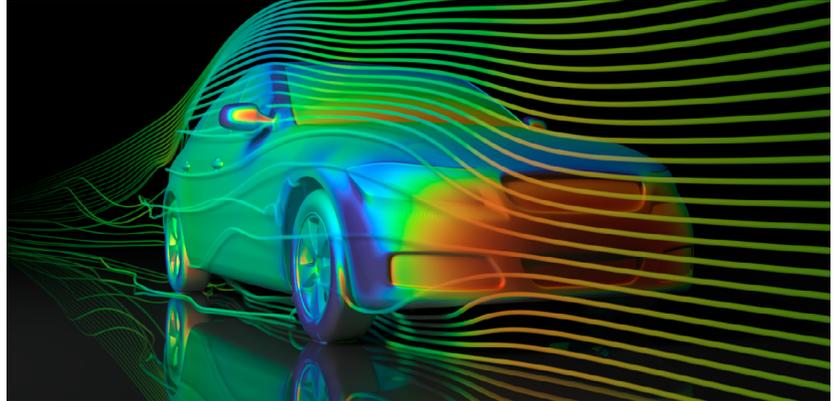


Ansys Fluent Native Multi-GPU Solver: CFD Validation Studies in Version 23R2

/ Introduction

The need for faster solutions for complex scientific and engineering problems is driving the development of numerical methods that make use of modern computer hardware. Nowhere is this more evident than in the realm of computational fluid dynamics (CFD). Methods for simulating fluid flow, heat transfer, combustion, and related physics using vector and parallel CPU computer architectures have been aggressively developed over the past four decades and have greatly benefited from speed and bandwidth advances in CPUs, RAM, network hardware, and storage media.



However, today a new era has launched with the advent of GPU computing. GPUs offer the promise of significant increases in throughput for CFD simulations, and Ansys is at the forefront of this revolution with its native GPU implementation of the Ansys Fluent CFD solver. This development effort has been accompanied by a major testing and validation program to ensure that the GPU solution quality is equal to the flagship Fluent CFD solver quality typically run on CPUs.

This document is a follow-on to the [first white paper](#) on the solution accuracy of the Fluent native GPU solver.^[1] Here, we will report additional results for several canonical and industrial problems typically used to validate and verify CFD codes. These cases span a range of flow and heat transfer physics and help demonstrate the capabilities of the current version of the native GPU solver in 23R2. The emphasis here is on the accuracy of the numerical solutions. Speed comparisons for many industry-relevant cases can be found in [Part 1](#) and [Part 2](#) of our GPU blog series and covers a range of hardware configurations.

/ Unsteady Laminar Flow over a Sphere

In the previous white paper, we presented the canonical case of steady-state, incompressible laminar flow over a sphere at low Reynolds number ($Re=100$). It was found that the native GPU solver accurately captured the salient features of this steady flow configuration with its stable vortex structures in the immediate wake of the sphere.

As a more difficult challenge, we are now investigating the unsteady drag on the sphere in the laminar regime at a higher Reynolds number of $Re=1000$. It is well known that above Reynolds numbers of approximately 200, the vortex structures in the wake behind the sphere begin to separate and form a vortex street. Moreover, the streamlines become aperiodic and irregular as the unsteady flow transitions toward fully turbulent conditions at higher Reynolds numbers.

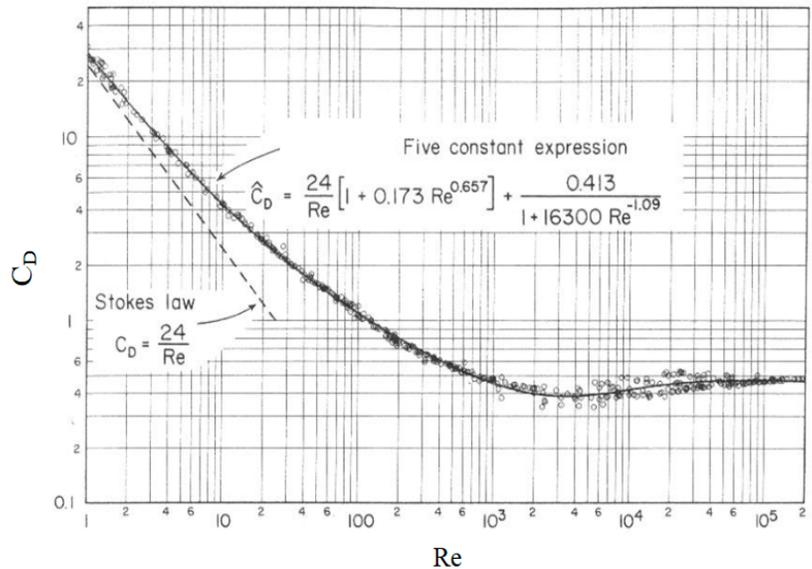


Figure 1. Drag coefficient versus Reynolds number for a sphere in cross flow^[1]

The variation in the drag coefficient on a sphere as a function of Reynolds number is shown in Figure 1. The experimental data in the laminar and transitional regimes are well represented by the “five constant correlation,” as published by Turton, R and Levenspiel.^[2] There is some scatter in the data as the flow becomes increasingly unsteady and aperiodic at $Re > 500$ and beyond.

The model used a refined unstructured polyhedral mesh consisting of 10.4 M cells, as shown in Figure 2. The model simulates the sphere in crossflow inside a rectangular wind tunnel configuration, with velocity prescribed at the inlet and pressure at the outlet. Boundary layer prisms were utilized near the sphere wall surface to capture boundary-layer gradients. The mesh was refined in the wake region to better resolve the vortex structures. The flow was assumed to be laminar and incompressible (density and viscosity were constant). Flow boundary conditions and fluid properties were set to provide the target Reynolds number of 1,000.

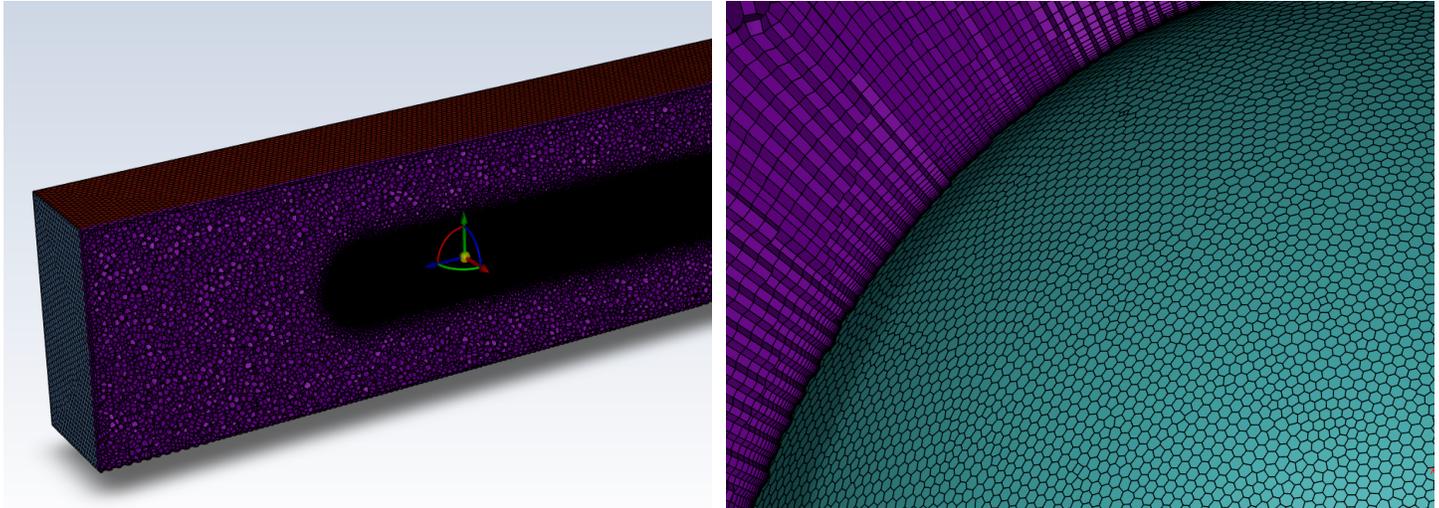


Figure 2. Mesh utilized for the unsteady sphere CFD model.

The solution was carried out using a transient formulation with a fixed time step sufficient to resolve the separation and vortex motions. The history of the drag coefficient is plotted in Figure 3. As can be seen in Figure 3 and the contour plot of velocity in Figure 4, the resulting flow is aperiodic (as expected) and was run until a statistically steady drag coefficient was achieved.

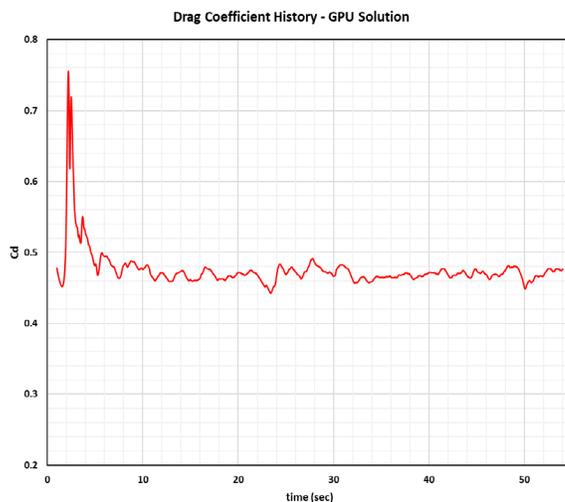


Figure 3. CFD time history of drag coefficient for a sphere in crossflow ($Re=1,000$)

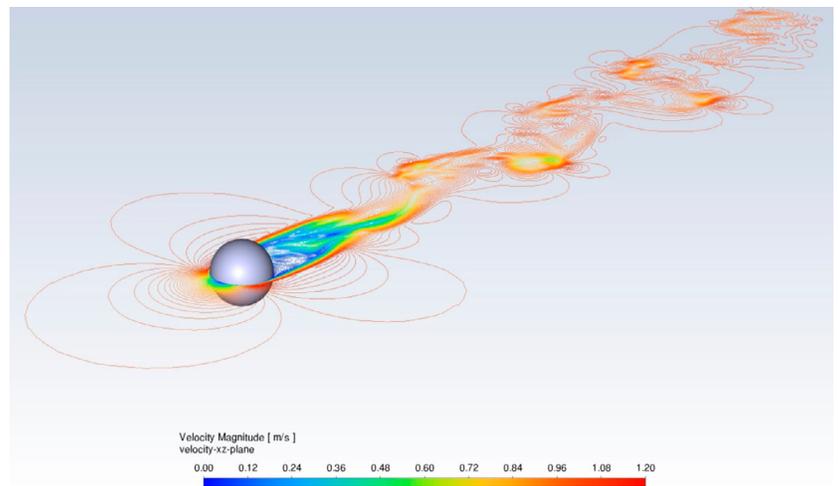


Figure 4. Instantaneous velocity contours illustrating the vortex shedding and aperiodic wake structure downstream of the sphere ($Re= 1,000$)

The mean drag coefficient for the GPU solution was derived from a time average of the final 5,000 time steps.

A comparison of this drag value with the correlation value at $Re=1,000$ is presented in the table below. The CFD result from the native GPU solution is in good agreement with the correlation (within 3%), and thus demonstrates that the native GPU Fluent CFD solver is capable of high-quality unsteady laminar flow solutions. This result is important because applications involving unsteady scale-resolving turbulence rely on accurate numerics in both space and time. The ability to capture unsteady laminar flow is therefore an important benchmark and provides confidence in simulating more complex unsteady flows.

	Drag Coefficient
Correlation ^[2]	0.455
CFD (GPU solver)	0.469

Table 1. Comparison of Drag Value with the Correlation Value at Re=1,000

/ Jet Impingement Heat Transfer

Jet impingement has long been utilized for enhancing heat transfer for important applications, including cooling hot components in turbomachines, electronic integrated circuit cooling, defogging, and more. High rates of heat and mass transfer are obtained at the stagnation point of the flow, and the resulting film of fluid spreads along the wall to provide further cooling. CFD can be applied to predict the heat transfer effectiveness of jet impingement, thereby allowing designers to optimize the geometry and flow patterns for specific configurations.

A canonical form of the jet impingement model is shown in Figure 5 below. It consists of a pipe of diameter (D) oriented perpendicular to a heated wall, with the wall set at a prescribed heat flux. There is a separation distance (L) between the end of the pipe and the wall. Fluid enters the pipe with a prescribed velocity profile and turbulence level and discharges at the perimeter of the heated plate to an open boundary. This configuration has been the focus of numerous studies in the open engineering literature [3], [4], [5], and [6]. The objective of the CFD simulation is to calculate the temperature distribution along the wall for a given inlet flow boundary condition.

A steady-state CFD model was developed using a rectangular 2D axisymmetric mesh, as shown in Figure 6. Mesh elements were focused on the tube and heated wall surfaces so that these high-gradient areas were resolved sufficiently to obtain mesh-independent solutions. Turbulence was modeled using the SST $k-\omega$ turbulence model, and inlet conditions of velocity, temperature, and turbulence intensity were prescribed at the pipe inlet.

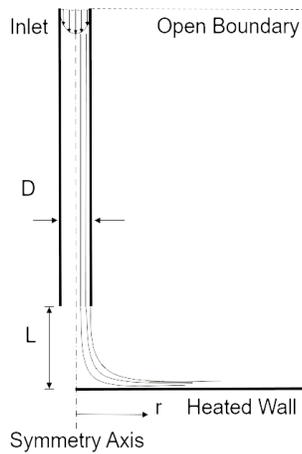


Figure 5. Schematic of the jet impingement heat transfer canonical problem

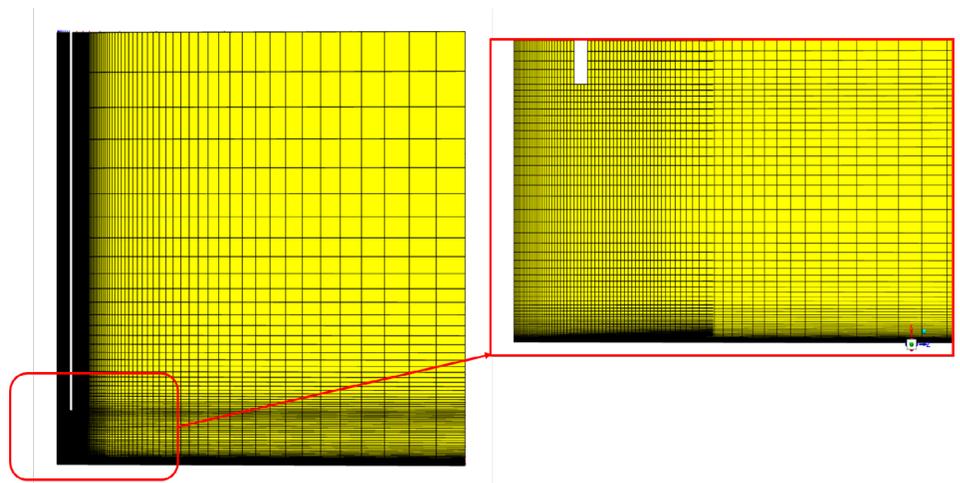


Figure 6. Mesh used for the jet impingement study

A typical velocity field for this case is illustrated in Figure 7, which compares the solutions for both the CPU and GPU versions of Ansys Fluent. As can be seen, the solutions are visually consistent with each other. Physically of note is the local acceleration of fluid along the wall as it moves away from the stagnation point. This is followed by a deceleration as the fluid spreads out radially. As we will see, this has an impact on the plate heat transfer distribution.

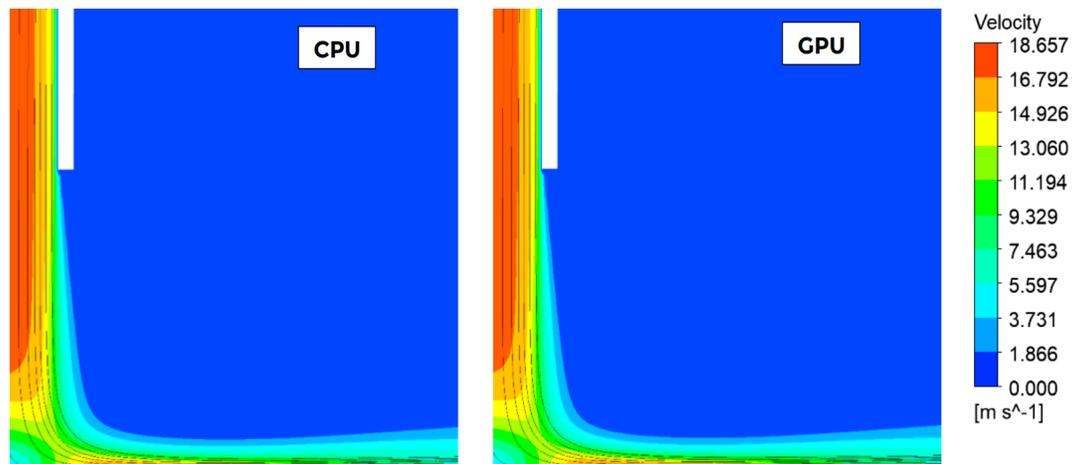


Figure 7. Velocity contours for the jet impingement heat transfer case

Profiles of normalized velocity at selected radial positions are presented in Figure 8. Good agreement is seen between the predicted profiles and the experimental data. The thin boundary layer near the stagnation point is seen to develop as the flow moves radially outward.

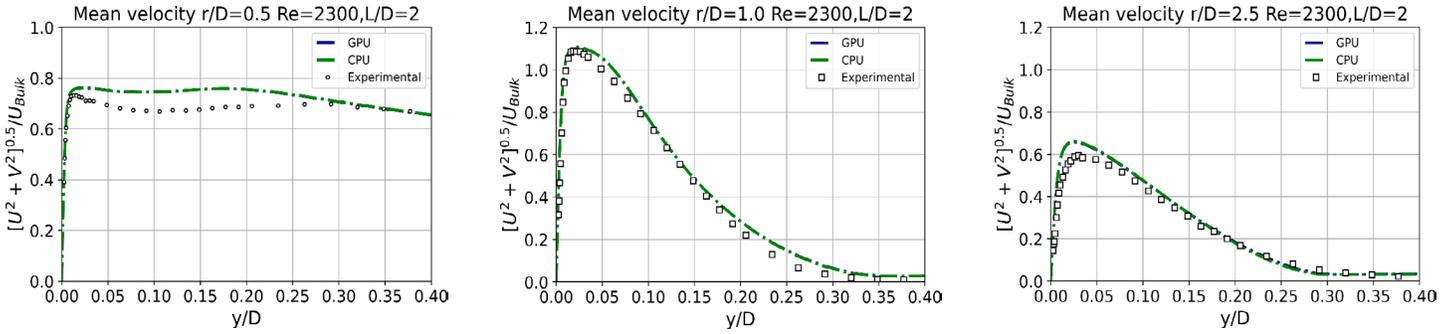


Figure 8. Local velocity profiles with comparison to experiment (Re=2,300): (a) $r/D=0.5$, (b) $r/D=1.0$, (c) $r/D=2.5$

The local Nusselt number distribution at the wall was computed using the predicted wall temperatures from CFD and compared with data from the open literature (Figure 9).

In general, the trend is seen to be consistent with the experiments, with the exception of the dip in Nusselt number at $r/D \sim 1.5$.

This is principally a result of the use of the general-purpose SST $k-\omega$ turbulence model, and improved results could likely be obtained with a turbulence model tuned for stagnation flow. Nevertheless, the GPU solver demonstrates its ability to tackle convective heat transfer cases such as the jet impingement problem, lending confidence in examining more complex cases.

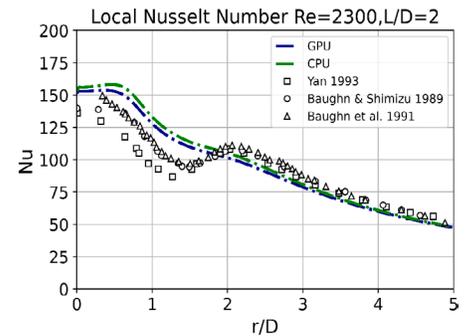


Figure 9. Local Nusselt number distribution for jet impingement heat transfer (Re=2,300)

Marine Propeller

Turbomachinery flows have long benefitted from CFD analysis, so it is no surprise that such cases would comprise an important part of the GPU solver test matrix. The case presented below is a three-bladed marine propeller developed from the well-known Potsdam marine propeller geometry. A full description of the extensive testing database can be found in [7].

The geometry and mesh for the propeller model are shown in Figures 10 and 11. For the experimental tests, the propeller was attached to a shaft and hull submerged in a towing tank. For a fixed propeller speed (n) and diameter (D), the propeller has a forward velocity (V) induced by translational motion in the towing tank. This defines a parameter called the advance ratio (J):

$$J = \frac{V}{n \cdot D}$$

For the CFD model, we can provide a relative forward velocity by imposing a velocity of the fluid at the domain inlet. The rotational speed is imposed on the propeller blades and spinner by employing a moving reference frame to the cell zone surrounding the propeller. This is connected to the remainder of the domain through a cylindrical nonconformal mesh interface, as shown in Figure 11.

The objective of the CFD calculation is to predict the performance of the propeller as a function of advance ratio. The performance is characterized by the rotor thrust force (T), and the rotor torque (Q). These can be further nondimensionalized as the thrust and torque coefficients, K_T and K_Q respectively:

$$K_T = \frac{T}{\rho \cdot n^2 \cdot D^4} \quad K_Q = \frac{Q}{\rho \cdot n^2 D^5}$$

The efficiency of the propeller (η) is defined in terms of the thrust and torque coefficients as follows:

$$\eta = \frac{J}{2\pi} \frac{K_T}{K_Q}$$

Steady-state solutions were carried out with the native GPU solver assuming incompressible, turbulent flow, with water as the working fluid. Turbulence was modeled using the SST $k-\omega$ turbulence model.

Of particular interest in the present case was the use of the coupled option for pressure-velocity coupling in the GPU solver algorithm. The coupled option (which is new in version 23R2) implicitly couples the continuity and momentum equations, leading to improved robustness. This can be especially helpful for turbomachinery cases since the use of a moving reference frame introduces additional acceleration terms into the momentum equations (which in turn benefit from the continuity-momentum equation coupling).

Contours of static pressure are shown in Figure 12, along with relative pathlines illustrating the swirling flow induced by the propeller motion near the tips of the blades. The lower pressures at the blade tips are indicative of the formation of a tip vortex structure as the flow turns from the pressure side towards the suction side of the blade tip.

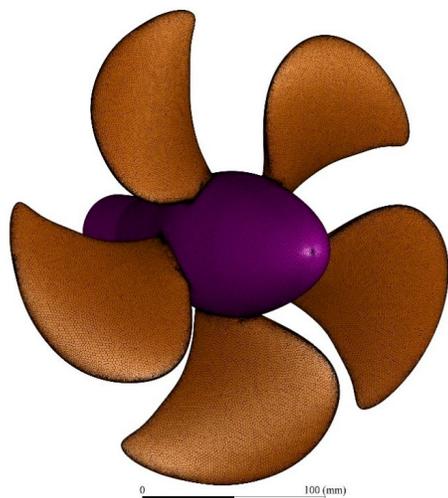


Figure 10. Marine propeller geometry

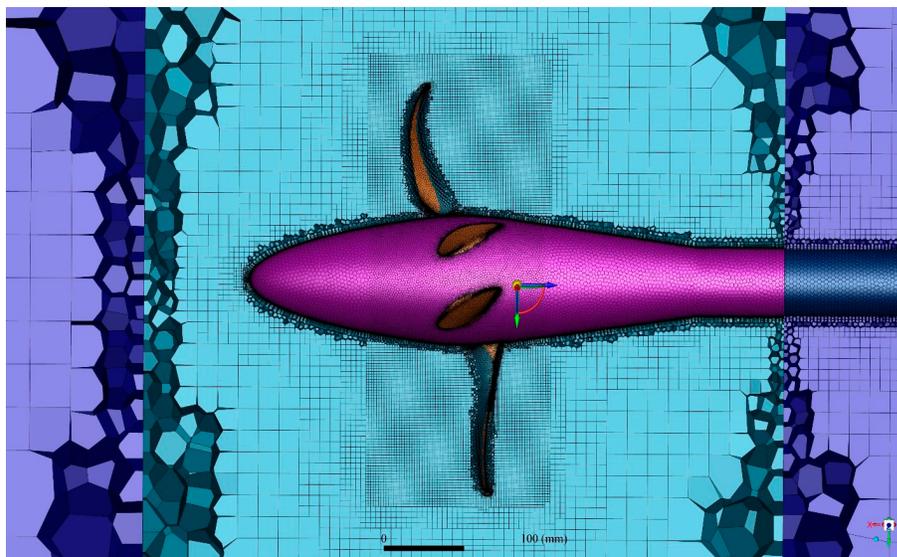


Figure 11. Side view of marine propeller geometry and mesh illustrating the local resolution

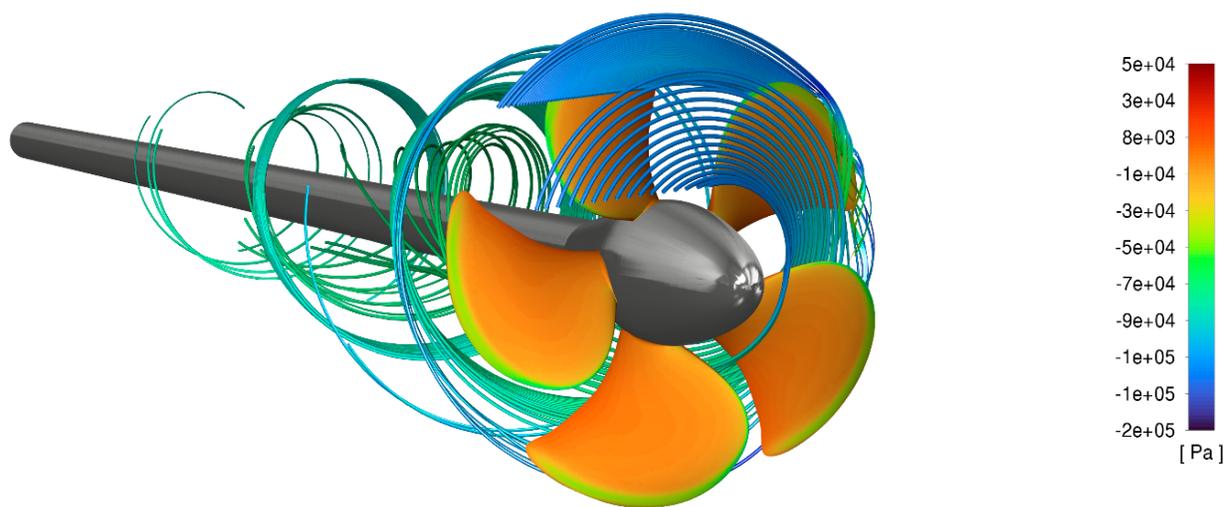


Figure 12. Pressure contours and relative streamlines for the marine propeller case

Quantitative comparisons of the GPU CFD solutions with test data for the Potsdam propeller are shown in Figures 13-15. The thrust and torque values for a range of advance ratios were computed from the predicted fluid force distributions, nondimensionalized, and compared with experimental data. As can be seen, the results are in excellent agreement with the published data, showing how CFD solutions from the GPU solver can accurately predict the linear drop in K_T and K_Q as advance ratio increases.

While the marine propeller model is a standard validation case, what it illustrates is that we are often not concerned with just one operating condition, but rather a range of multiple conditions. This requires computing perhaps 10 or more cases in order to determine the propeller performance – something that would be important if the blade shape were being parametrically optimized for improved efficiency or higher thrust.

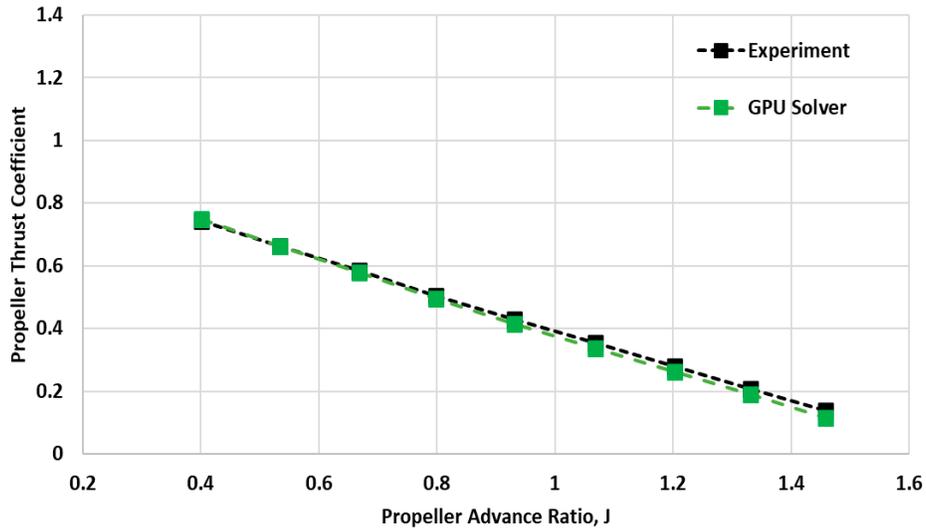


Figure 13. Marine propeller thrust coefficient versus advance ratio: comparison of GPU solutions with test data^[7]

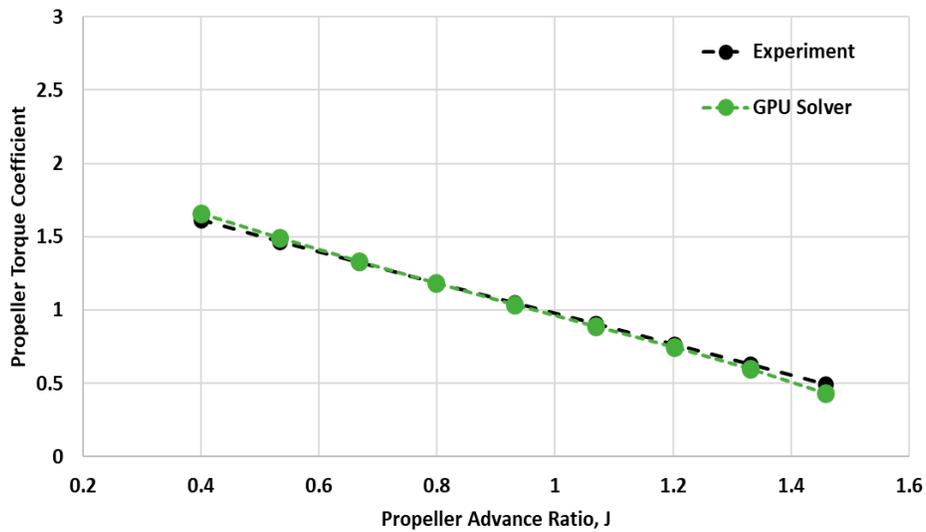


Figure 14. Marine propeller torque coefficient versus advance ratio: comparison of GPU solutions with test data^[7]

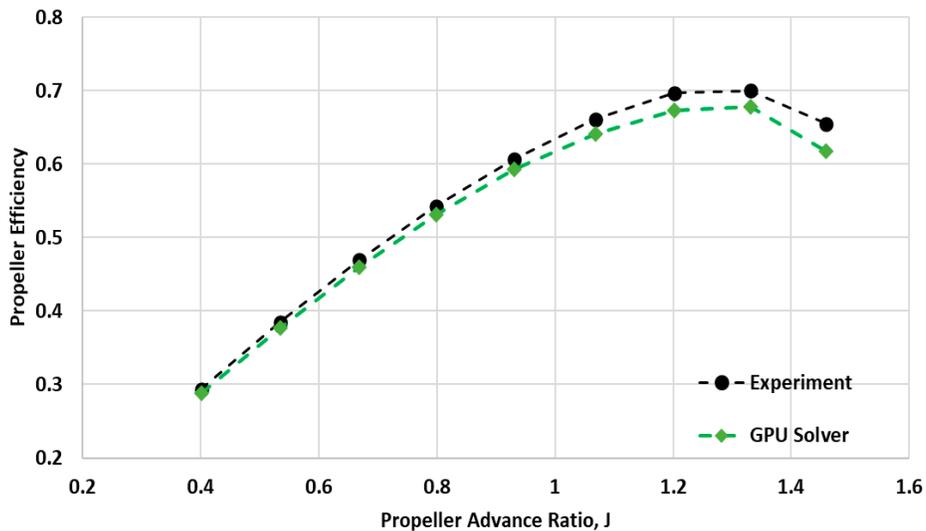


Figure 15. Marine propeller efficiency versus advance ratio: comparison of GPU solutions with test data^[7]

/ Summary

In this paper, we have examined three cases to highlight some of the extensive testing that the ANSYS Fluent native GPU solver has undergone over the past year. Our test program employs well-known test cases from the open literature for both canonical and industrial model tests. In addition, we strive to ensure that the overall accuracy is consistent with our flagship (CPU-based) CFD solver, Ansys Fluent.

One topic which was not discussed (but is of great importance) was the timing and hardware comparisons. In general, our collective experience is that the GPU solver is much faster than the legacy CPU-based solver. This can be thought of in two ways. First, the GPU solver running on one or more GPU cards is many times faster than the CPU solver running on a fixed number of cores. You could also view this as the GPU solver being equivalent to running the parallel CPU solver on a large number of cores. Either way, rest assured – the GPU solver is fast!

For more information on GPU benchmarking and the 2023 R2 release of the GPU solver, please watch the [2023 R2 Fluent on-demand webinar](#).

/ References

- [1] “Speed AND Accuracy – First of a kind broad-spectrum CFD solver built natively on GPUs,” Ansys White Paper, 2022.
- [2] Turton, R.; and Levenspiel, O., “A short note on the drag correlation for spheres,” Powder Technol., 47, 83-86, 1986.
- [3] Baughn, J.W., Hechanova, A.E., and Xiaojun Yan, “An Experimental Study of Entrainment Effects on the Heat Transfer From a Flat Surface to a Heated Circular Impinging Jet,” ASME Journal of Heat Transfer, November 1991, Vol. 113, pp 1023 – 1025.
- [4] Baughn, J.W., and S. Shimizu, “Heat Transfer Measurements From a Surface With Uniform Heat Flux and an Impinging Jet,” ASME Journal of Heat Transfer, November 1989, Vol. 111, pp. 1097 – 1098.
- [5] Cooper, D., Jackson, D.C., Launder, B.E., and G.X. Liao, “Impinging jet studies for turbulence model assessment – I. Flow-field experiments,” International Journal of Heat and Mass Transfer Volume 36, Issue 10, July 1993, pp. 2675-2684.
- [6] Yan, X., Baughn, J.W., and Masood Mesbah, “The Effect of Reynolds Number on the Heat Transfer Distribution from a Flat Plate to an Impinging Jet,” HTD-Vol. 226, Fundamental and Applied Heat Transfer Research of Gas Turbine Engines, ASME, 1992.
- [7] Potsdam Model Basin, “Potsdam Propeller Test Case (PPTC) Open Water Tests with the Model Propeller VP1304,” Report 3752, Potsdam, April 2011.

ANSYS, Inc.
Southpointe
2600 Ansys Drive
Canonsburg, PA 15317
U.S.A.
724-746-3304
ansysinfo@ansys.com

When visionary companies need to know how their world-changing ideas will perform, they close the gap between design and reality with Ansys simulation. For more than 50 years, Ansys software has enabled innovators across industries to push boundaries by using the predictive power of simulation. From sustainable transportation to advanced semiconductors, from satellite systems to life-saving medical devices, the next great leaps in human advancement will be powered by Ansys.

Ansys and any and all ANSYS, Inc. brand, product, service and feature names, logos and slogans are registered trademarks or trademarks of ANSYS, Inc. or its subsidiaries in the United States or other countries. All other brand, product, service and feature names or trademarks are the property of their respective owners.

Visit www.ansys.com for more information.

©2023 ANSYS, Inc. All rights reserved.