Chair of Aerodynamics and Fluid Mechanics
TUM School of Engineering and Design
Technical University of Munich

Integration of NVIDIA Warp Kernels into the Fully-differentiable JAX-Fluids CFD Code

Interdisciplinary Project (IDP)

Contact: Deniz Bezgin (deniz.bezgin@tum.de)

JAX-Fluids is a computational fluid dynamics (CFD) solver for compressible single- and multiphase flows. Built on the JAX library, the entire code base is inherently differentiable, making it ideal for gradient-based optimization tasks. JAX-Fluids utilizes JAX-based parallelization primitives to enable distributed computations across multiple XLA devices, including the computation of automatic differentiation gradients across XLA devices. The solver has demonstrated excellent scaling on up to 512 NVIDIA A100 GPUs and up to 1024 TPU cores.

This project focuses on integrating NVIDIA Warp kernels into the JAX-Fluids solver framework. While JAX's just-in-time compilation significantly increases performance, substituting compute-intensive subroutines with dedicated NVIDIA Warp kernels may further increase computational performance. The workflow of this project includes the following steps:

1. **Comprehensive profiling**: Profiling the current codebase to identify the most compute-intensive subroutines for both single- and multiphase simulations.
2. **Kernel substitution**: Replacing identified subroutines with optimized NVIDIA Warp kernel implementations.
3. **Performance Benchmarking**: Comparing the performance of the hybrid JAX-NVIDIA Warp implementation against the purely JAX-based version, including serial and parallel computations.
4. **Testing differentiability**: An essential goal is to ensure that the hybrid codebase maintains full differentiability, allowing seamless gradient computations across both JAX and NVIDIA Warp components.

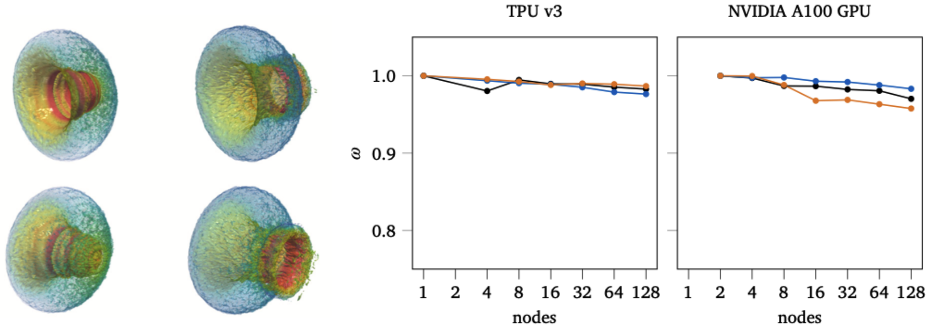

Figure 1: Left: Simulation result of a shock-bubble interaction. Right: Weak scaling efficiency of JAX-Fluids. Pictures taken from Bezgin et al. 2025.

Requirements

- Programming experience in Python.
- Interest in HPC, numerical solution of partial differential equations, and CFD.
- Beneficial: Programming experience in JAX, Advanced knowledge of fluid dynamics (e.g., Gas Dynamics, Applied CFD, Turbulent Flows, Numerical Methods for Conservation Laws).

Application

Please send an up-to-date transcript of records to deniz.bezgin@tum.de