ExaWind: Then and Now

NREL: M.A. Sprague (PI), S. Bidadi, M. Brazell, N. Brunhart-Lupo, M. Henry de Frahan, J. Rood, A. Sharma, I. Topcuoglu, G. Vijayakumar
SNL: P. Crozier, L. Berger-Vergiat, D. Dement, N. de Velder, J. Hu, R. Knaus, D.H. Lee, N. Matula, J. Overfelt, P. Sakievich, T. Smith, A. Williams
ORNL: A. Prokopenko
UTA: R. Moser, J. Melvin

ExaWind Objective: To create an open-source predictive physics-based simulation capability that will provide a validated "ground truth" foundation for wind plant siting, operational controls, and reliably integrating wind energy into the grid

ExaWind Motivation: Validated, predictive wind plant simulations will reduce the cost of energy by providing a path to a better understanding of wind plant flow physics, which will lead to

- new plant layout design in complex terrain
- new turbine technologies to optimize plant performance
- a foundation for improved computer-aided engineering models, which will enable better design optimization

ExaWind Partnership: Development has been jointly funded by the U.S. Department of Energy (DOE) Exascale Computing Project (ECP) and the DOE Wind Energy Technologies Office (WETO) since 2016

ExaWind Challenge Problem: Predictive simulation of a wind farm with tens of megawatt-scale wind turbines dispersed over an area of 50 square kilometers

Minimum Requirements: 2 × 2 array of megawatt-scale turbines, 3 km × 3 km × 1 km domain, at least 30-billion grid points

Then (2016)

Approach: Create computational fluid/structure dynamics (CFD and CSD) codes for Reynolds-averaged Navier-Stokes (RANS)/large-eddy simulations (LES) where wind turbine geometry and blade boundary layers are resolved and include moving meshes, fluid-structure interaction, and atmospheric turbulence

Starting-Point Codes:

Nalu: https://github.com/nalucfd/

- Unstructured-grid, incompressible-flow CFD
- LES turbulence model
- C/C++
- Built on Trilinos STK, Tpetra/Belos/MueLu solvers, and Kokkos
- Mesh rotation achieved through a sliding-mesh interface

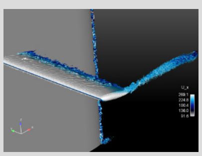
OpenFAST: https://github.com/openfast/

- Whole-turbine simulation code (structural dynamics, control)
- Fortran90

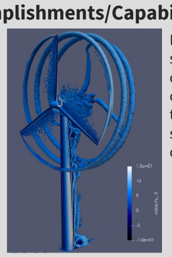
Challenges:

- Target problem requires resolving spatial scales going from blade boundary layers (e.g., 10^{-5} m) to the wind farm domain (e.g., 10^{3} m), i.e., at least eight orders of magnitude
- Finite volumes with extreme aspect ratios (e.g., 10,000), which are necessary for hybrid-RANS/LES, were a serious challenge linear-system solvers
- Time-integration scheme required impractically small time-step sizes (e.g., 10^{-6} s) for production simulations
- Sliding-mesh approach presented mesh-creation challenges and no clear pathway for yaw motions

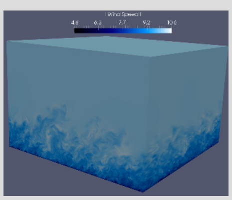
Then (2017): Nalu Example Accomplishments/Capabilities



Nalu simulation of the McAlister-Takahashi blade, which has extensive wind-tunnel experimental data. 68- and 300-million grid point meshes examined, with RANS and LES.



Nalu LES results for a fully resolved kilowatt-scale turbine in uniform flow; isosurfaces of Q-criterion colored by the streamwise-component of velocity; mesh motion achieved through a sliding-mesh interface; for a simulation with 45-million grid points on 4096 cores, 48-hour wall-clock time per rotation.



Nalu LES results for a neutrally stable atmospheric boundary layer (ABL).

Systems Employed

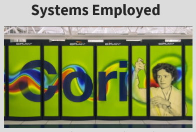


Image: DOE, National Energy Research Scientific Computing Center (NERSC)

Image: DOE, National Renewable Energy Laboratory

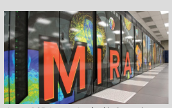
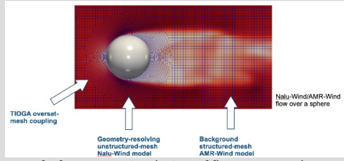


Image: DOE, Argonne Leadership Computing Facility (ALCF)

Now (2023)

Shift in Approach: Added AMR-Wind as a background solver and made Nalu-Wind the near-body solver; coupling via overset meshes



Proof-of-concept simulation of flow over a sphere using the hybrid Nalu-Wind/AMR-Wind solver.

Primary Application Codes:

Nalu-Wind

- https://github.com/exawind/nalu-wind
- Wind-specific offshoot from Nalu; primarily used for near-body flows
- *hypre* is primary linear-system-solver package
- Hybrid-RANS/LES with time integrator that enables practical time step sizes
- Overset meshes (via TIOGA, https://github.com/jsitaraman/tioga) is primary method for moving meshes
- Performant on NVIDIA GPUs; Advanced Micro Devices, Inc. (AMD) GPUs are in progress

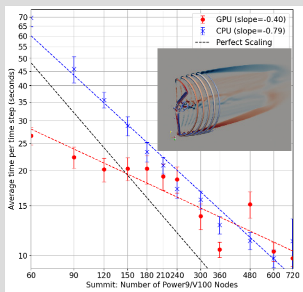
AMR-Wind

- https://github.com/exawind/amr-wind
- Structured-grid adaptive mesh refinement (AMR) CFD code; background solver
- C++ and built on the AMReX library
- Performant on NVIDIA and AMD GPUs

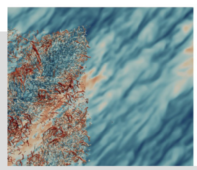
OpenFAST

- No pathway to support parallelization or GPUs
- Starting new FY23 WETO project to create replacement: OpenTurbine https://github.com/exawind/openturbine

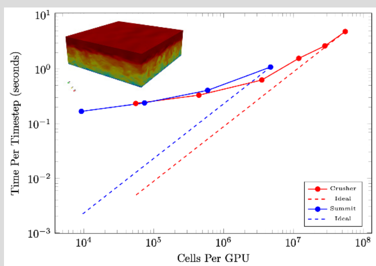
Now (2023): ExaWind Example Accomplishments/Capabilities



Nalu-Wind strong-scaling results for blade-resolved simulations of the a 5-megawatt turbine on the Summit supercomputer; model had 640-million grid points; GPU and CPU calculations used all GPUs and all CPU cores, respectively, on each node.



Hybrid AMR-Wind/Nalu-Wind simulation of a small wind farm in a stable ABL; isosurfaces of Q-criterion colored by velocity magnitude; for a simulation with 650-million grid points on 6046 CPU cores, 3.6-hour wall-clock time per rotation.



AMR-Wind strong-scaling plot for ABL simulations on Summit (NVIDIA) and Crusher (AMD) GPUs. Simulations of different model sizes were run on a fixed resource of 16 Crusher nodes (128 GCDs) or 16 Summit nodes (96 GPUs). The largest models were 0.5- and 7-billion grid points on Summit and Crusher, respectively.

Takeaways

- The shift to a new Nalu-Wind time integration scheme and overset-enabled loose coupling between AMR-Wind and Nalu-Wind was a game changer
- Blade-resolved wind farm simulations that span eight orders of magnitude in spatial scales have been achieved with practical simulation times and up to 10-billion grid points
- Nalu-Wind and AMR-Wind on Summit GPUs outperform CPUs by a wide margin for many grid points per GPU
- Nalu-Wind GPU strong-scaling limit is about 300,000 grid points per Summit GPU
- AMR-Wind on Crusher scales well to about 2-million cells per AMD Graphics Complex Die (GCD)

Key Remaining Challenges

- Fluid-structure interaction capabilities are currently being tested; required for challenge problem
- Porting Nalu-Wind to Crusher/Frontier AMD GPUs has been challenging, e.g., building and linking Nalu-Wind dependencies on Crusher with HIP relocatable device code

Ongoing and Future Directions

Reduce time to solution, tighten coupling between Nalu-Wind and AMR-Wind, couple to the mesoscale Energy Research and Forecasting (ERF) code; offshore wind energy (two-phase fluid dynamics, floating-platform offshore turbines), concentrated solar power

This work was authored in part by the National Renewable Energy Laboratory (NREL), operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes. This poster is published as NREL technical report, NREL/PO-5000-84980.

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.This report followed the Sandia National Laboratories formal review and approval process (SAND2023-10860C). As such, the technical report is suitable for unlimited release.

This research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of two U.S. Department of Energy (DOE) organizations (Office of Science and the National Nuclear Security Administration) responsible for the planning and preparation of a capable exascale ecosystem, including software, applications, hardware, advanced system engineering, and early testbed platforms, in support of the nation's exascale computing imperative.

The research was performed using computational resources sponsored by DOE Office of Energy Efficiency and Renewable Energy and located at National Renewable Energy Laboratory and resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the DOE Office of Science under Contract No. DE-AC05-00OR22725.

Exascale Computing Project Annual Meeting
Royal Sonesta Galleria, Houston, Texas
Jan. 17 – 20, 2023