ExaWind: Exascale Predictive Wind Plant Flow Physics Modeling

NREL: M.A. Sprague (PI), S. Ananthan, R. Binyahib, M. Brazell, M. Henry de Frahan, R. King, P. Mullowney, J. Rood, A. Sharma, S. Thomas, G. Vijayakumar

SNL: P. Crozier, L. Berger-Vergiat, L. Cheung, D. Dement, N. de Velder, D. Glaze, J. Hu, R. Knaus, D. Lee, N. Matula, T. Okusanya, J. Overfelt, S. Rajamanickam, P. Sakievich, T. Smith, J. Vo, A. Williams, I. Yamazaki

ORNL: J. Turner, A. Prokopenko, R. Wilson

UTA: R. Moser, J. Melvin

Parallel Geometric Algorithms: J. Sitaraman

NREL/PO-5000-80015

Project Overview

Objective: Create a predictive physics-based simulation capability that will provide a validated "ground truth" foundation for siting and operational controls of wind plants, and the reliable integration of wind energy into the grid

Motivation: Validated, predictive wind plant simulations will reduce the cost of energy by providing:

- a path to better understanding of wind plant flow physics, which will lead to
  - new plant layout design for complex terrain
  - new wind turbine technology to optimize plant performance
- a foundation for improved computer-aided engineering models, which will enable better design optimization

Primary Application Codes:

- **Nalu-Wind**
  - https://github.com/exawind/nalu-wind
  - Unstructured-grid computational fluid dynamics (CFD) code
  - C/C++
  - Built on Trilinos, Sierra Toolkit (STK), *hypre*, Topology Independent Overset Grid Assembler (TIOGA)
- **AMR-Wind**
  - https://github.com/exawind/amr-wind
  - Structured-grid adaptive-mesh-refinement (AMR) CFD code
  - C/C++
  - Built on the AMReX library
- **OpenFAST**
  - https://github.com/openfast/openfast
  - Whole-turbine simulation code; blades, control system, tower, etc.
  - Fortran 90; dedicated Intel Parallel Computing Center (IPCC) for parallelization

ExaWind Software/Library Partnerships

- ***Trilinos***, https://trilinos.org/
  - ***MueLu:*** Aggregation-based multigrid preconditioners
  - ***Ifpack2:*** SOR-based, polynomial and incomplete factorization preconditioners
  - ***Kokkos-Kernels:*** Shared-memory algorithms: graph-coloring, SpMV, SPMM, iterative and incomplete factorization preconditioners
  - ***Tpetra:*** Distributed memory, sparse linear algebra objects
  - ***Belos:*** Templated Krylov and recycling solvers
  - ***Amesos2:*** Sparse direct solvers
  - ***STK:*** Unstructured-mesh in-memory, parallel-distributed database as well as I/O, load-balancing, proximity search, transfers, etc.
- ***hypre***, https://github.com/LLNL/hypre
  - Multigrid solvers and preconditioners based on classic Ruge-Stüben algebraic multigrid (AMG) algorithm
- ***Kokkos***, https://github.com/kokkos
  - Programming model in C++ for writing performance-portable applications targeting all major HPC platforms
- ***TIOGA***, https://github.com/jsitaraman/tioga
  - Library for overset-grid assembly on parallel distributed systems
- **ALExa / ArborX**, https://github.com/arborx/ArborX
  - Performance-portable algorithms for geometric search
- ***VTK-m***, https://gitlab.kitware.com/vtk/vtk-m
  - *In situ* visualization and analysis capabilities
- ***Spack***, https://github.com/spack/spack
  - Package manager for exascale software
- ***AMReX***, https://github.com/AMReX-Codes/amrex
  - Software framework for block-structured AMR

ECP Key Performance Parameter (KPP-2)

Challenge Problem:

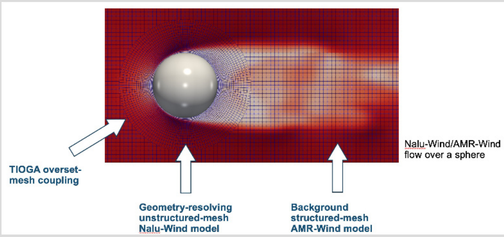
Predictive simulation of a wind farm with tens of megawatt-scale wind turbines dispersed over an area of 50 square kilometers

Minimum Requirements:

- 2x2 array of megawatt-scale turbines operating at rated speed
- 3 km x 3 km domain with height of 1 km
- Hybrid-RANS/LES model
- At least 30-billion grid points

2020-2021 Highlights

Established new hybrid Nalu-Wind/AMR-Wind solver strategy: Leveraging the best of structured & unstructured grids (FY21 Q2 Milestone)

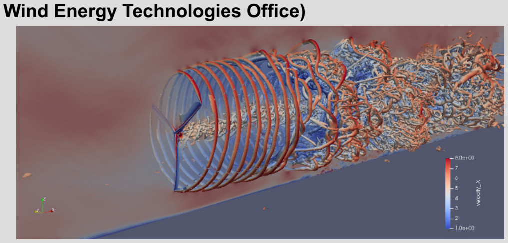


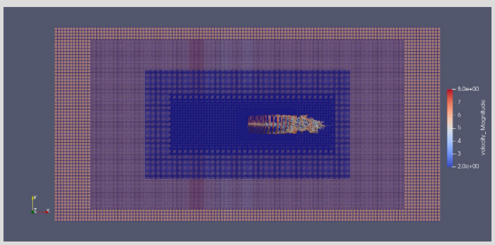
Flow over a sphere proof-of-concept simulation. The near-body flow is resolved with an unstructured-grid Nalu-Wind model and the background flow is resolved with a structured-grid AMR-Wind model. The models are coupled through overset meshes handled by TIOGA.

The hybrid-solver approach:

- Relies on loosely coupled solves of global linear systems (FY20 Q3 milestone: "Develop and compare algorithms for decoupling of global linear systems in Nalu-Wind solver within an overset mesh paradigm")
- Provides "optimal" solvers for the different grids and the decomposition provides a new pathway to overcoming weak-scaling challenges
- Removes constraint equations from linear systems that degrade linear-system-solver performance
- Avoids the need to re-initialize linear systems in Nalu-Wind due to moving meshes

New hybrid solver enables validation-quality blade-resolved turbine simulations (in collaboration with DOE Wind Energy Technologies Office)



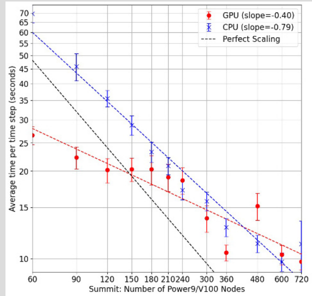


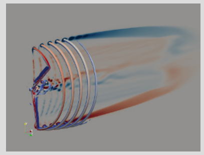
AMR-Wind/Nalu-Wind simulation of the NM-80 DanAero wind turbine in turbulent flow with 122M grid points. Simulations were performed as part of the IEA Task 29 validation campaign. Simulation performed on NREL's Eagle computer.

This simulation demonstrates:

- Coupling AMR-Wind and Nalu-Wind
- Coupling hybrid-RANS/LES models
- Moving meshes
- Wake generation and evolution in turbulent inflow

Optimized Nalu-Wind/*hypre* simulations performed on over 4000 Summit GPUs

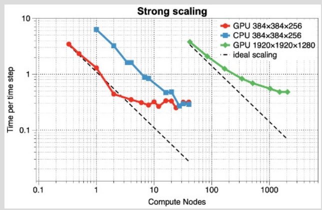


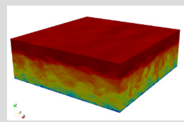


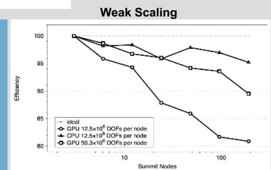
Strong-scaling results for blade-resolved simulations of the NREL 5-MW turbine on Summit. The model has 640M grid points. GPU and CPU calculations used all GPUs and CPU-cores, respectively, on each node.

- Nalu-Wind-only simulations employed loosely coupled linear systems
- Novel preconditioner, AMG setup, and matrix assembly algorithms developed specifically to handle Nalu-Wind unstructured-mesh simulations
- Highly tuned, specialized GPU algorithms required for good performance at this scale
- GPUs outperform CPUs by a wide margin for many grid points per GPU
- GPU strong-scaling limit is about 300k grid points per GPU

AMR-Wind strong/weak-scaling atmospheric-boundary-layer (ABL) simulations on Summit reach billions of grid points







- GPUs are faster than CPUs with enough grid-points per GPU
- Excellent strong scaling down to 3M grid-points per GPU
- AMR-Wind shows good weak scaling (2M & 8M grid points per GPU shown)
- Largest simulation to date was 51.5B grid points and it was run on almost 90% of the Summit GPUs

ExaWind software at core of industry collaboration with GE

- Atlantic coast low-level jet (LLJ) and impact on offshore wind farm performance project
  - New York State Energy Research and Development Authority (NYSERDA) project
  - GE-NREL Partnership
- High-fidelity modeling toolkit for wind resource characterization and wind farm development
  - DOE Technology Commercialization Fund (TCF) Project
  - NREL-SNL-GE Partnership

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 (SAND2021-4472 C). 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 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.

Exascale Computing Project Annual Meeting
Virtual
April 12-16, 2021