SAND2020-4088R

NREL
NATIONAL RENEWABLE ENERGY LABORATORY

A2e High-Fidelity Modeling (HFM) Project

Overview of FY20 Q2 milestone completion:

Actuator Disk Improvements and Hardening

13 April 2020

Philip Sakievich, Robert Knaus, Alan Hsieh, Lawrence Cheung, Myra Blaylock, David Maniaci, Luis 'Tony' Martinez-Tossas, Matthew J Churchfield

NREL is a national laboratory of the U.S. Department of Energy, Office of Energy Efficiency and Renewable Energy, operated by the Alliance for Sustainable Energy, LLC.

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.

FY20 Q2 Milestone Description

Enhance Nalu-Wind's actuator disc model through hardening, documenting, stress-testing, verifying, and validating. Existing workflows will be improved by reducing the data output stream, and by making the analysis capabilities more modular and generally better. These model capabilities are needed by other A2e areas, namely Wake Dynamics, AWAKEN, and VV&UQ.

Milestone Team:

SNL: Philip Sakievich (lead), Robert Knaus, Alan Hsieh, Lawrence Cheung, Myra Blaylock, David Maniaci

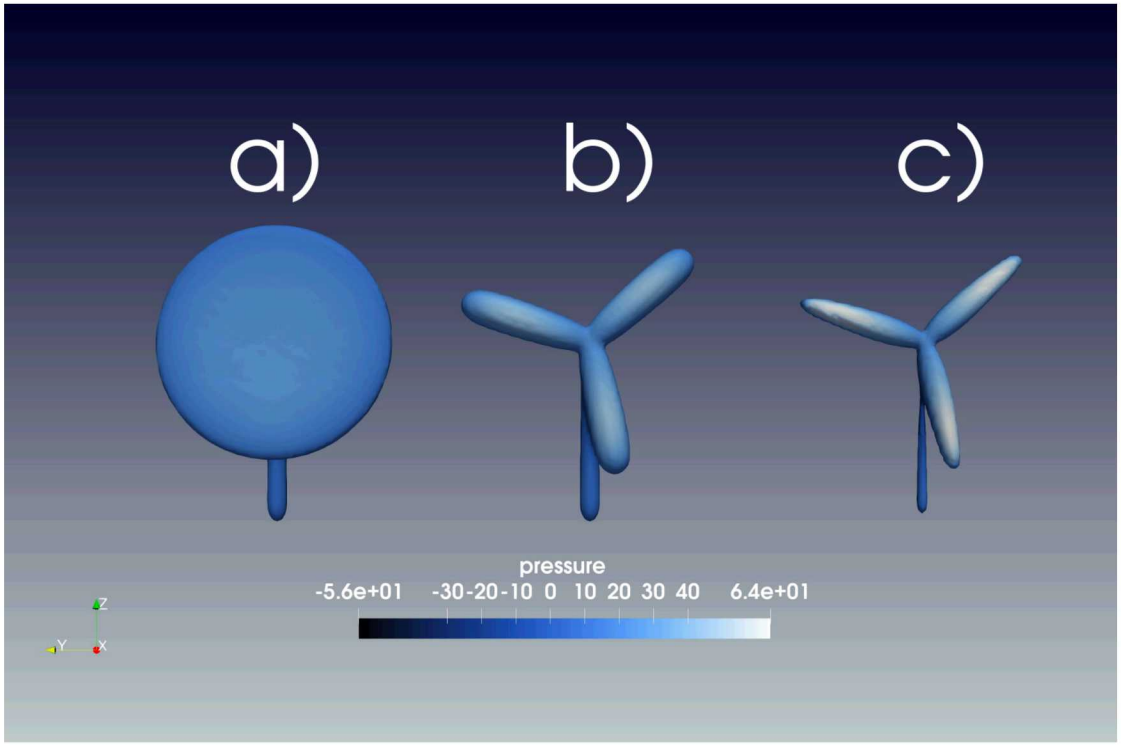
NREL: Luis 'Tony' Martinez-Tossas, Matthew J. Churchfield

FY20 Q2 Milestone Goals, Background, & Team

- **Main goals in milestone:**
  - Make the actuator disk model (ADM) production ready in Nalu-Wind by improving its scalability and software quality
  - Add additional testing, verification and validation to the actuator models (line and disk) in Nalu-Wind and update the code documentation with these results
  - Document performance changes resulting from the code refactor
  - Outline the performance capabilities of the actuator disk model in simulations with coarse meshes and large time-steps relative to the actuator line model
- **Background:**
  - The actuator disk capability in Nalu-Wind is relatively new, and has not been utilized very much despite being one of the most heavily used turbine models in the wind community. This gap needs to be closed as Nalu-Wind's user base grows
  - A scalability issue under mesh refinement for the ADM was identified during the preliminary scoping of this milestone
    - The original actuator models (line and disk) ghosted all elements interacting with a turbine to a single "owning" rank. Since the ADM's disk intersects a large number of elements this leads to a dramatic load imbalance and memory overload under mesh refinement.
  - The scalability issue coupled with the need to make the code extensible to next generation platforms (NGPs) necessitated a complete refactor of the actuator code base within Nalu-Wind

Background: Actuator Models in Nalu-Wind

- Nalu-Wind has a model hierarchy for actuator methods: actuator disk, actuator line and the advanced actuator line methods (ADM, ALM, and AALM respectively)
- Each level of the hierarchy adds additional fidelity to the model, but requires increased resolution in time and/or space



Examples of the actuator disk (a), actuator line (b), advanced actuator line (c) models by plotting an isosurface of the actuator source term

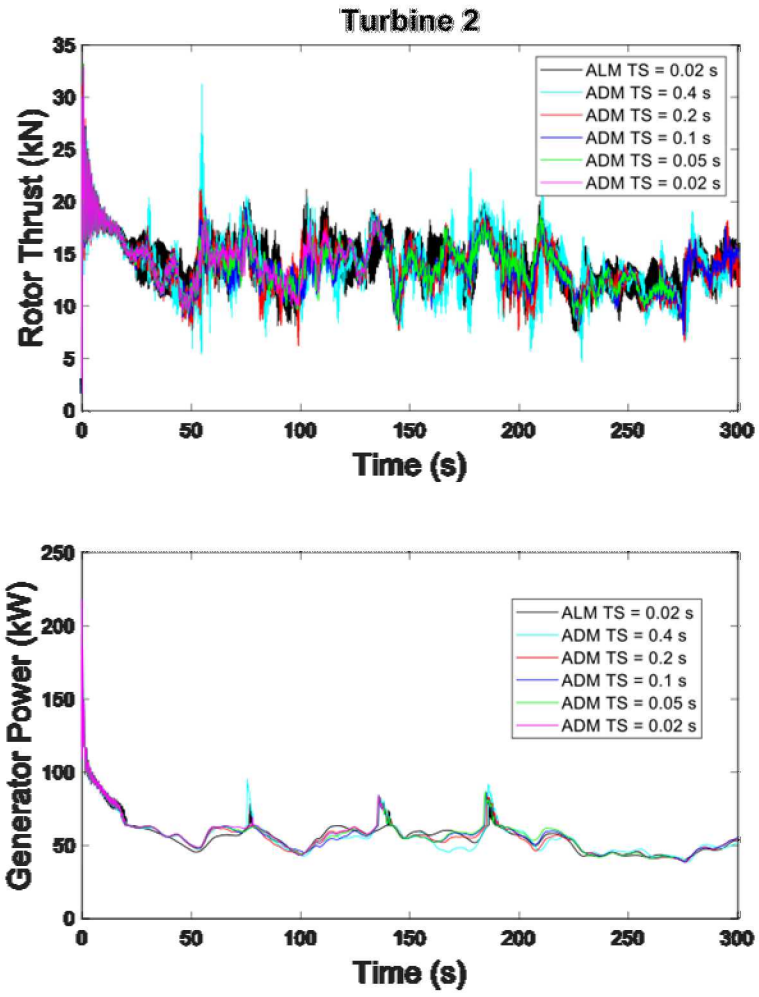
a) ADM models the rotational path of a turbine's blades as a disk and can be considered a time averaged approximation

b) ALM models each individual turbine blade to better capture details of the wake

c) AALM adds additional features to the ALM such as chord scaling to allow for a more precise representation of the blade geometry

Advantages of the Actuator Disk Method

- ADM can be used in coarser meshes due to its coarser representation of a turbine
- ADM results are consistent regardless of time step
  - ~1% change in generated power between dt = 0.02 and 0.4 seconds
- ADM results for upwind turbine are very similar to the ALM results for generated power (within ~1%)
  - Downwind turbine shows greater discrepancy (~2-3%), likely due to AD smoothing of wake features
- Main time step constraint for ADM becomes stability of the numerical method, instead of the turbine's tip speed ($\Delta t_{ALM} = \Delta x / U_{tip}$)
  - AD model allows a ~20x increase in time step compared to AL model
- ADM is attractive for UQ and industry
  - Experienced no errors during a multi-level-multi-fidelity UQ pilot study for two separate meshes
  - Fifteen samples of five varying turbine input parameters



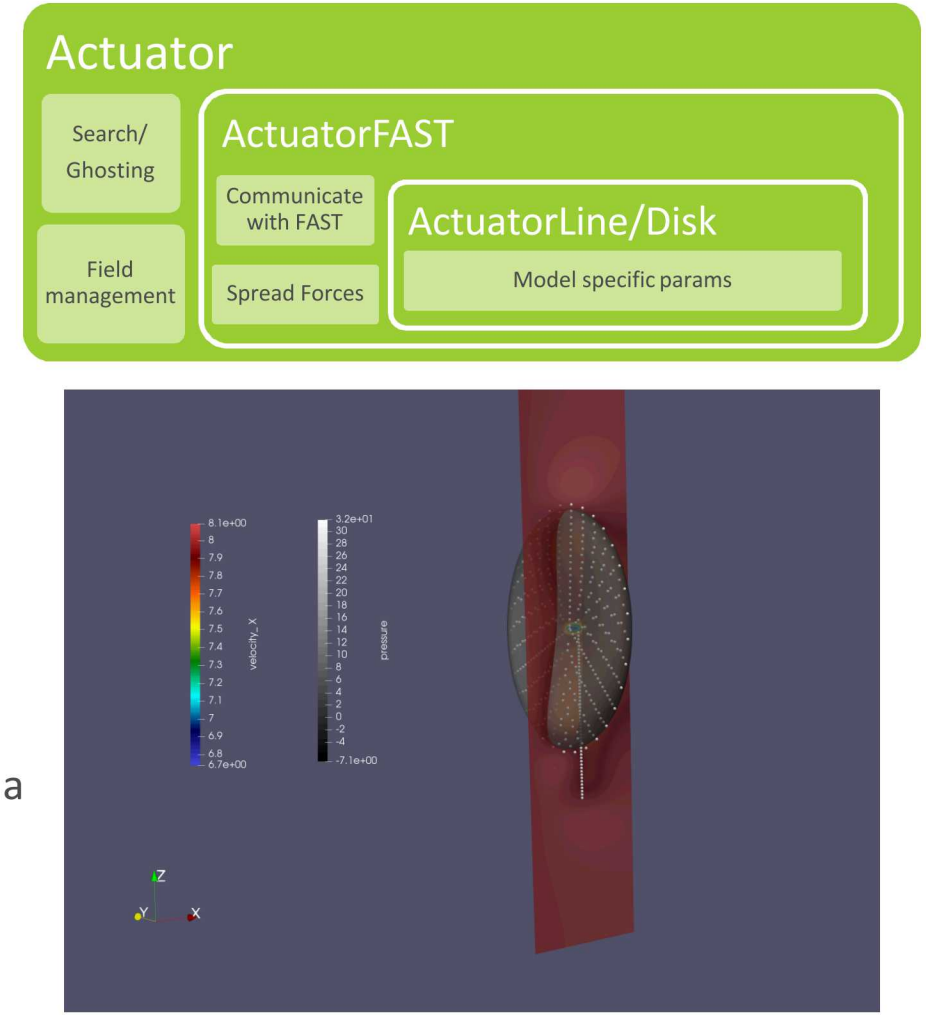
Comparison of integral quantities for ALM and ADM using different timesteps at the second turbine in a 2 turbine ABL simulation

ATMOSPHERE TO ELECTRONS
U.S. DEPARTMENT OF ENERGY

Old ADM Implementation and Scaling Descriptions

Implementation Overview:

- Built around OpenFAST actuator lines
  - ~80-90% + code reuse between ADM/ALM
  - 1 coarse search at start
  - Create additional points
  - Adjust force calculation
- Force from ALM points summed and distributed evenly at each radius to create a disk
- One line change in input deck to switch from ALM to ADM

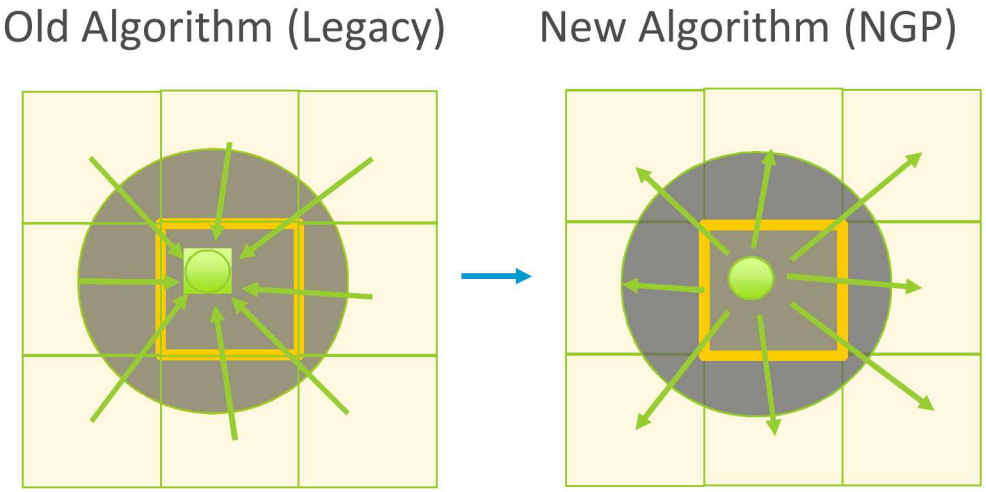


Block diagram of code description (top) and a sample image of ADM displaying actuator points, an iso-surface of the actuator force and contour of velocity (bottom)

Scaling Issue:

- Scaling issue is due to parallel communication pattern
- All elements intersecting the Gaussian to a certain radius are ghosted to single rank
- Ghosting causes load and memory imbalance across ranks leading to a catastrophic bottle neck under mesh refinement
- Scaling issue is also present for the ALM method, but is less pronounced due to far fewer actuator points

Explanation of Code Refactor



- Initial algorithm was unscalable
  - All cells influenced by a point communicated to a unique owner
  - Initial algorithm relied heavily on C++'s standard template library (STL) which is not portable to all next generation platforms (NGP) such as execution on GPUs
- New algorithm distributed with all-reduce communication for actuator data
  - Even load balancing
  - Local only search
  - Concept of code reuse between ALM and ADM maintained
  - New algorithm utilizes Kokkos to allow for NGP execution and performance portability

Explanation of Code Refactor

- New code broken into working chunks that minimize dependencies and can be unit tested
- Code designed to be extendible for models that don't need to rely on OpenFAST
  - Advantageous for new research developments and incoming industry partners
- Utilizes integration point-based assembly providing increased accuracy and robustness
- Keep legacy code in place for comparative testing and deprecation
  - Discovered and fixed a noteworthy bug in the legacy ADM during the refactor process
- Changes made to improve OpenFAST code base as well

Code Example

```
void
SpreadForceInnerLoop::operator()(
  const uint64_t pointId,
  const double* nodeCoords,
  double* sourceTerm,
  const double dual_vol,
  const double scvIp) const
{

  auto pointCoords =
    Kokkos::subview(actBulk_.pointCentroid_.view_host(), pointId, Kokkos::ALL);

  auto pointForce =
    Kokkos::subview(actBulk_.actuatorForce_.view_host(), pointId, Kokkos::ALL);

  auto epsilon =
    Kokkos::subview(actBulk_.epsilon_.view_host(), pointId, Kokkos::ALL);

  double distance[3];
  double projectedForce[3];

  actuator_utils::compute_distance(
    3, nodeCoords, pointCoords.data(), &distance[0]);

  const double gauss =
    actuator_utils::Gaussian_projection(3, &distance[0], epsilon.data());

  for (int j = 0; j < 3; j++) {
    projectedForce[j] = gauss * pointForce(j);
  }

  for (int j = 0; j < 3; j++) {
    sourceTerm[j] += projectedForce[j] * scvIp / dual_vol;
  }
}
```

Summary of Additional Testing

- Unit testing
  - Unit tests added to all major computational kernels (30+ new unit tests)
  - New unit tests also cover OpenFAST API, and parsing to ensure user input errors receive specific Nalu-Wind generated error messages
  - Prior implementation only had 2 unit tests for peripheral calculations
- Regression tests
  - Changed current ALM and ADM tests to use the new code base
  - Add anisotropic Gaussian ALM test using new code base
  - AALM chord scaling test added using the legacy code base
    - AALM still needs to be converted to the new code base, see Next Steps on final slide
- Verification & Validation
  - Full scale ABL with ADM
  - Fixed wing

Fixed Wing Verification Problem

Create a simple problem which can be used for quick verification of actuator computations

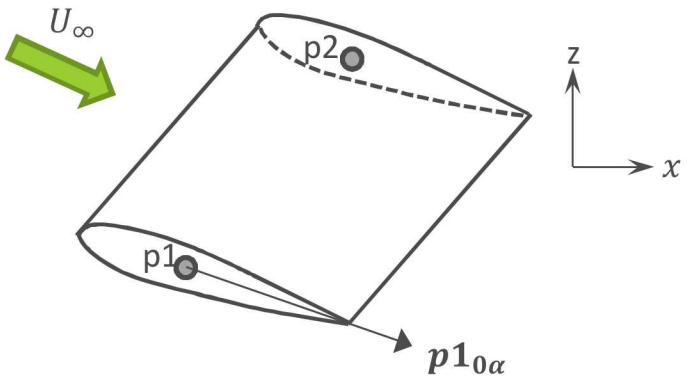
- Choose fixed wing with simple physics
  - Rigid body -- no aero-elastic coupling
  - No blade motion
  - No FAST dependencies
- Airfoil aerodynamics computed from polar tables
  - Lift and drag at each section defined by

$$L = \frac{1}{2}\rho U^2 C_L(\alpha) A_{blade}$$

$$D = \frac{1}{2}\rho U^2 C_D(\alpha) A_{blade}$$

  - Forces calculation can be verified by hand
- Both NGP and non-NGP versions implemented

➢ Can build additional validation cases from fixed wing problem



Input parameters

```
actuator:
   type: ActLineSimple
   search_method: stk_kdtree
   search_target_part: Unspecified-2-HEX
   n_simpleblades:  1
   n_turbines_glob: 0
   Blade0:
     num_force_pts_blade: 10       # Number of blade stations
     epsilon: [3, 3, 3]
     p1: [0, -4, 1]                # Start of blade
     p2: [0,  4, 1]                # End of blade
     p1_zero_alpha_dir: [1, 0, 0]  # Direction of zero AOA at p1
                                   #    measured from LE to TE
     chord_table: [1.0, 1.0, 1.0]  # Chord at each blade station.
     twist_table: [0.0]            # Twist at each blade station.
     aoa_table: [-180, 0, 180]     # AOA for polar table
     cl_table:  [-19.739,0,19.739] # CL's for polar table
     cd_table:  [0]                # CD's for polar table
```

Fixed Wing Verification Problem

Using a simple 2D flat plate, with linear lift curve:

$$C_L = 2\pi\alpha$$
$$C_D = 0$$

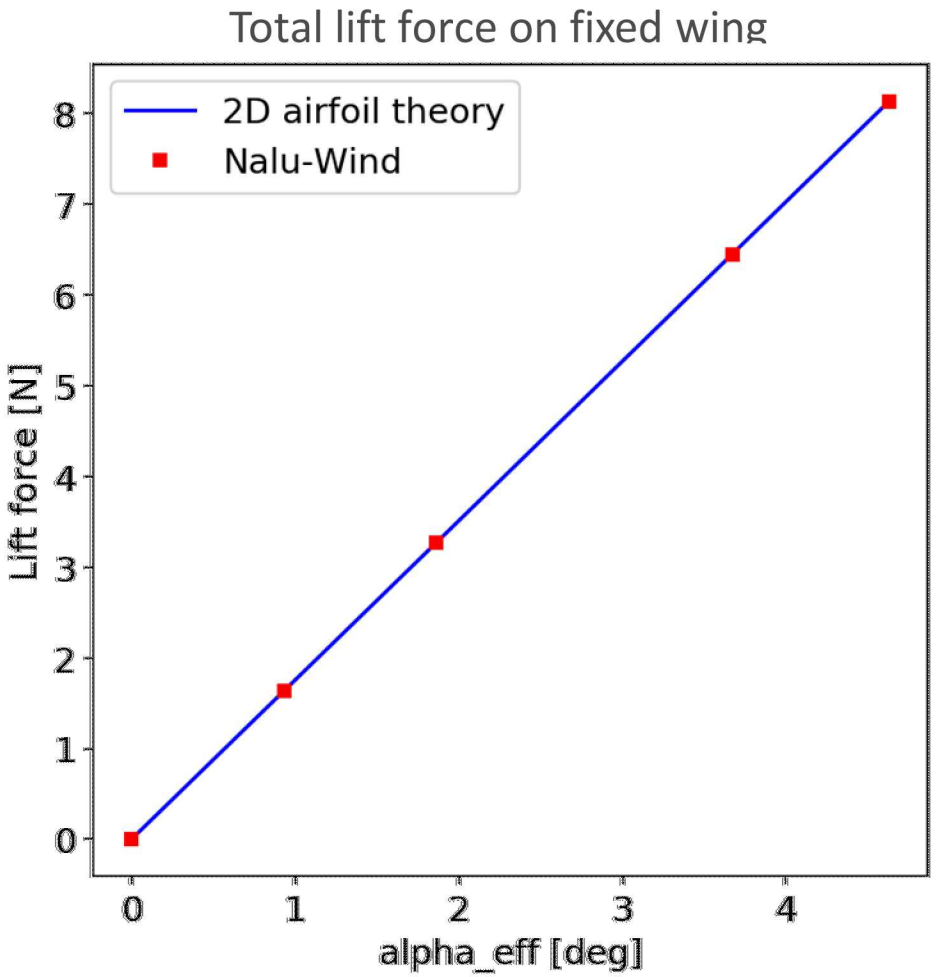
Isotropic force spreading

$$\varepsilon = 3$$

Blade parameters

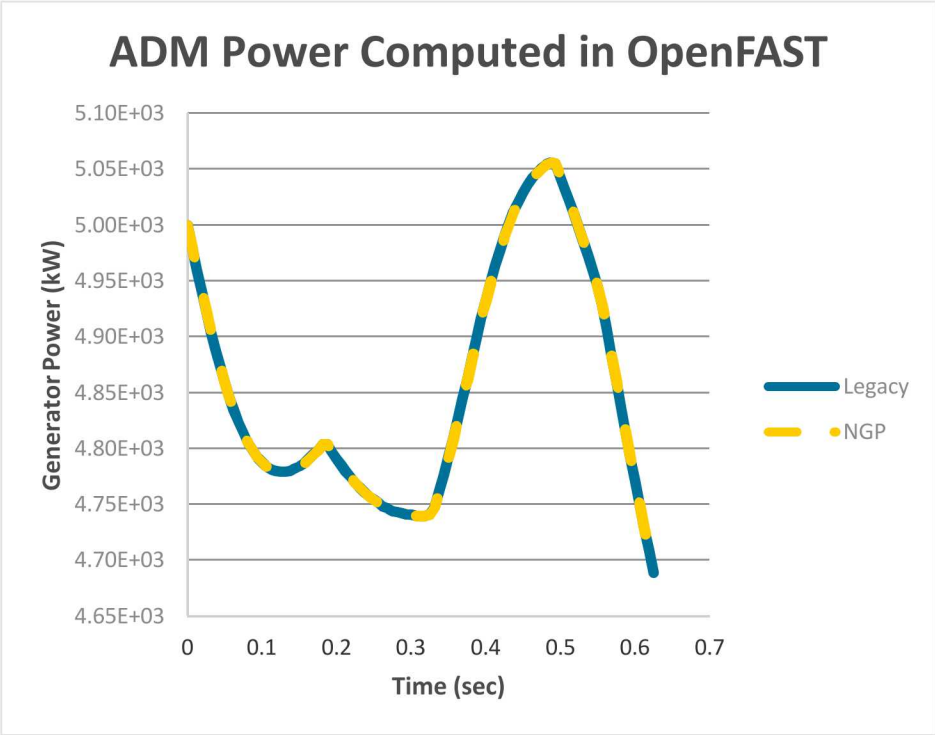
- Chord = 1m
- Span = 8m
- 20 blade sections

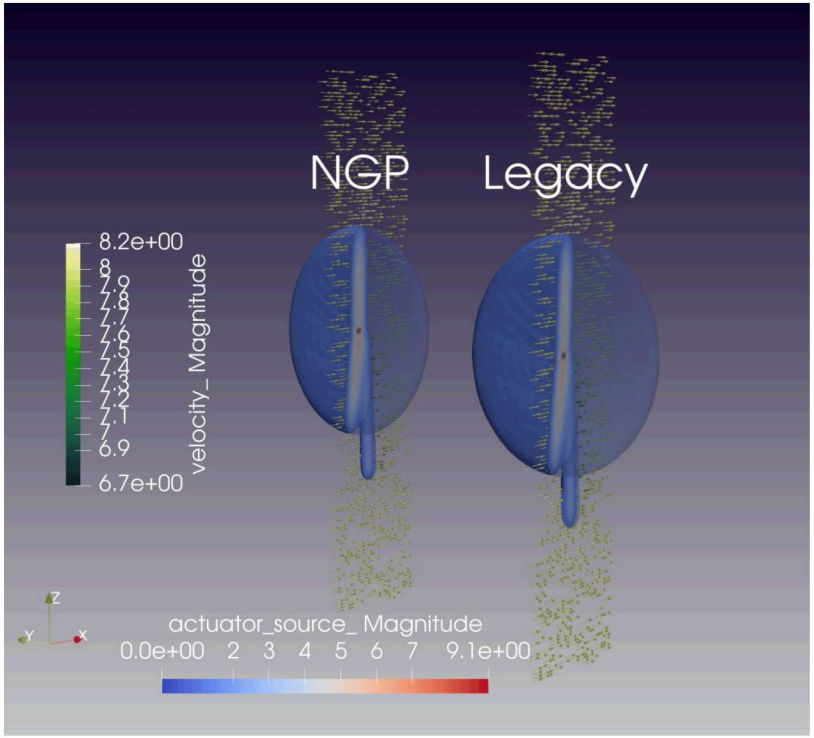
All lift errors < 0.1%



Nalu-Wind's actuator model is able to reproduce the theoretical lift force for a fixed wing with variable angle-of-attack

Old vs New Model: Single Turbine

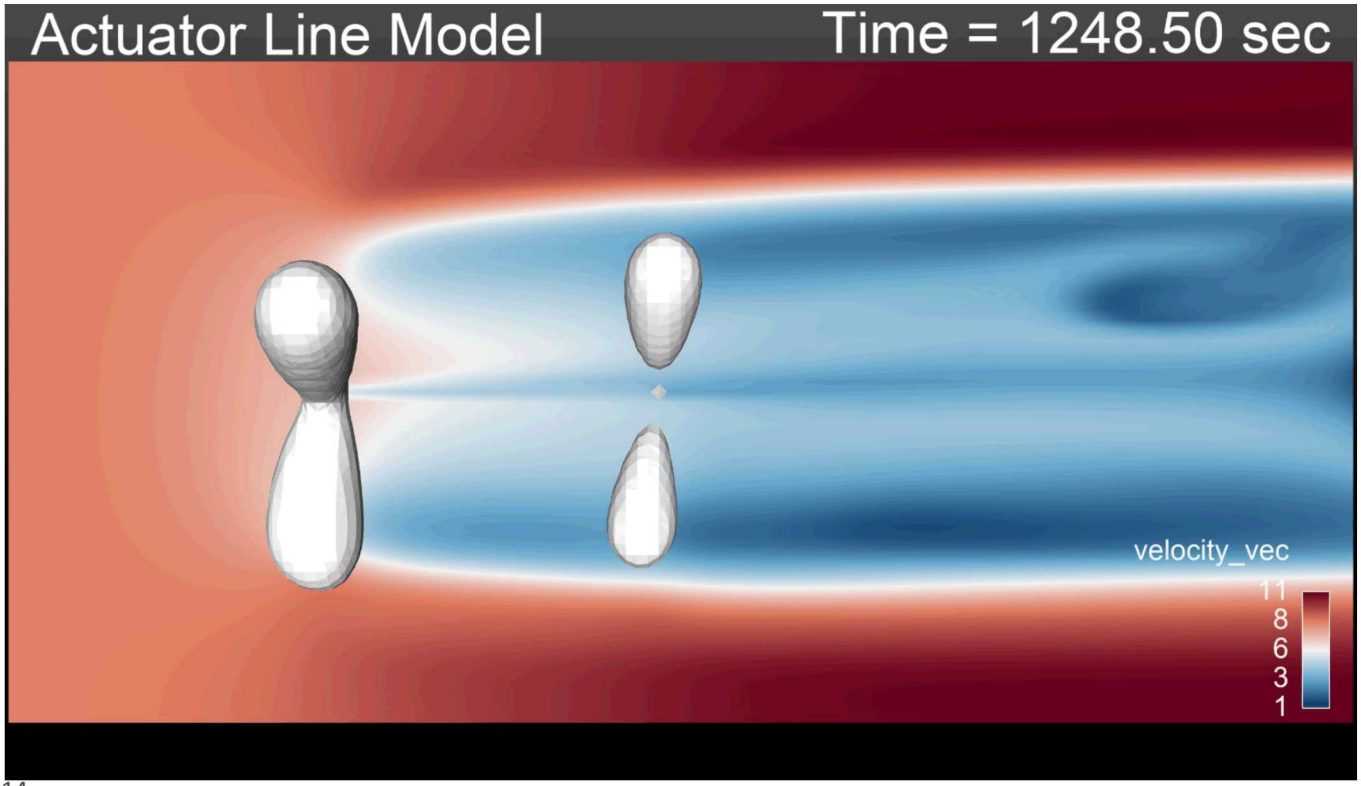




- The new ADM is ~20% faster for a single turbine case running on 4 MPI ranks
- Similar results are also seen for the ALM
- More detail on timing in the V&V cases

Old vs New Model: Multi-turbine

- New Model can run multi-turbines with both ALM and ADM
- ~25% faster for two ALM turbines on 32 processors

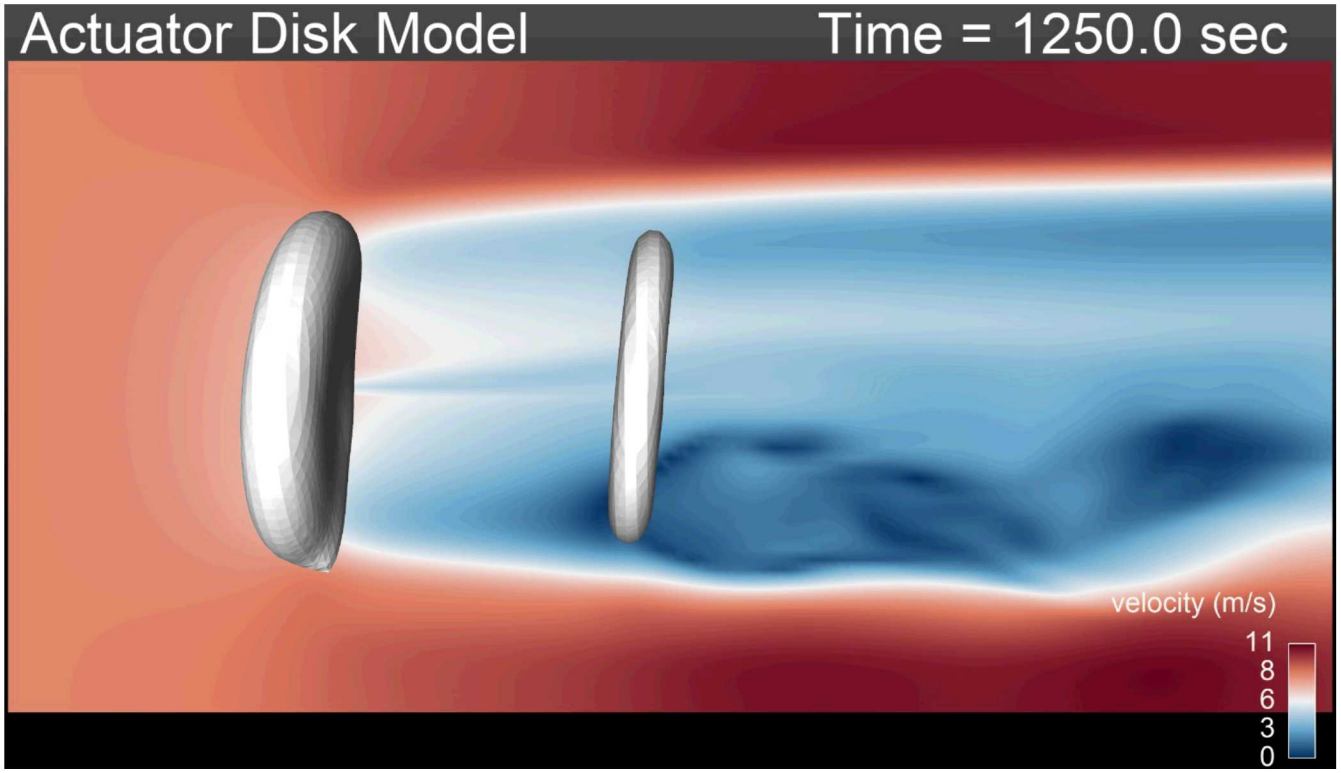


- Steady 8m/s inflow
- NREL 5MW Turbines
- White shows ALM

ATMOSPHERE TO ELECTRONS
U.S. DEPARTMENT OF ENERGY

Old vs New Model: Multi-turbine

- With the same number of total force points on the rotor, the ADM runs ~3 times faster than the ALM



- Steady 8m/s inflow
- NREL 5MW Turbines
- White shows ADM

Old vs New Model: Multi-turbine

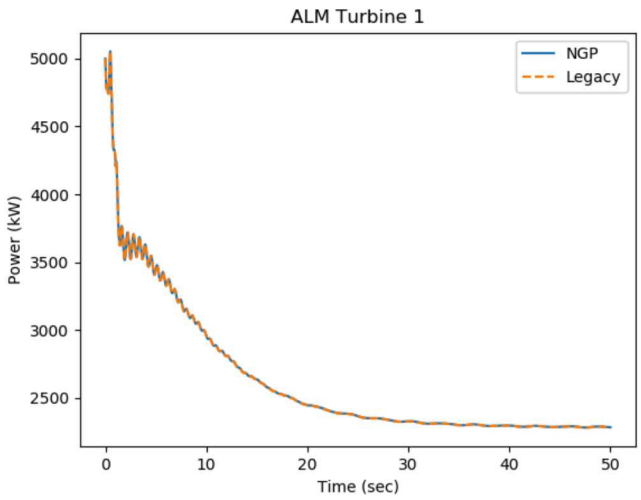
- Mesh has 524,288 elements ran on 32 processors
  - 16k elements/processor
- 50 sec simulation time

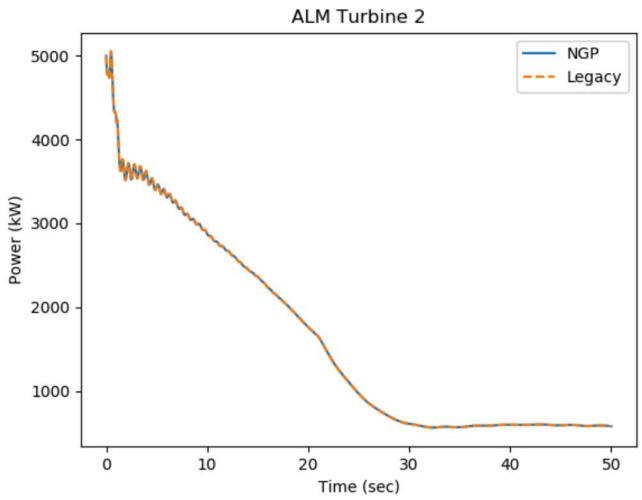
Model	Time Step (sec)	N (# radial actuator points)	Total # Actuator Points	# Time Steps	Walltime to Finish	Actuator Total Time* (sec)
ALM - legacy	0.125	57	171	400	27 min	1011
ALM –new NGP	0.125	57	171	400	19 min	507
ADM - legacy	0.5	8	172	100	6 min	201
ADM –new NGP	0.5	8	172	100	6 min	183

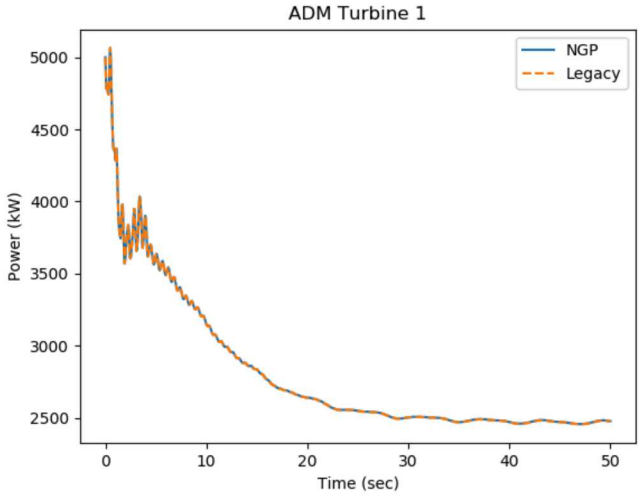
*Total time spent executing the actuator code averaged across all MPI ranks

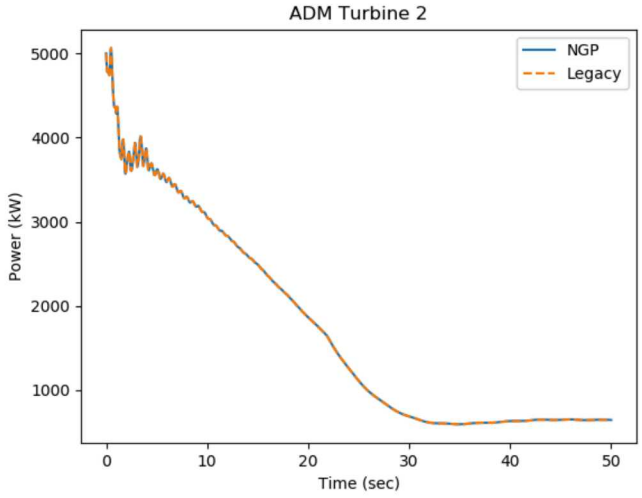
Old vs New Model: Multi-turbine

- Both ALM and ADM methods show same results for NGP and legacy model









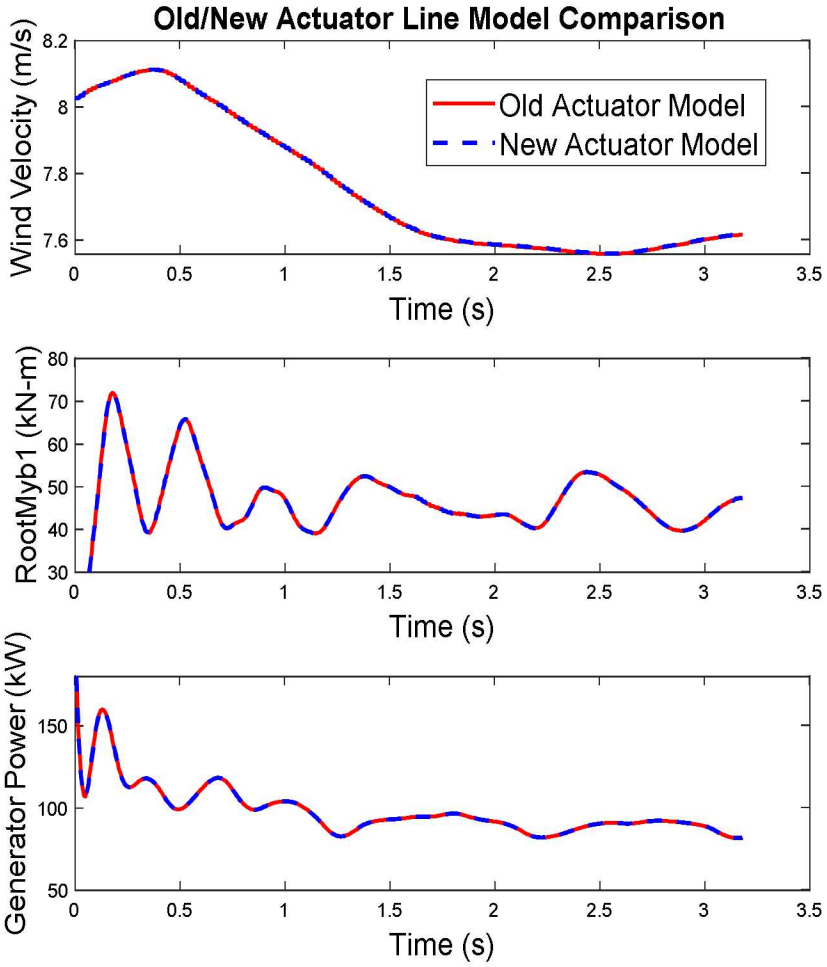
Old vs. New Model: Full ABL Simulations

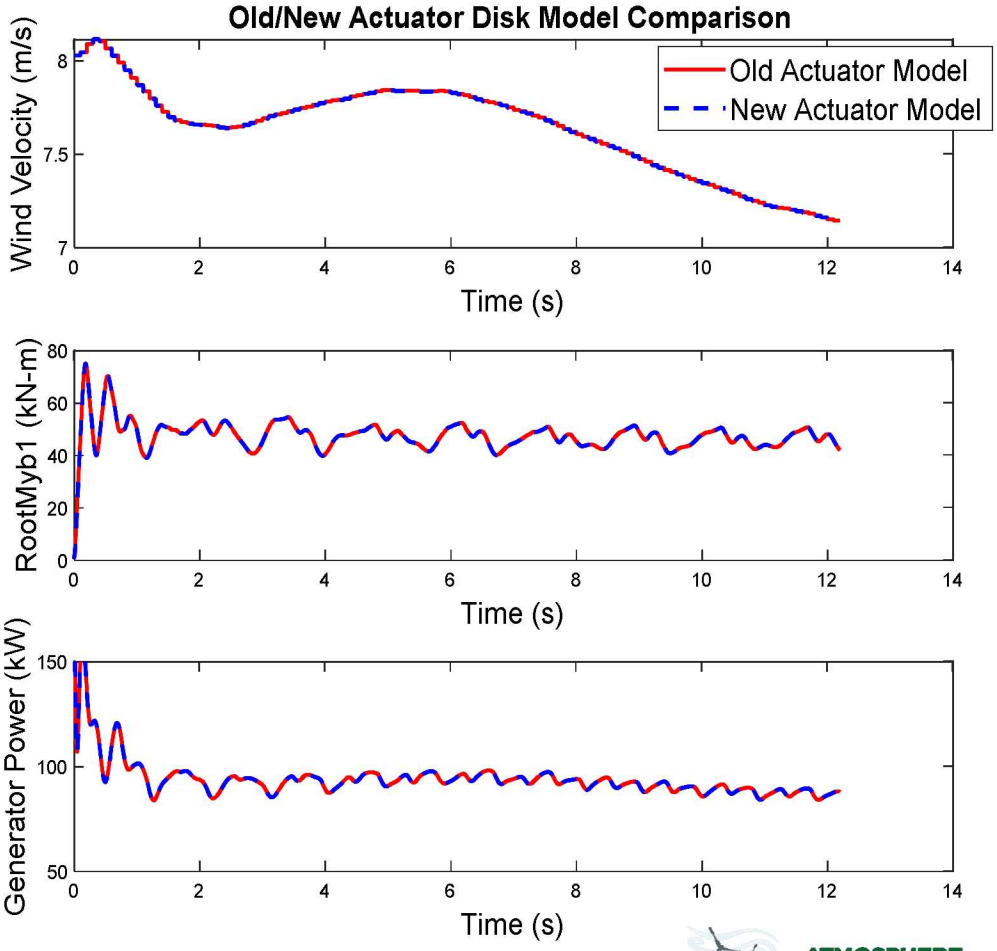
- Full-scale ABL simulation comparisons using old (legacy) and new (NGP) actuator models; everything else constant
- Neutral ABL, U_∞=8.69 m/s, 3x3x1 km domain, single V27 turbine over 10 sim. minutes
- Walltime per iteration are compared between legacy/NGP for both total Nalu-Wind time per timestep and the actuator specific code per timestep (Act. walltime)
- Significant timing performance improvements in both Nalu-Wind time step and actuator are observed for all mesh/model combinations; efficiency gains are most prominent at higher mesh resolutions

Turbine Model	# Mesh Elements	Time Step (sec)	Legacy (Nalu walltime secs / iter)	NGP (Nalu walltime secs / iter)	% Change	Legacy (Act. walltime secs / iter)	NGP (Act. walltime secs / iter)	% Change
Actuator Disk	10.5e6	0.2	7.1	5.1	-28.2%	5.42	1.63	-69.9%
Actuator Disk	11.8e6	0.2	27.2	11.4	-58.1%	24.20	1.79	-92.6%
Actuator Disk	22.5e6	0.1	176.0	43.2	-75.5%	170.4	2.29	-98.7%
Actuator Line	10.5e6	0.02	1.7	1.6	-5.9%	0.33	0.18	-45.5%
Actuator Line	11.8e6	0.02	3.0	2.6	-13.3%	0.99	0.22	-77.8%
Actuator Line	22.5e6	0.02	12.1	6.5	-46.3%	5.86	0.46	-92.2%

Old vs. New Model: Full ABL Simulations

- OpenFAST turbine outputs for both the new actuator disk and actuator line models are within 0.05% of the old model results

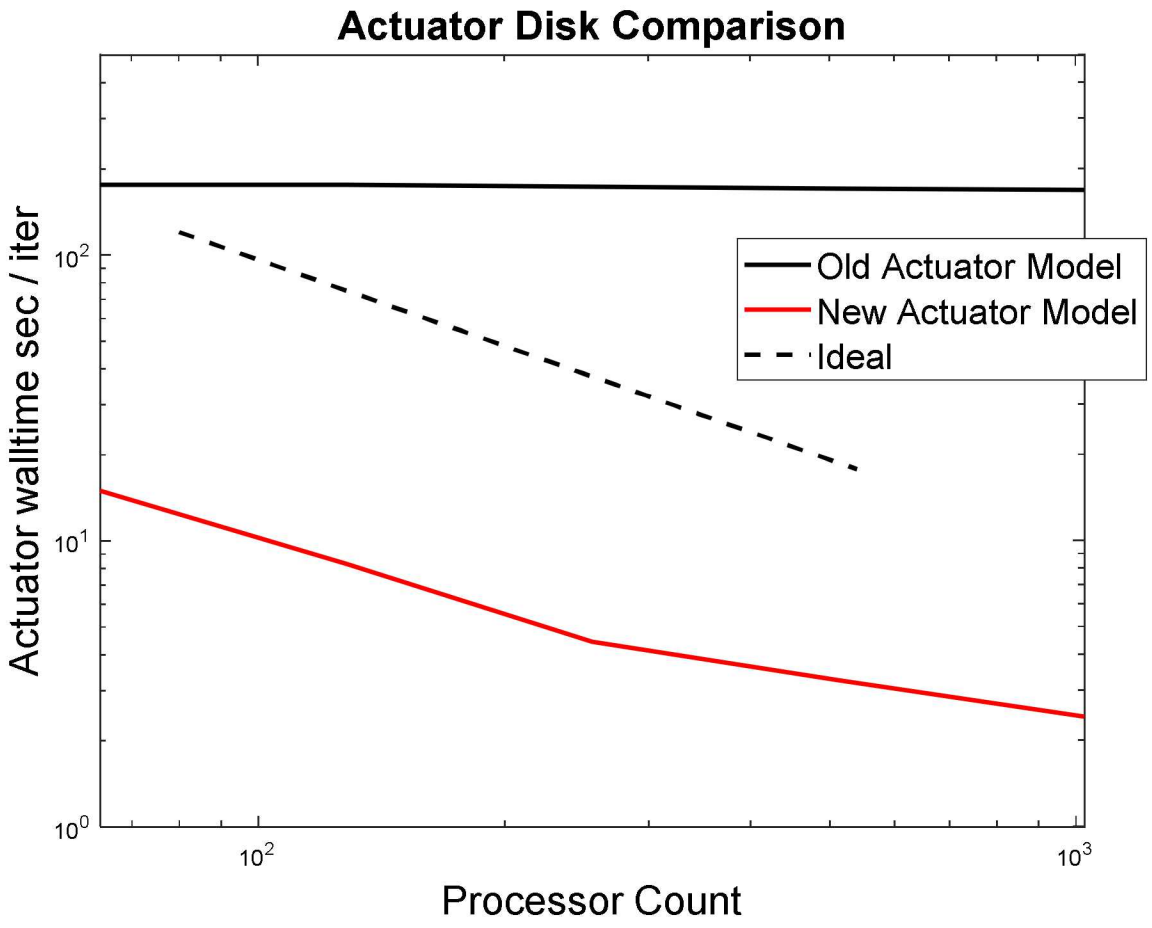




ATMOSPHERE TO ELECTRONS
U.S. DEPARTMENT OF ENERGY

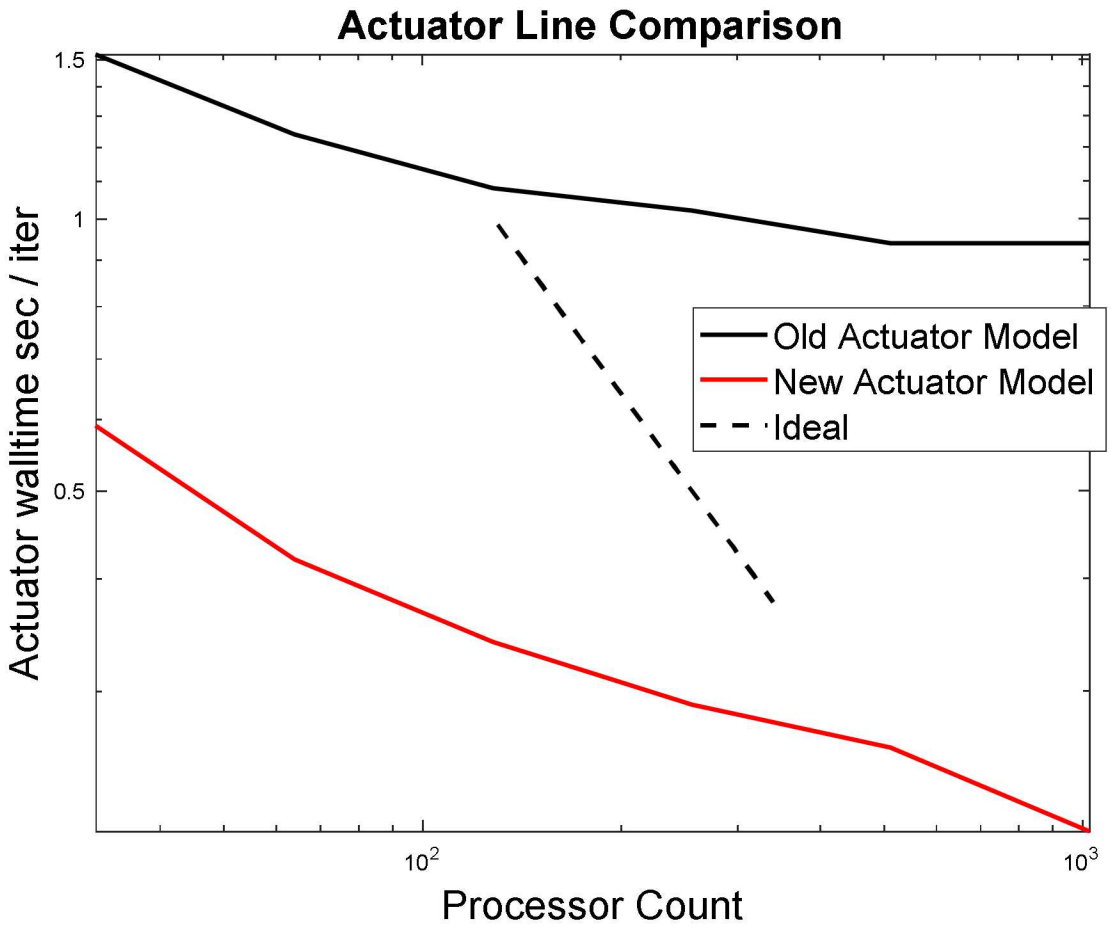
Old vs. New Model: Full ABL Simulations

- Strong scalability comparison of new/old actuator disk models
- 22.5e6 element mesh
- Actuator code execution time per iteration
- New NGP ADM shows significant improved timing and scaling compared to non-scalable legacy ADM



Old vs. New Model: Full ABL Simulations

- Strong scalability comparison of new/old actuator line models
- 11.8e6 element mesh
- Actuator code execution time per iteration
- New NGP ALM shows significant improved timing and scaling compared to legacy ALM



New AD Model: Full ABL Simulations

- Time step parametric study for new actuator disk model
- 11.8e6 element mesh
- Similar to the old model, the new actuator disk model can increase the time step from AL simulations by a factor of 20 (0.02 s -> 0.4 s)
- Similar to the old model, the new actuator disk model encounters numerical instability at dt ≥ 0.5 s for this case

Time Step (sec)	# iterations	Simulation Time (sec)	Walltime (sec)	Time/Tstep (sec / iter)	CFL	Mean GenPwr (kW)	% Change
0.05 s	1644	82	21600	13.1	0.5	87.5	-
0.1 s	1552	155	21600	13.9	0.9	86.8	-0.7%
0.2 s	1476	295	21600	14.6	1.7	85.9	-1.7%
0.3 s	1373	412	21600	15.7	2.6	85.3	-2.4%
0.4 s	1042	417	21600	20.7	3.5	85.0	-2.8%
0.5 s	894	447	21600	24.2	306	79.7	-8.8%

New AD Model: Full ABL Simulations

- Total actuator point count parametric study
- 22.5e6 element mesh
- % change for walltime/iter is relative to the N=8 case; % change for mean GenPwr is relative to the N=50 case
- Significant speed-ups in simulation efficiency by reducing the total number of actuator force points; N = 8 achieves a factor of 8 reduction in simulation time compared to N = 50 with less than 1% change in solution accuracy

Turbine Model	Time Step (sec)	N (# of radial points)	Total # of Actuator Points	Time/Tstep (sec / iter)	% Change	Mean GenPwr (kW)	% Change
ADM	0.1	8	172	7.4	-	90.54	0.9%
ADM	0.1	12	409	9.6	+29.7%	90.08	0.4%
ADM	0.1	16	746	11.8	+59.5%	89.93	0.2%
ADM	0.1	20	1184	14.4	+94.6%	89.85	0.1%
ADM	0.1	50	7672	59.2	+700%	89.76	-

Conclusions and Next Steps

- New code is performing well, on average 20-25% speedup over old code with increasing returns as mesh is refined (75% documented for ADM in ABL)
- More robust scalability, memory scaling issues resolved
- Large improvement in testing: 30+ unit tests added
- Workflow improvements, mainly guardrails
- Initial V&V efforts completed, but more in progress
- Next Steps
  - Convert AALM to new code base
  - Deprecate and remove legacy code
  - Move as much execution to GPU as possible, add GPU testing
  - Face average (Rhie-Chow-esque) interpolation for ALM/ADM methods
  - Additional formal documentation and V&V tests

Acknowledgements

Funding provided by the U.S. Department of Energy Office of Energy Efficiency and Renewable Energy Wind Energy Technologies Office.

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 work was authored in part by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308.

NREL
NATIONAL RENEWABLE ENERGY LABORATORY

ATMOSPHERE TO ELECTRONS
U.S. DEPARTMENT OF ENERGY