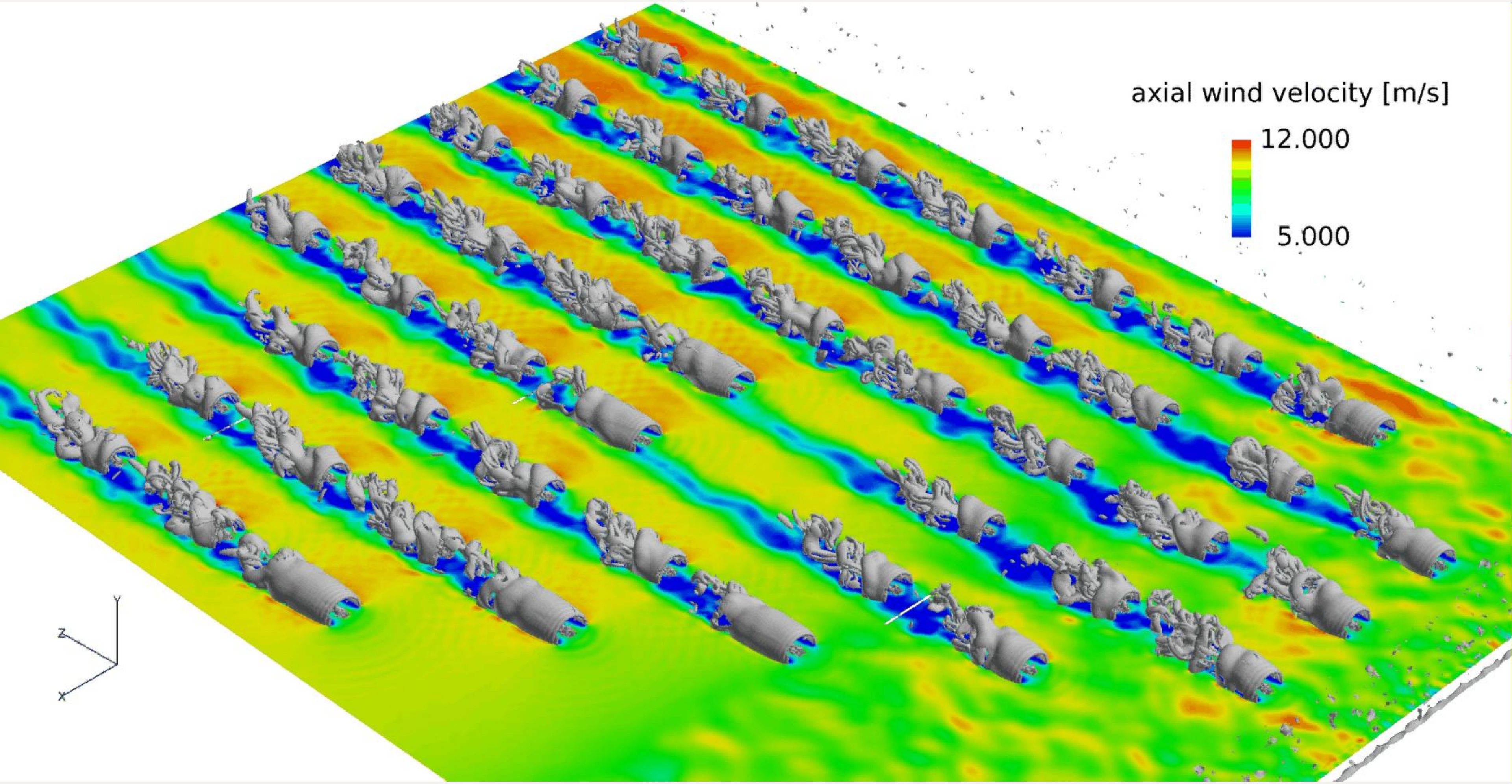
Advancing Complex Multidisciplinary Simulations in the Exascale Era

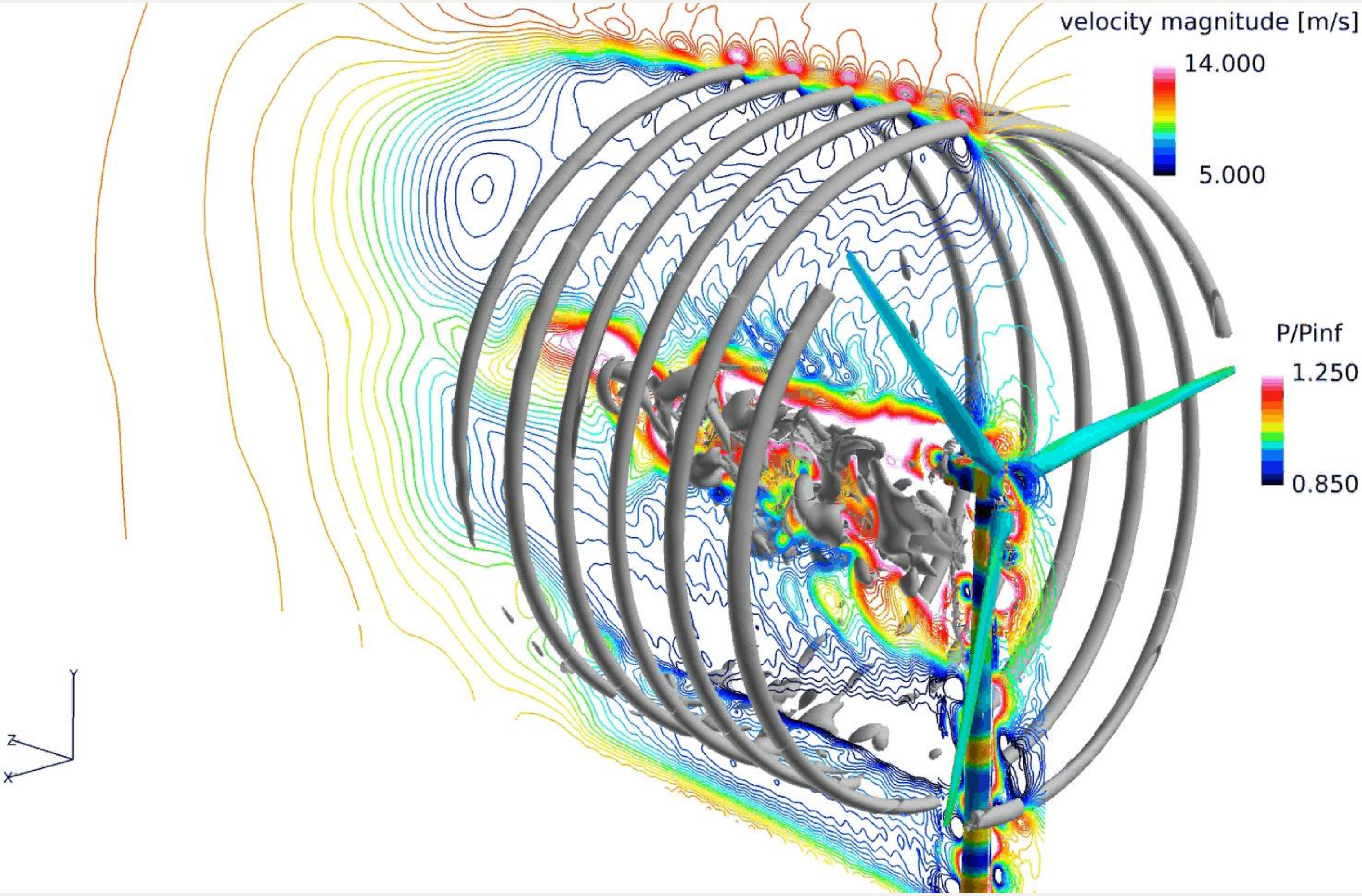
Dimitri Mavriplis
University of Wyoming

Motivation

- Realistic and useful simulations are becoming more complex and interdisciplinary
- Emerging HPC hardware trends:
  - FLOPS are free
  - Memory access more expensive
  - Massive parallelism
  - Favors simple algorithms (dense linear algebra)
- How to manage opposing trends ?
  - A narrative of our experience and a future vision

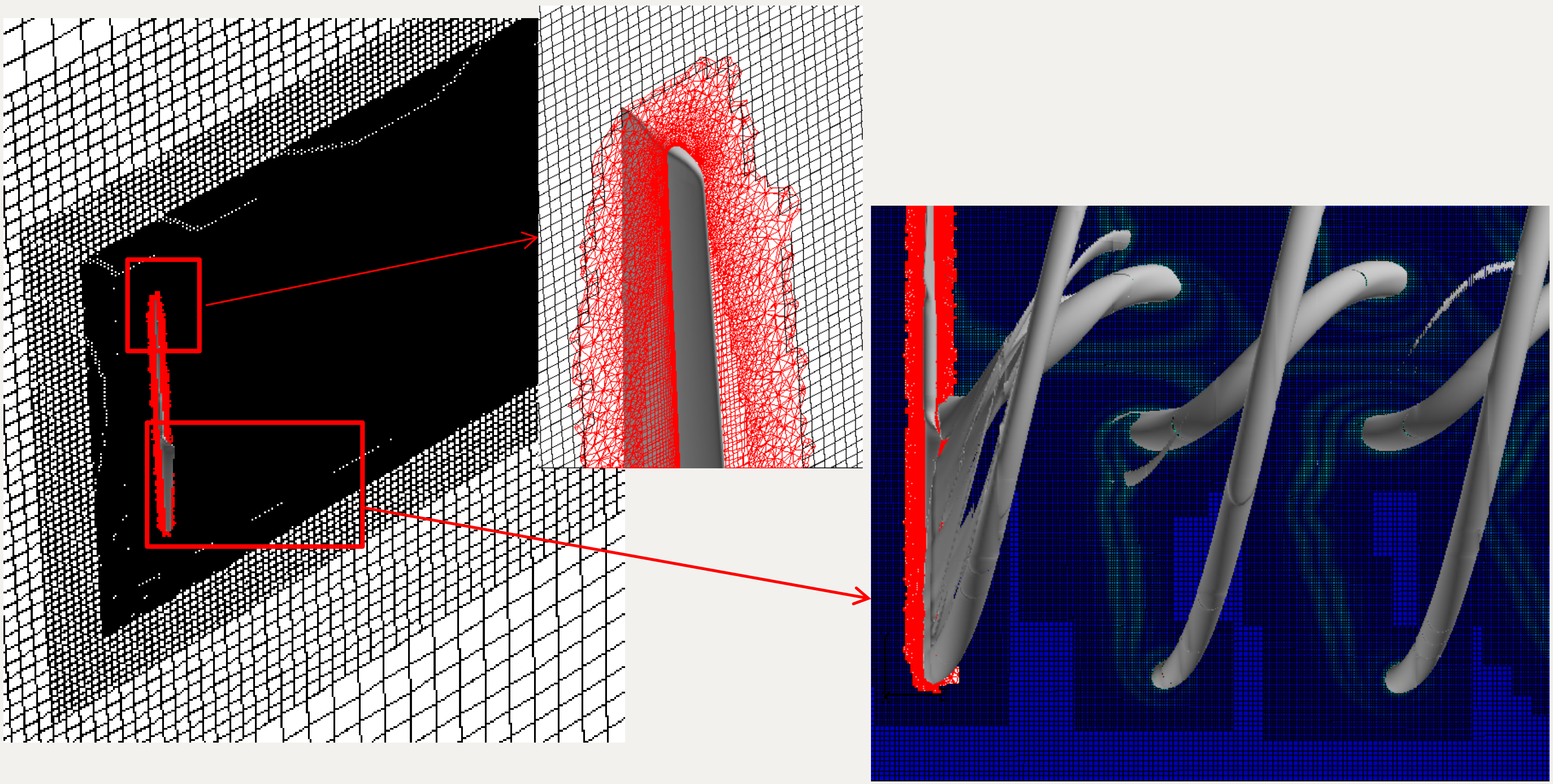
48 Turbine Wind Farm Simulation using HELIOS




velocity magnitude [m/s]
14.000
5.000
P/Pinf
1.250
0.850
Y
Z
X

HELIOS Wind Energy Simulation

- Wide range of scales (up to 10 orders of magnitude)
  - Blade boundary layer resolution (sublayer: microns)
  - Blade scale: meters
  - Wind farm scale: kilometers

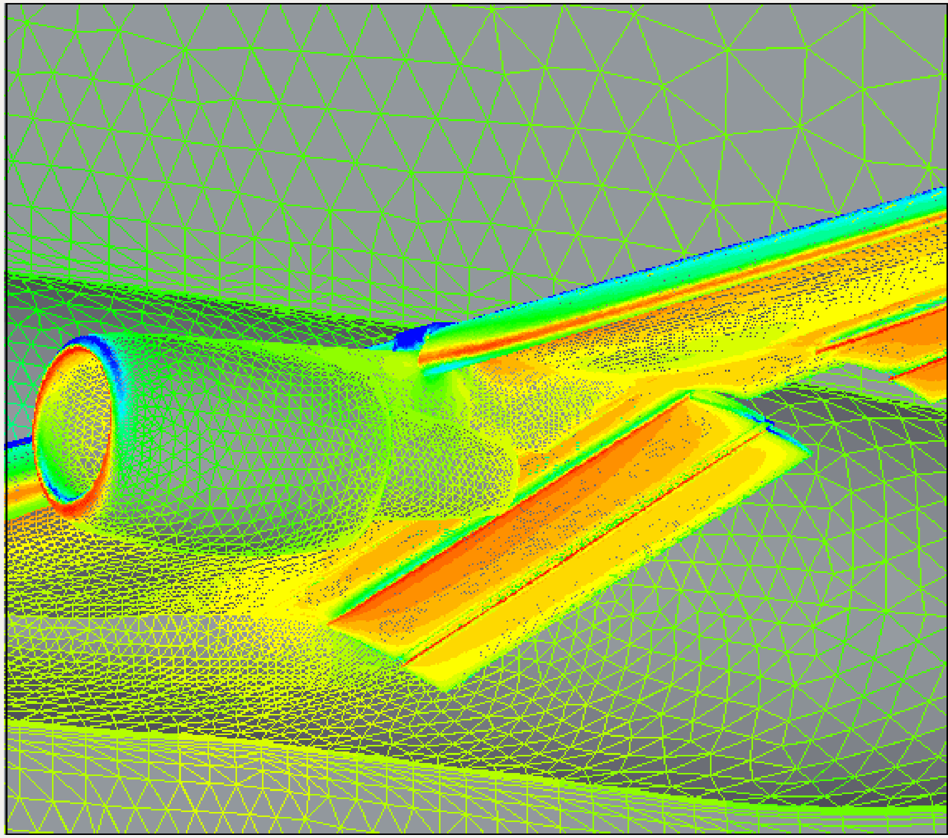


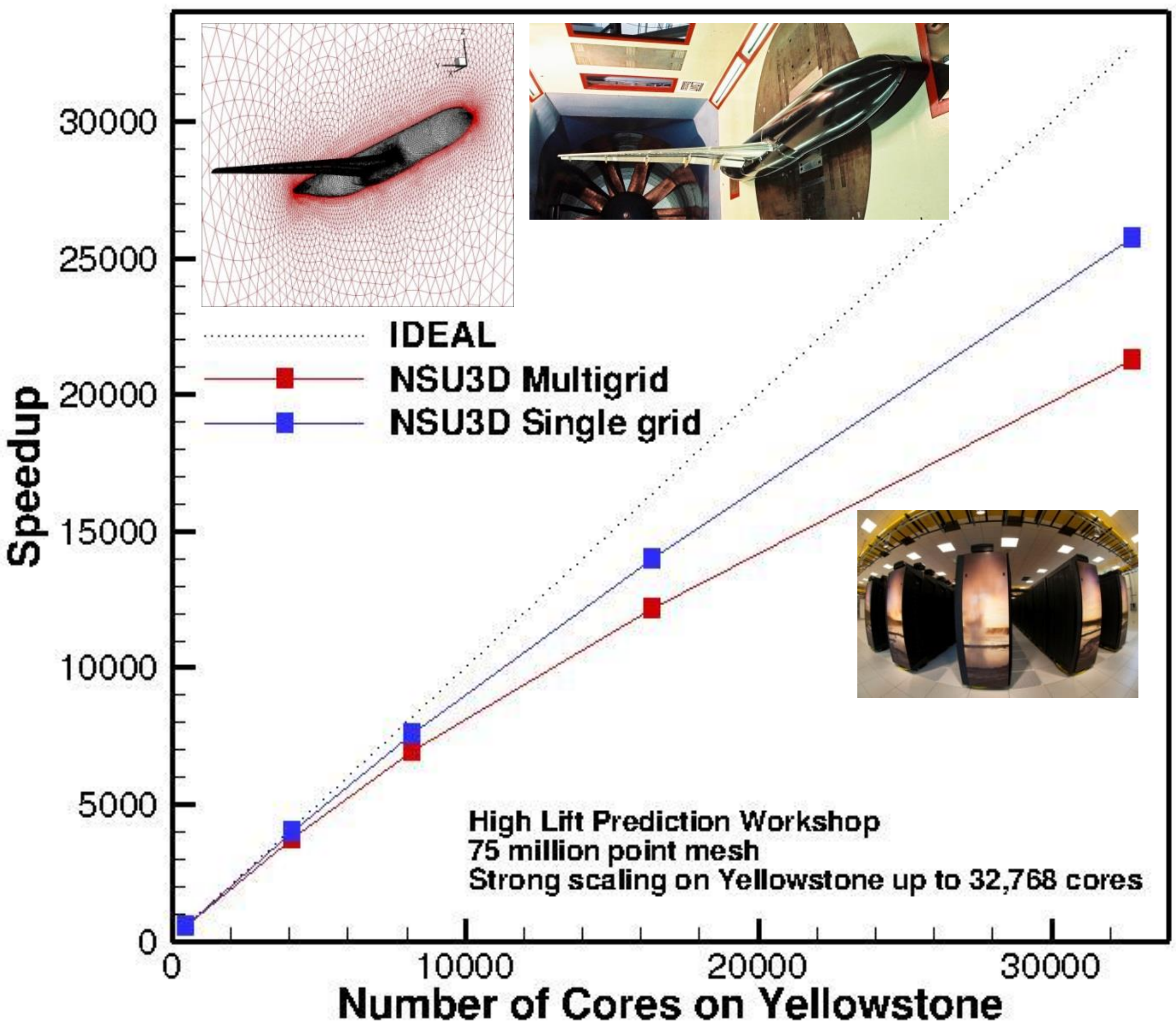
Ingredients

- Multidisciplinary
  - CFD
  - Atmospheric turbulence
  - Structural dynamics
  - Controls
  - Acoustics
- Multisolver
  - Near body unstructured
  - Off body structured/Cartesian
- Adaptive
- Overset

NSU3D: Unstructured Navier-Stokes Solver

- High fidelity viscous analysis
  - Resolves thin boundary layer to wall
    - $O(10^{-6})$ normal spacing
    - Stiff discrete equations to solve
    - Suite of turbulence models available
  - High accuracy objective: 1 drag count
- Unstructured mixed element grids for complex geometries
  - VGRID: NASA Langley
  - ICEM CFD, Others
- Production use in commercial, general aviation industry
- Extension to Design Optimization and Unsteady Simulations

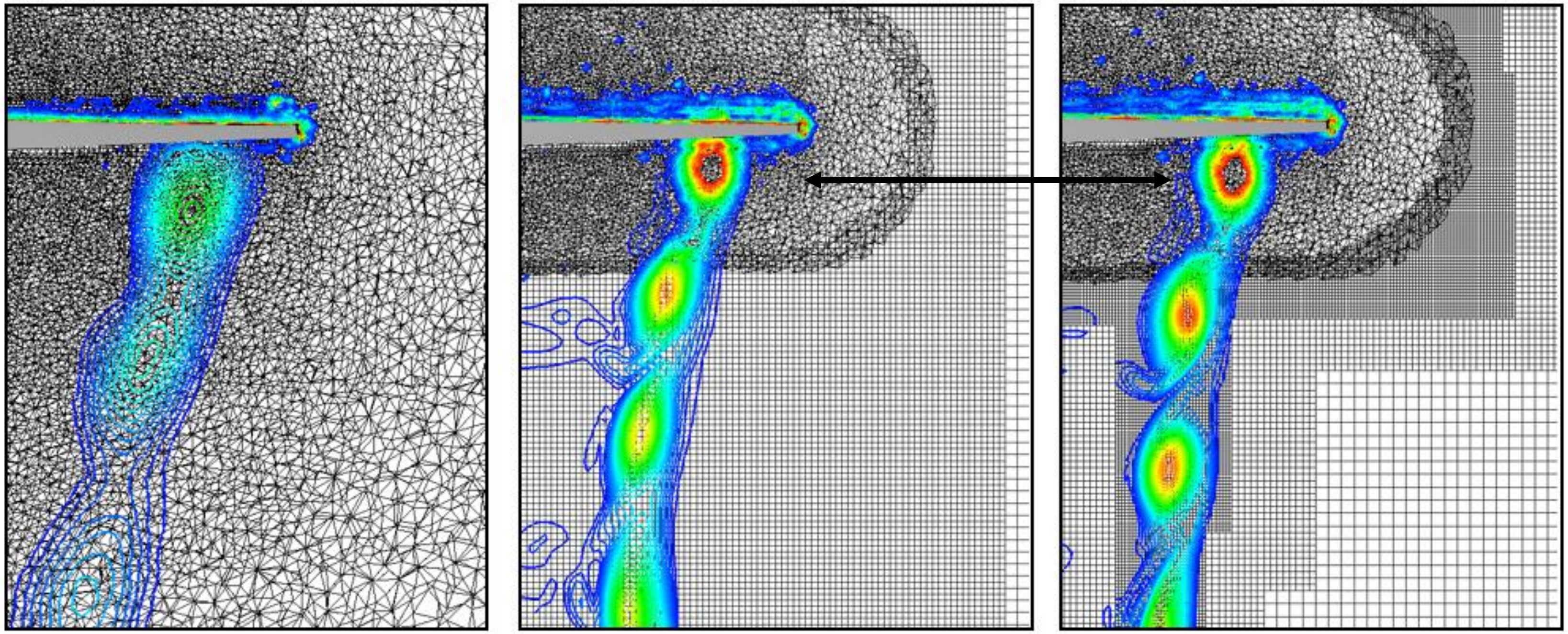




Strong scaling of AMG solver up to 32K cores

Adaptive Wake Resolution

14^o coll, M_{tip}=0.625



Unstructured | **Fixed Off-body** 7 levels | **Adaptive Off-body** 8 levels

- **Little difference in resolution of first vortex pass**
- **Downstream resolution maintained well by AMR**

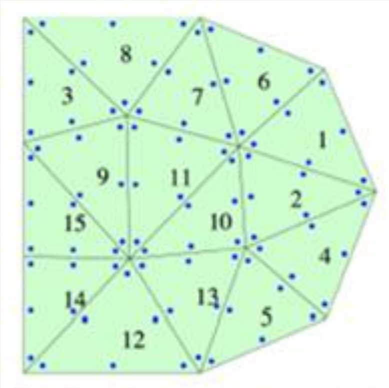
Medium NB mesh

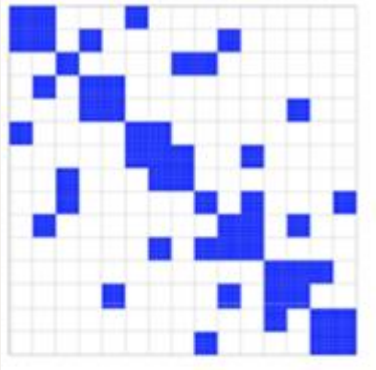
Strategy

- Computational intensive parts well suited for emerging hardware
  - Higher order methods
    - Traditional multicore CPUs
    - Many core CPUs (Intel PHI)
    - GPUs
- Complex but less expensive components on traditional hardware
  - Adaptive
  - Overset
- Consider additional relevant physics
- Modularity
  - Tight coupling
  - Correct level of modularity ?

High Order DG

- Nearest neighbor stencil
- Dense block matrices
- High computational rates
- Well suited for
  - AMR (simple stencil)
  - Overset (simple stencil)
  - HPC (computationally intensive)





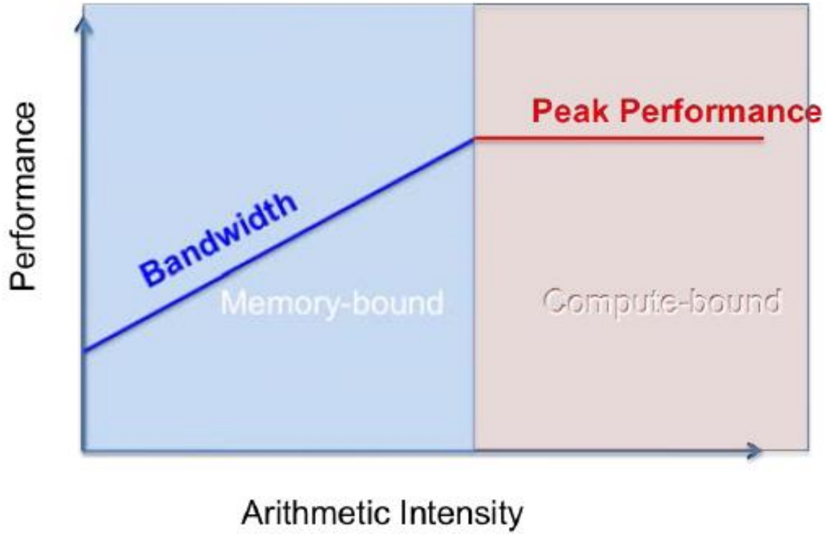
Computational Efficiency

Approximate power cost in picoJoules

	2011
DP FMADD flop	100 pJ
DP DRAM read	4800 pJ
Local Interconnect	7500 pJ
Cross System	9000 pJ

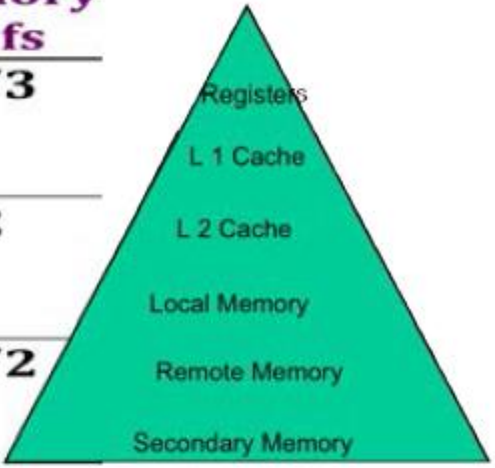
Source: John Shalf, LBNL

Roofline model

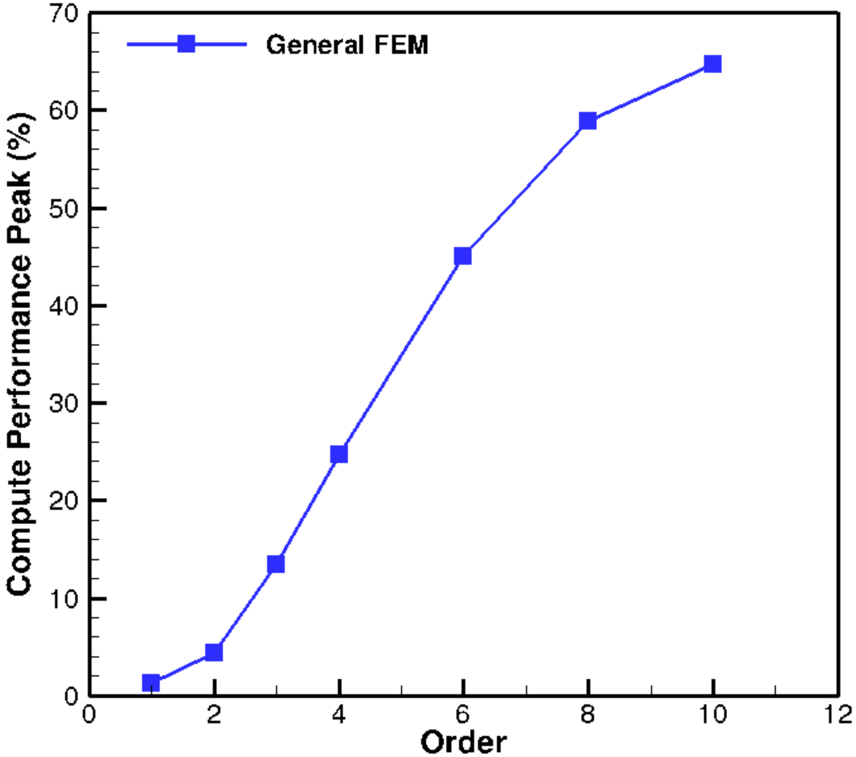


Computational Efficiency

BLAS	Memory Refs	Flops	Flops/ Memory Refs
Level 1 $y=y+\alpha x$	3n	2n	2/3
Level 2 $y=y+Ax$	n^2	$2n^2$	2
Level 3 $C=C+AB$	$4n^2$	$2n^3$	n/2

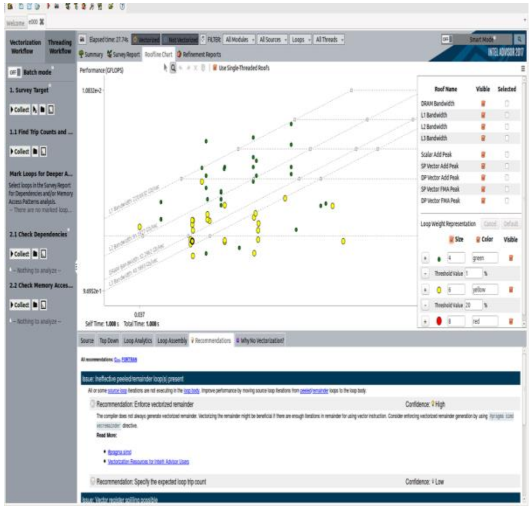


Computational Efficiency Discontinuous Galerkin Code



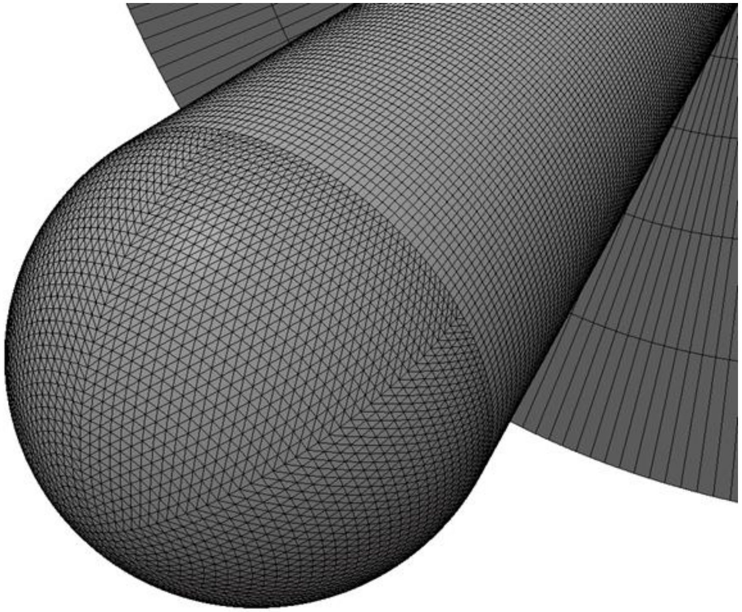
- Intel i7-560X
- 8 cores (1 socket)
- Theoretical Peak: 384 Gflops
  - (3GHz x 8 cores x 16 flops/clock)
- AVX-2 instruction set
- TAU Benchmark: 5.25 secs

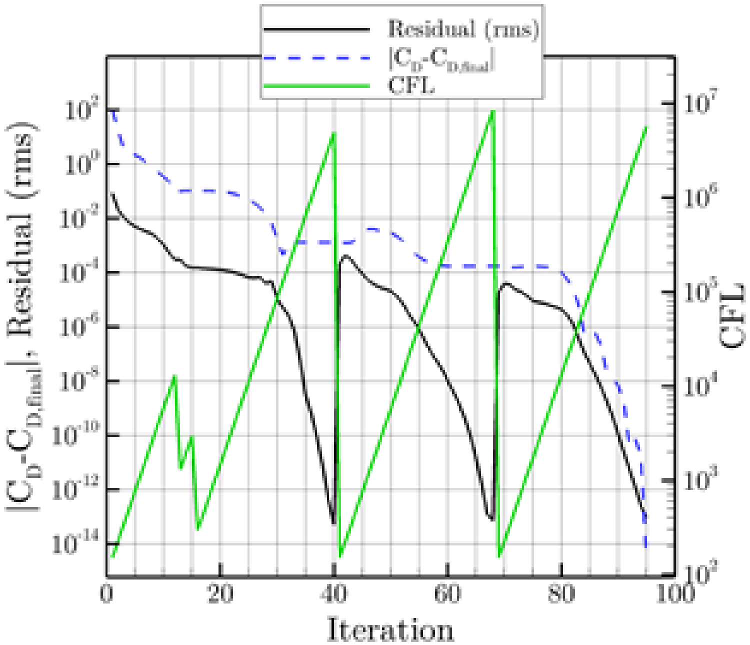
- **Level 3 BLAS implementation (Intel MKL)**



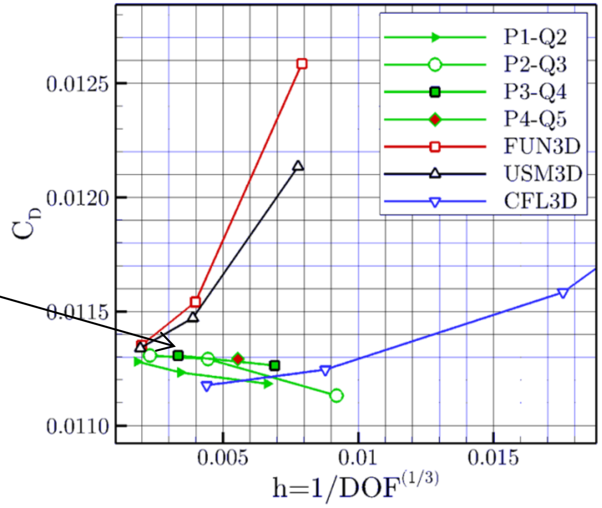
- Computational rates increase with p
  - 4.6 Gflops per socket at p=1
  - 250 Gflops per socket at p=10

Hemi-sphere Case (DG p=3)

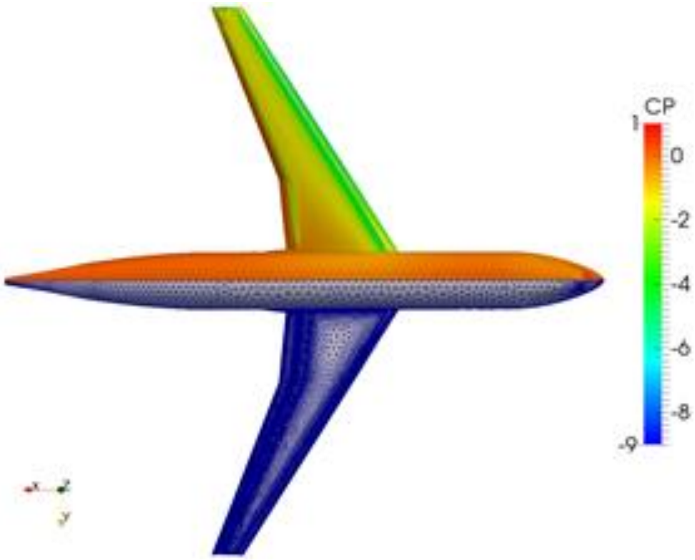


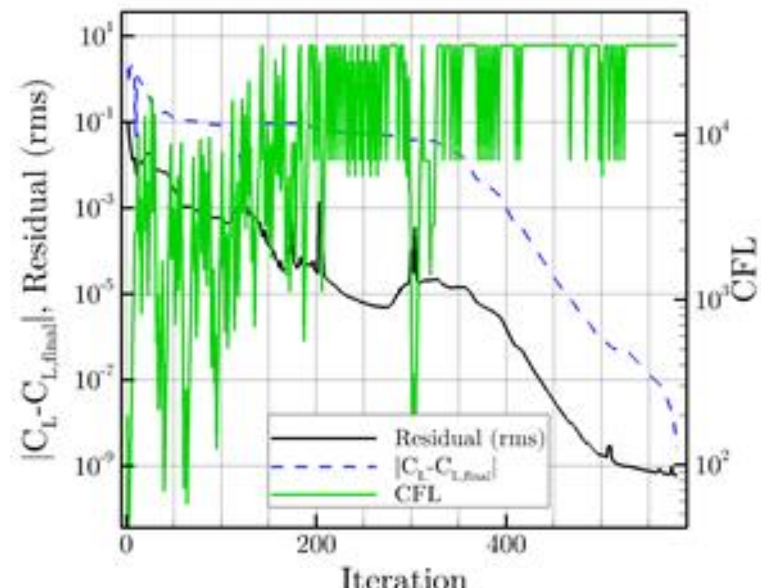


- NASA TMR Web Site Test Case
- Mesh curved to p+1 order
- p-continuation effective for non-linear convergence
- **p=3 most efficient for delivered accuracy**
- Easy test case
  - Nonlinear convergence p independent
  - Nonlinear convergence in < 50 iterations



HLPW2 (DLR-F11) Case (DG p=1)

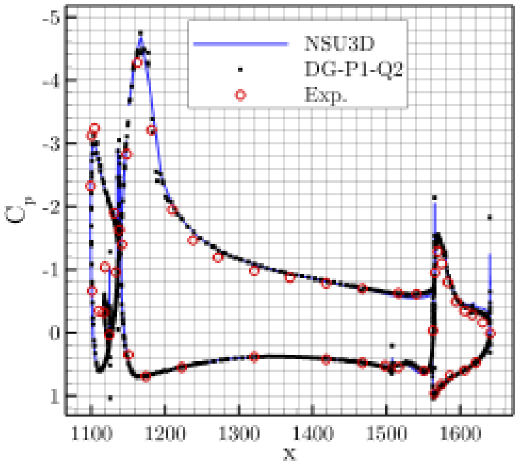




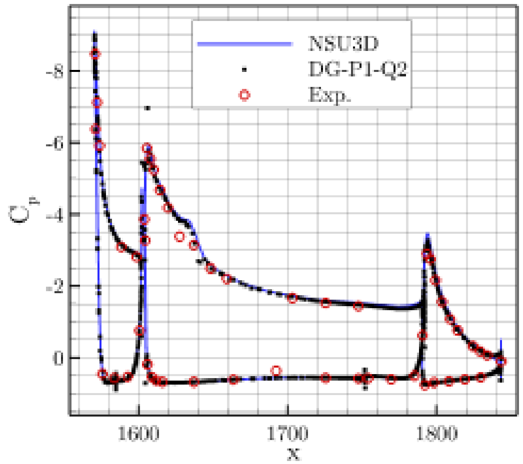
Incidence=7°

- Mesh curved to p+1 order (p=2 : quadratic)
- Mixed element mesh ~10M cells (similar to medium HLPW2 grid)
- ILU(0) preconditioner used, lines also demonstrated
- Frequent CFL limiting occurs
- Nonlinear solution requires over 500 steps

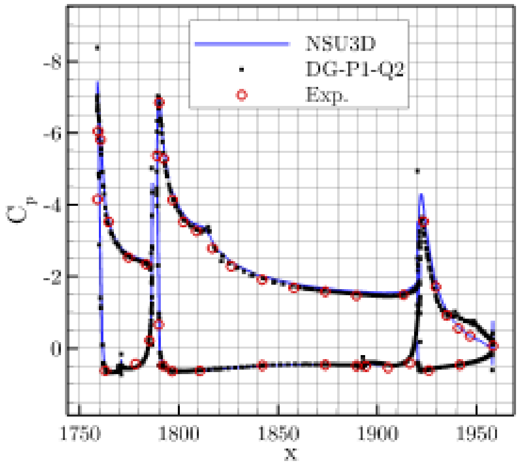
HLPW2 (DLR-F11) Case (DG p=1)



Station 1

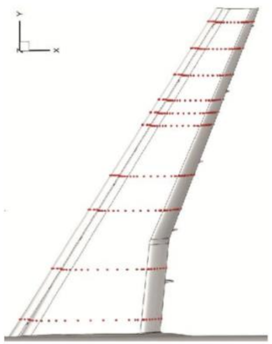


Station 6

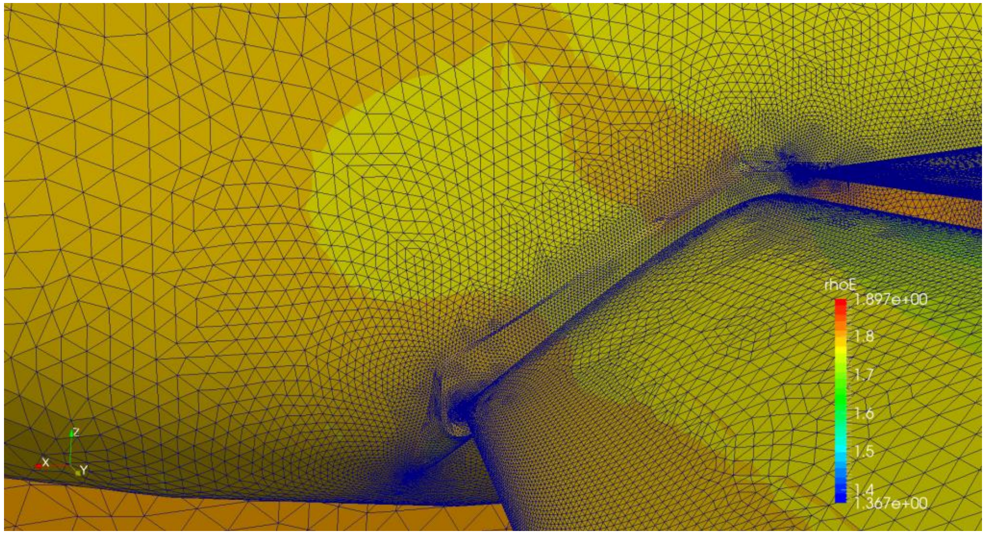


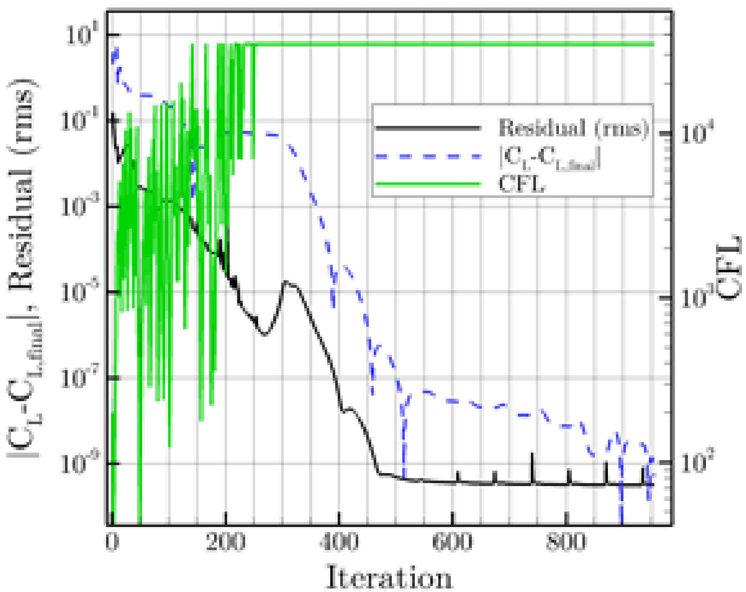
Station 10

- Mesh curved to p+1 order (p=2 : quadratic)
- Mixed element mesh ~10M cells (similar to medium H
- ILU(0) preconditioner used, lines also demonstrated
- Frequent CFL limiting occurs
- Nonlinear solution requires over 500 steps



HLPW2 (DLR-F11) Case (DG p=1)





Incidence=16°

- 6 hours on 8192 processors
  - Compare to FV (NSU3D): 1.5 hours on 1024 processors
- p=2 solutions also attempted
  - Solved to 10^{-6}, but very slow below this level

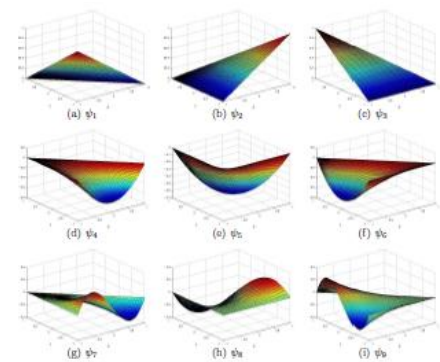
Alternative Point of View

- Near-body not best suited for application of high order methods
  - Benefits largest at very high order
  - Geometric singularities
  - Flow discontinuities (shocks)
  - Curved meshes required
  - Near body solver scales (almost) indefinitely through replication
    - Many instances of individual turbines
- Off-Body solver
  - Spans entire domain
  - High accuracy for convected flow features (wakes, vortices)
  - Large Eddy Simulation of turbulence
- Efficient implementation for very high order DG possible
  - Tensor product formulation
  - Explicit time-stepping

Tensor Product DG

- Abandon flexibility of modal bases for arbitrary element types

$$\psi(\xi,\eta,\zeta) = a + b\xi + c\eta + d\zeta + e\xi^2 + f\xi\eta + \ldots$$



- Tensor product bases:
  – Best suited for hexahedral elements

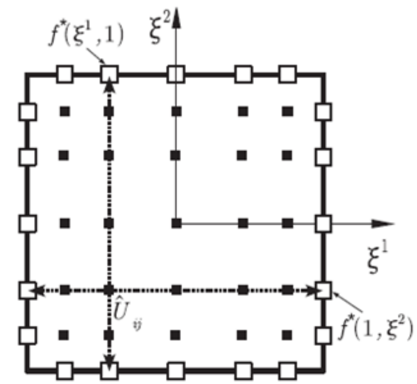
$$\psi_{ijk}(\xi,\eta,\zeta) = l_i(\xi)\, l_j(\eta)\, l_k(\zeta)$$

- l_i, l_j, l_k = 1-D Legendre polynomials:
  – values at quadrature points of integration become solution values
  – Removes requirement of reconstructing solution at quadrature points
  – All integrals reduce to dimension-by-dimension 1-D summations

Tensor Product DG

- Abandon flexibility of modal bases for arbitrary element types

$$\psi(\xi,\eta,\zeta) = a + b\xi + c\eta + d\zeta + e\xi^2 + f\xi\eta + \ldots$$



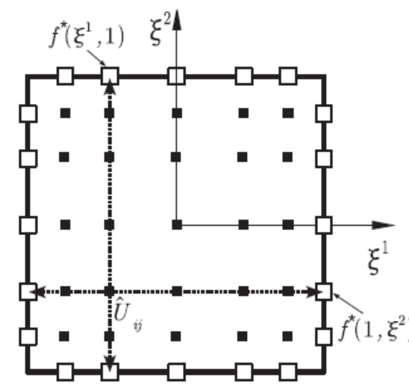
- Tensor product bases:
  - Best suited for hexahedral elements

$$\psi_{ijk}(\xi,\eta,\zeta) = l_i(\xi)l_j(\eta)l_k(\zeta)$$

- l_i, l_j, l_k = 1-D Legendre polynomials:
  - values at quadrature points of integration become solution values
  - Removes requirement of reconstructing solution at quadrature points
  - All integrals reduce to dimension-by-dimension 1-D summations

Tensor Product DG

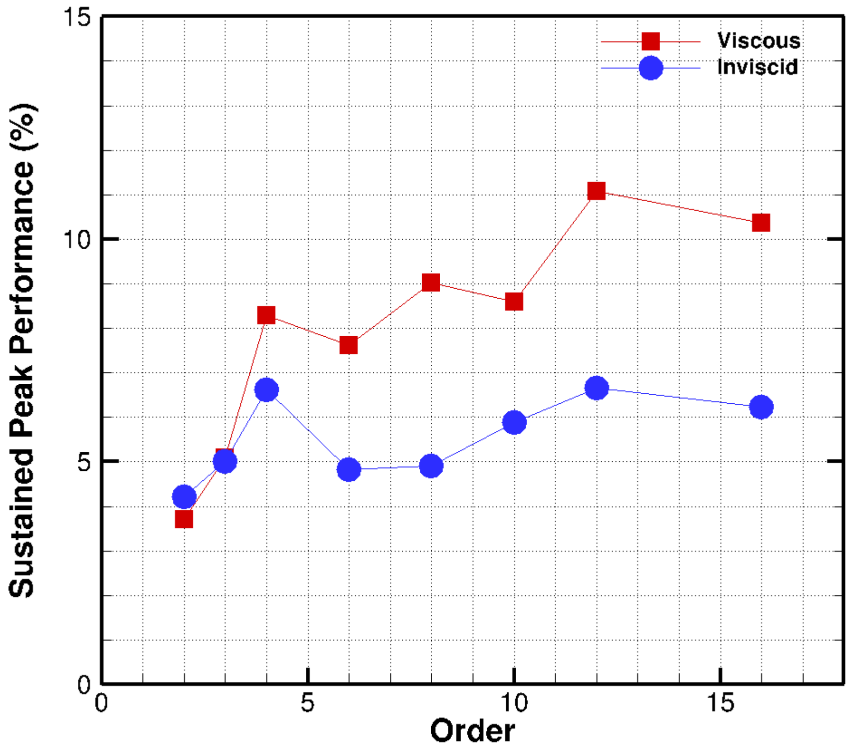
- Abandon flexibility of modal bases for arbitrary element types
  - Cost: $O(N^2)$ or $(p+1)^6$



- Tensor product bases:
  - Cost: $O(N^{4/3})$ or $(p+1)^4$

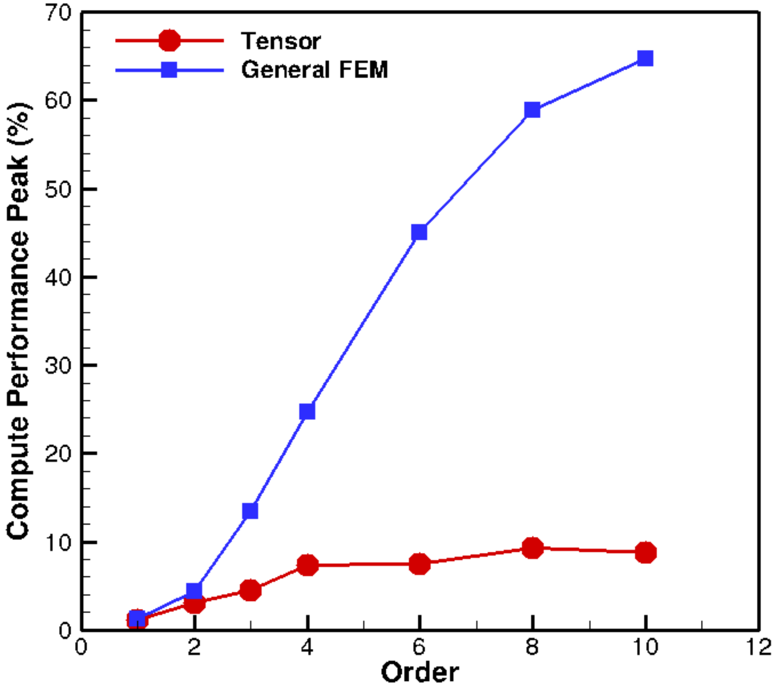
  - N = degrees of freedom
  - N = number of cells x dof per cell
  - dof per cell = $(p+1)^3$
  - p = order of accuracy
- Shown to be equivalent in cost to finite differences on cartesian mesh of same order (for residual evaluation)

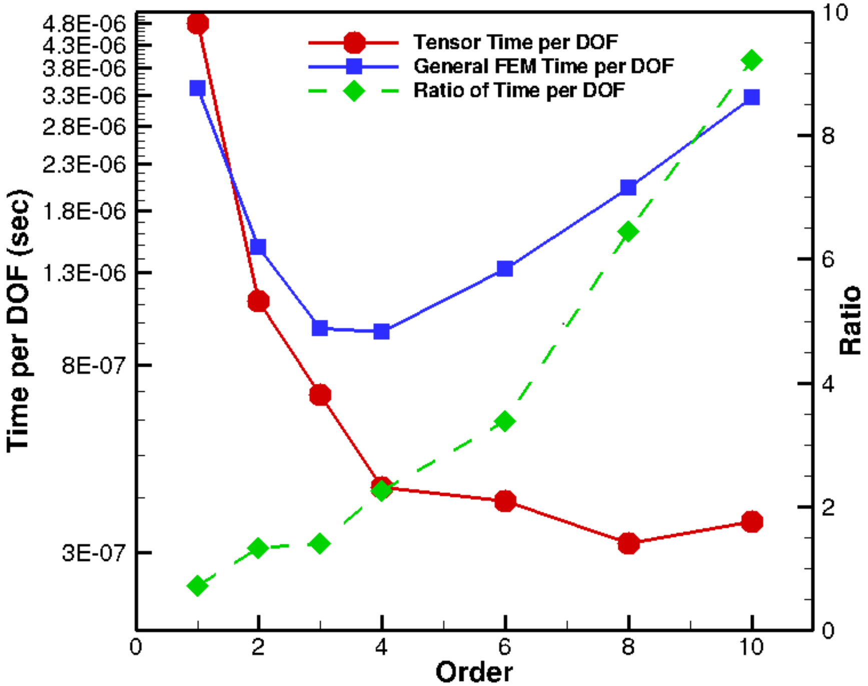
Tensor Product DG



- Higher order is more computationally intensive
  - 12% peak at p=12
- Less computationally intensive than general formulation

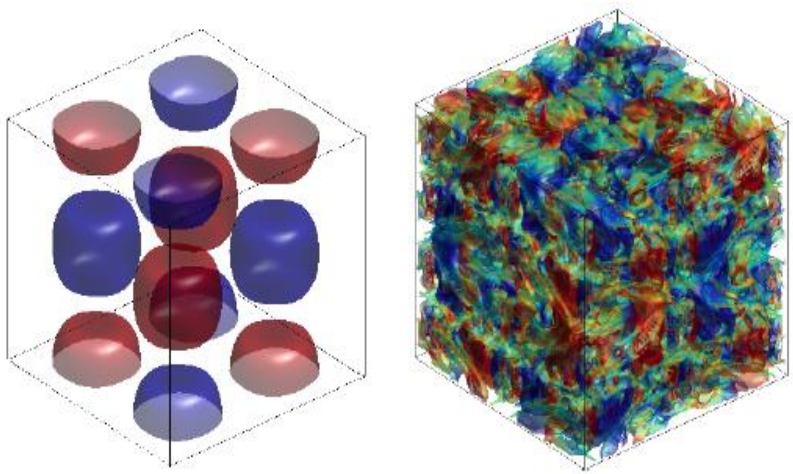
Tensor Product DG

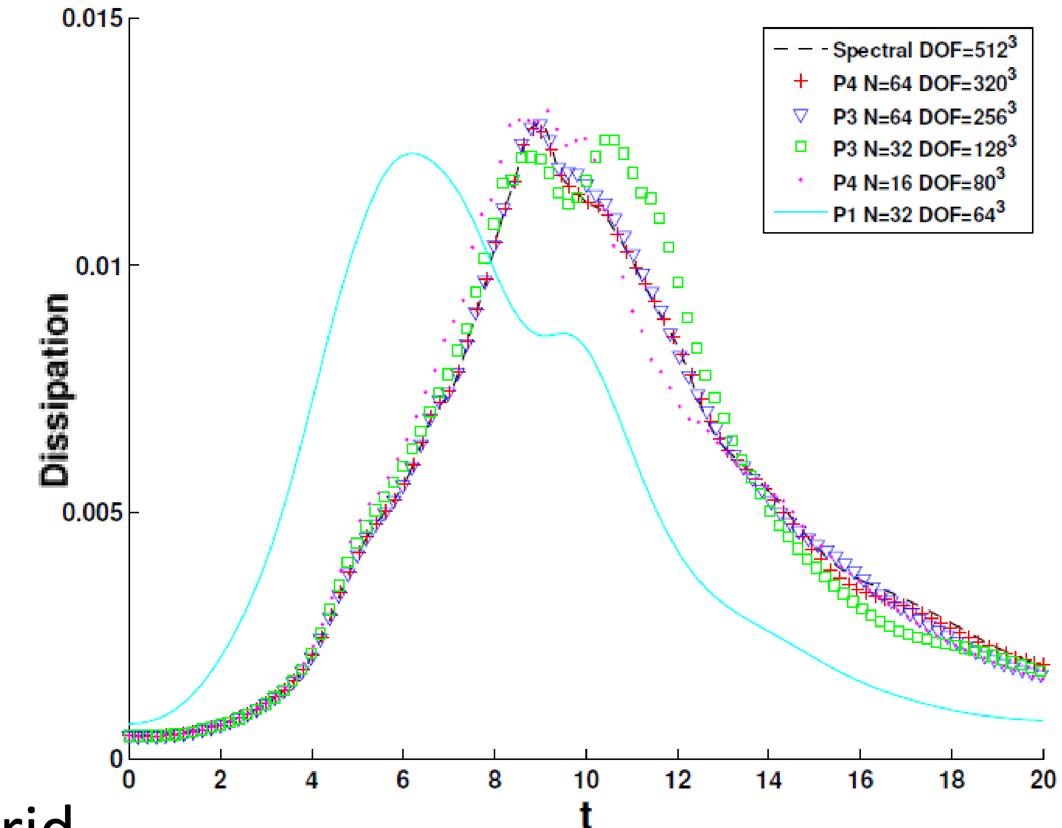




- Less computationally intensive than general formulation
- Overall cost much lower per degree of freedom
  - Cost per d.o.f decreases or flat with larger p
  - Faster than finite-difference

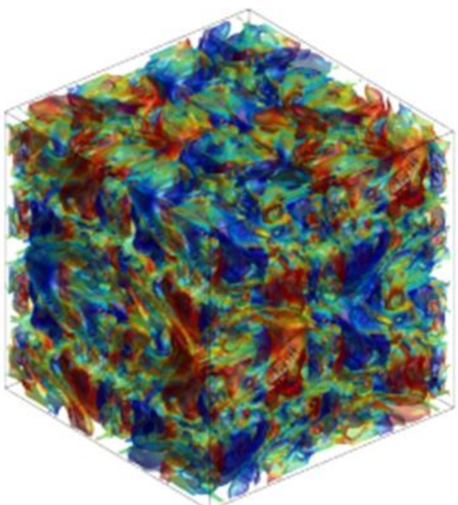
Taylor Green Vortex Validation

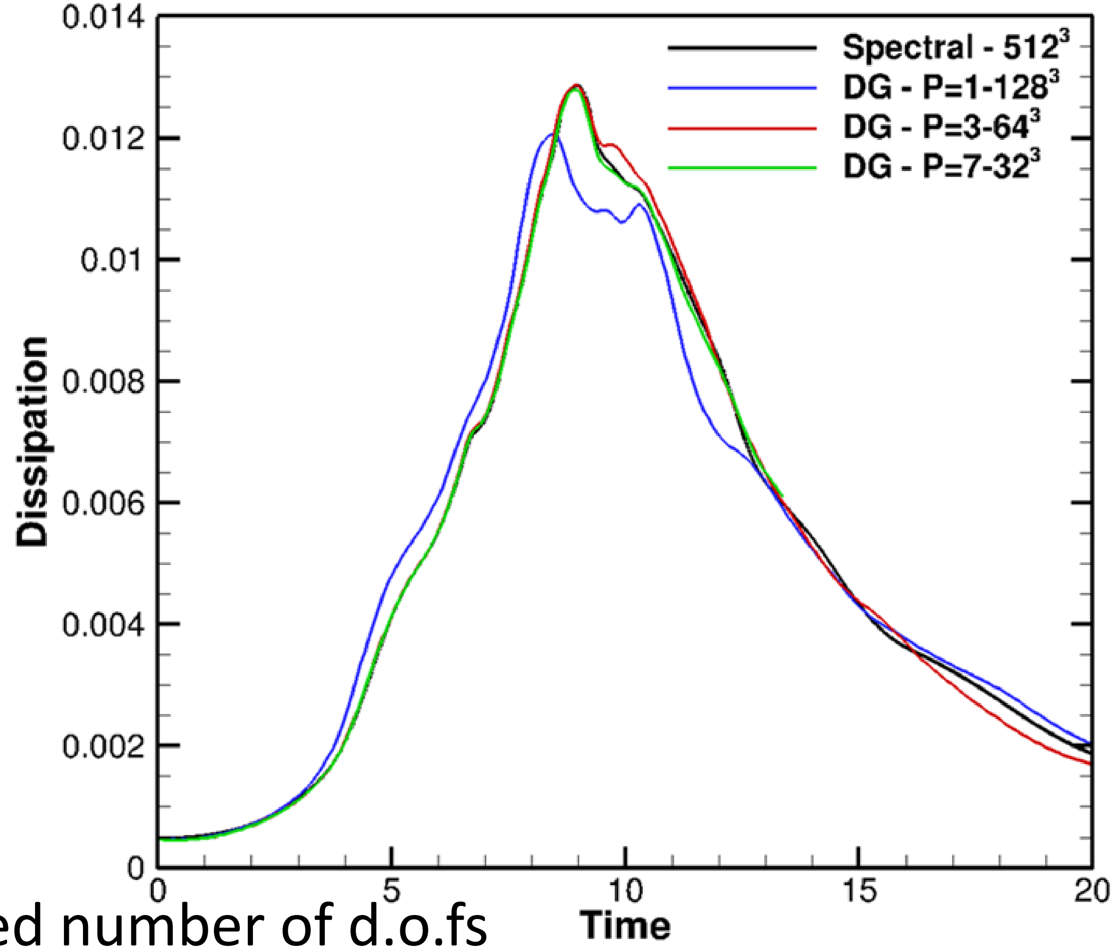




- Increasing p on “fixed” grid
  - Accuracy increases dramatically
    - Number of d.o.fs increases
    - Cost of simulation increases

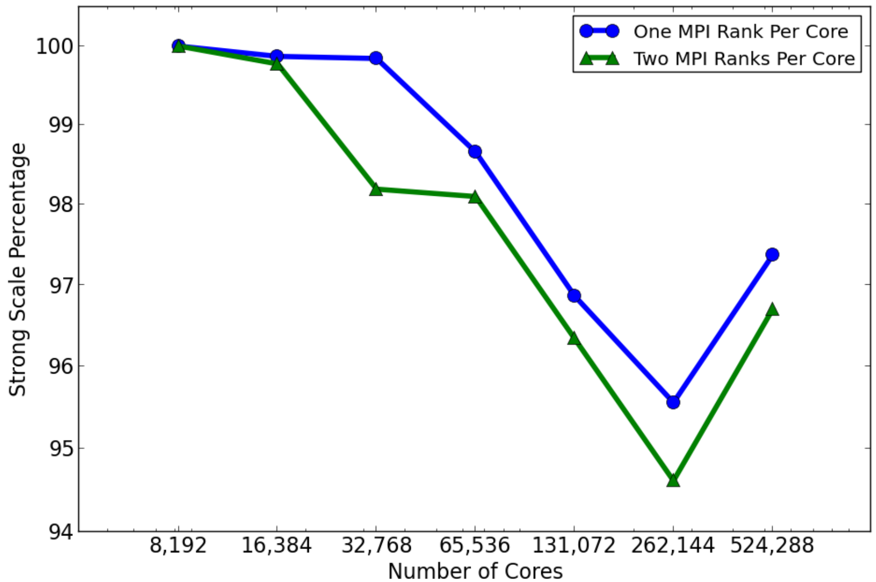
CartDG Solver Validation

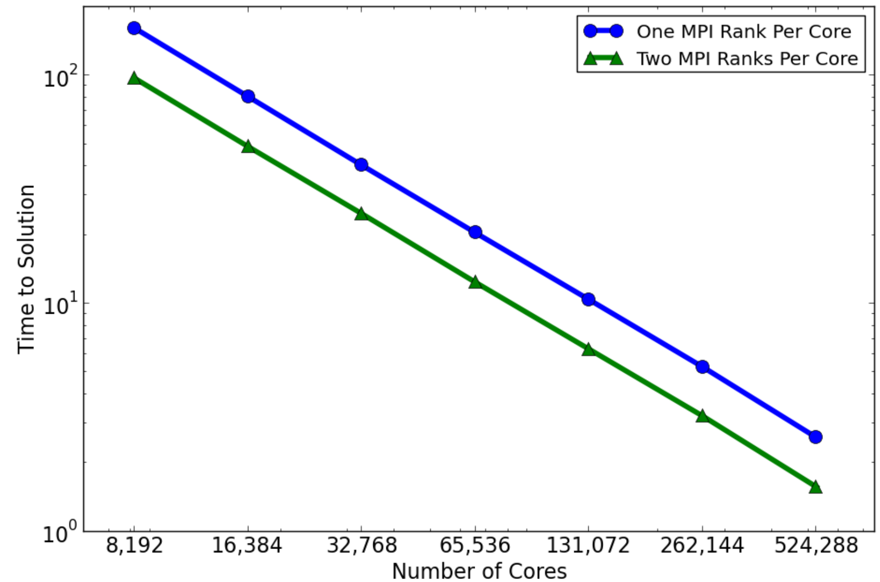




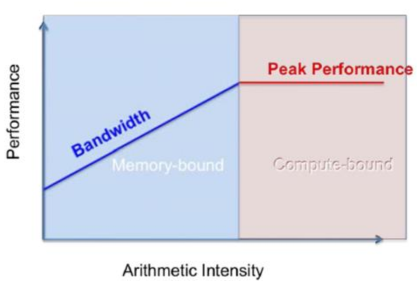
- Increasing p at fixed number of d.o.fs
  - Coarser meshes at higher p
  - Accuracy increases
  - Simulation cost decreases (per time step)

CartDG Scalability





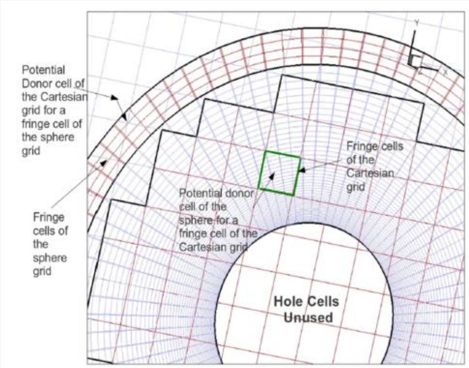
- Strong scaling on MIRA (@Argonne National Lab) for 84 billion degrees of freedom (512^3 mesh @p=4)using up to 1 million MPI ranks (2 per core)
  - Ideally suited for GPUs, Intel KNL (many core)



High-p Off-Body Solver

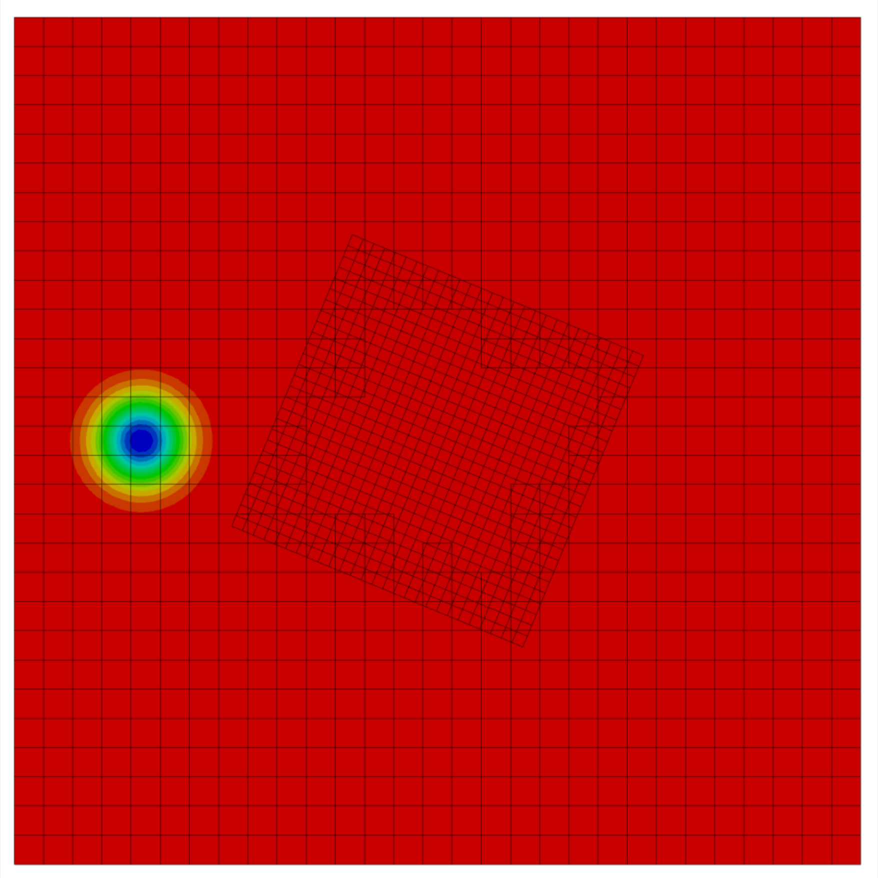
- Higher accuracy for same number of d.o.f
- Lower cost per degree of freedom
  - Explicit time step restriction
- Higher accuracy with fewer mesh cells
- Restricted to cartesian meshes
- Requires overset approach for complex geometries
  - Simplifies overset interpolation
    - Nearest neighbor stencil

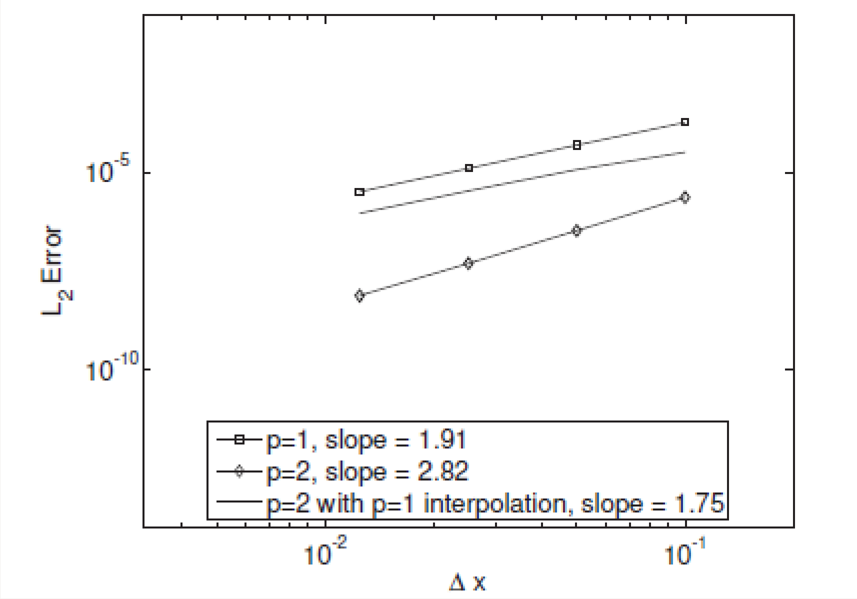
High-Order Overset Mesh Interpolation



- Considerations
  - ✓ High-order interpolation must be used to preserve design accuracy
  - ✓ Point inclusion algorithms must allow for curved mesh elements used in high-order discretizations
  - ✓ High-order DG discretizations contain multiple degrees of freedom within each element
- Basic criteria
  - ✓ Maintain design accuracy of individual mesh solvers
  - ✓ Maintain stability
  - ✓ Non-conservative (currently)

High Order Overset Interpolation

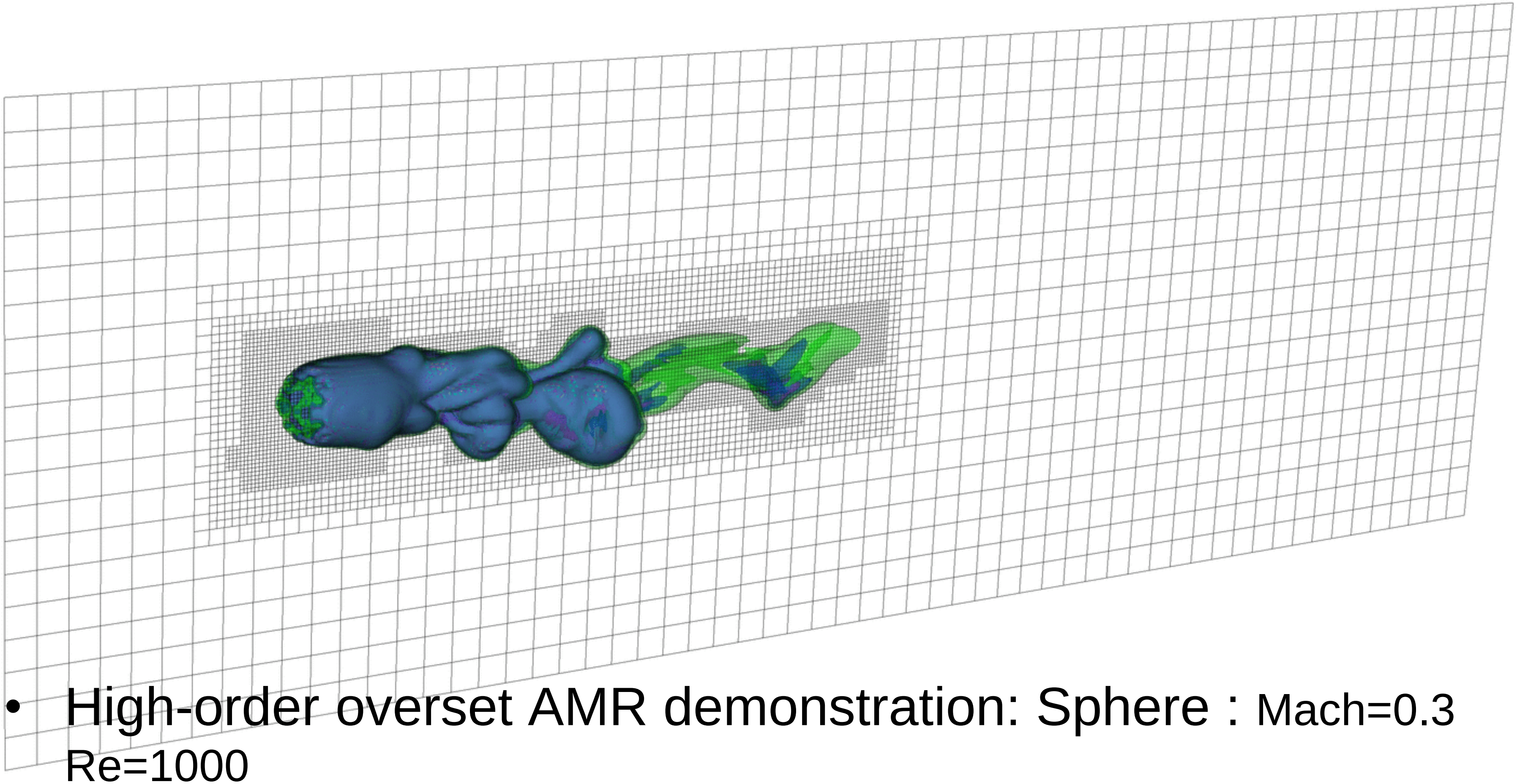




Preserves design accuracy of solver(s) (Steady Ringleb flow)

- Implemented as call-back functions in TIOGA (J. Sitaraman open source)

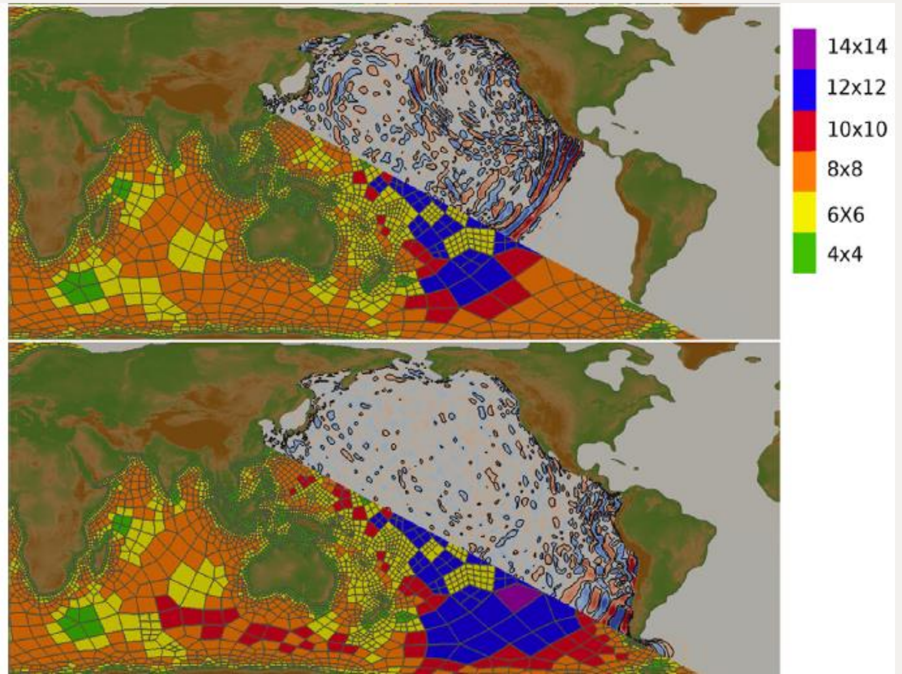
High-Order with Overset Meshes



- High-order overset AMR demonstration: Sphere : Mach=0.3 Re=1000
  - p=2 DG near-body (fixed mesh, curved elements)
  - p=3 DG off-body (adaptive mesh)
  - Good agreement with experiment Cd_{av}=0.4822 (expt: 0.48 – 0.51)

AMR High-Order Off Body Solver

- Largest benefits at very high order
- Higher accuracy with fewer mesh cells
- Simplifies AMR tasks
  - Nearest neighbor stencil
  - Coarse meshes for AMR work load
  - Ability to do h-p refinement: Exponential convergence

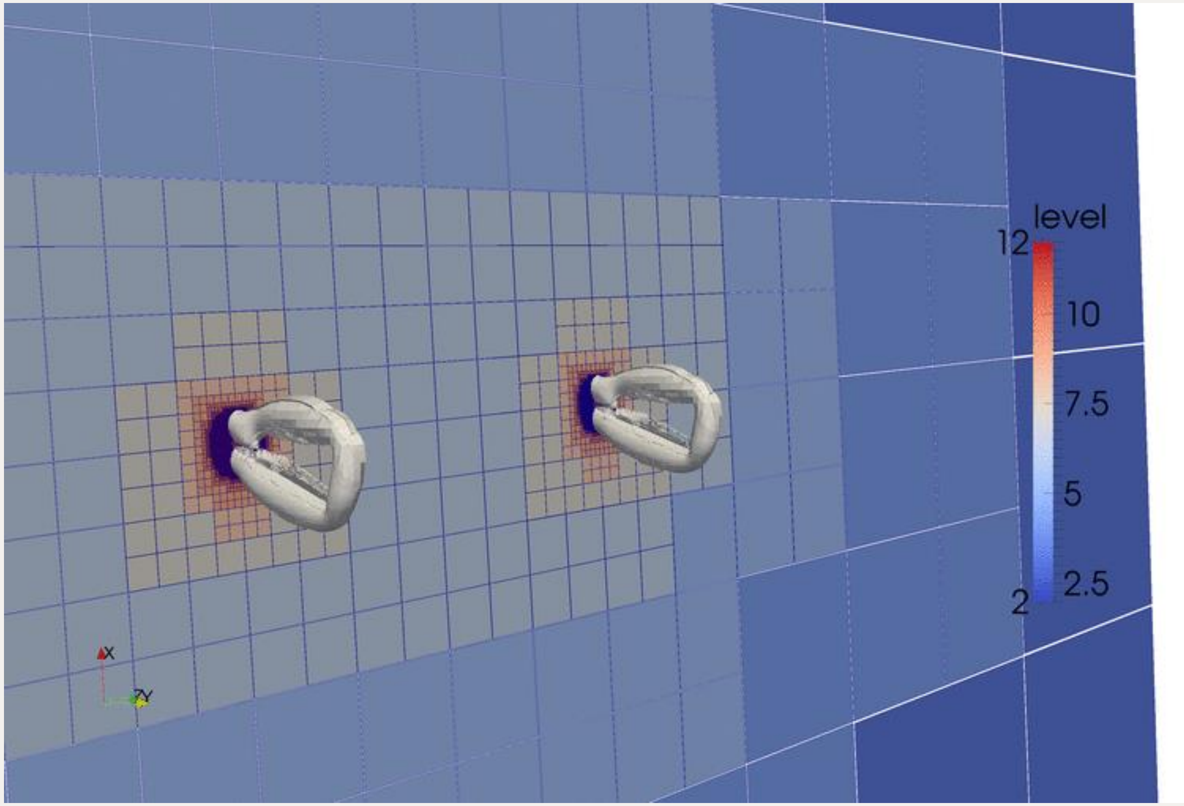


Discontinuous Galerkin unsteady discrete adjoint method for real-time efficient tsunami simulations, Blaise, St-Cyr, Mavriplis and Lockwood, JCP 2013

High-Order Adaptive DG

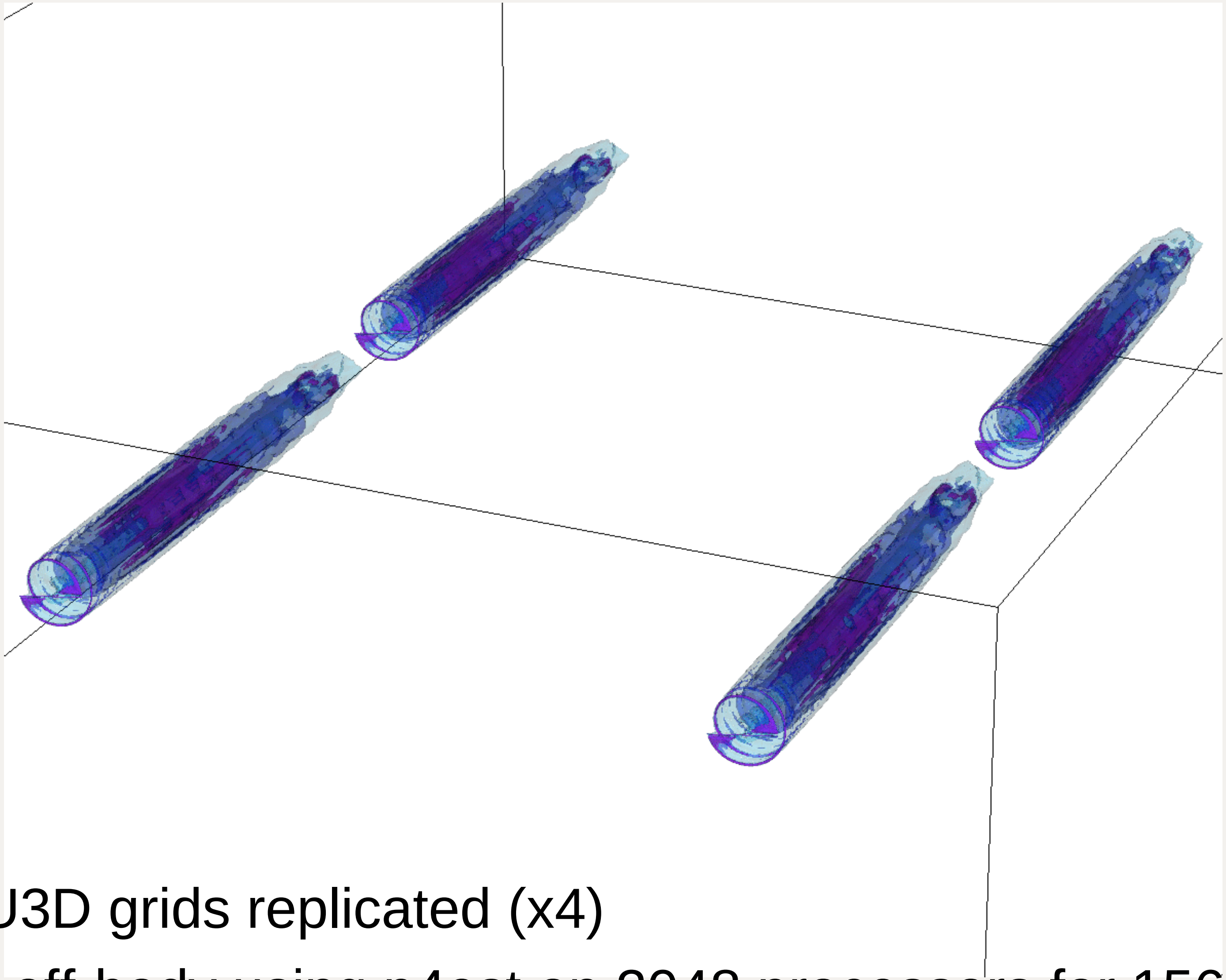
- Initial implementation SAMRAI: Patch based

- New implementation p4est: Octree based
  - ➢ Simpler implementation in p4est for DG discretization
  - ➢ Element-based viewpoint
  - ➢ Nonconforming elements handled naturally with DG



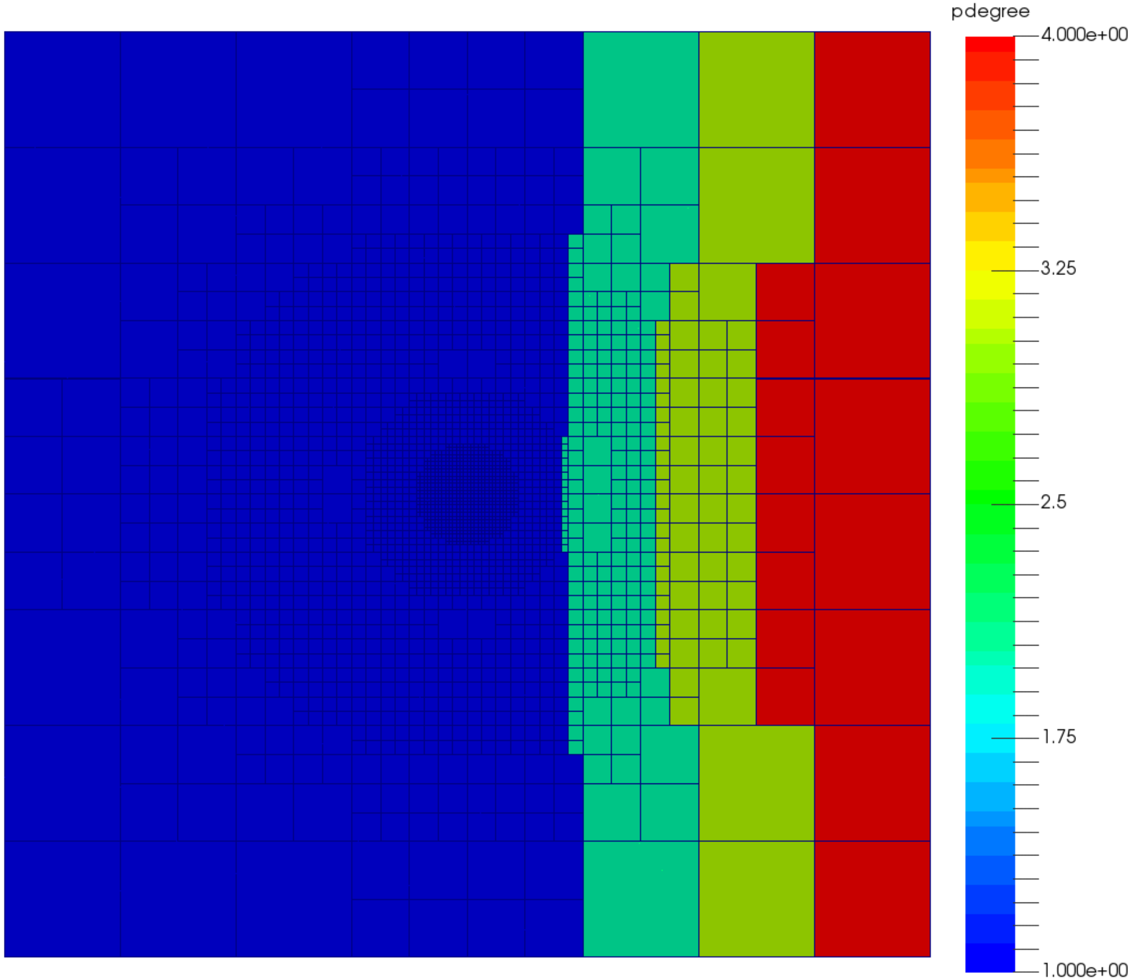
- NSU3D (finite-volume unstructured) near body
- p=1 AMR off-body (8 d.o.f. per cell)

4 Turbine Test Case using p4est



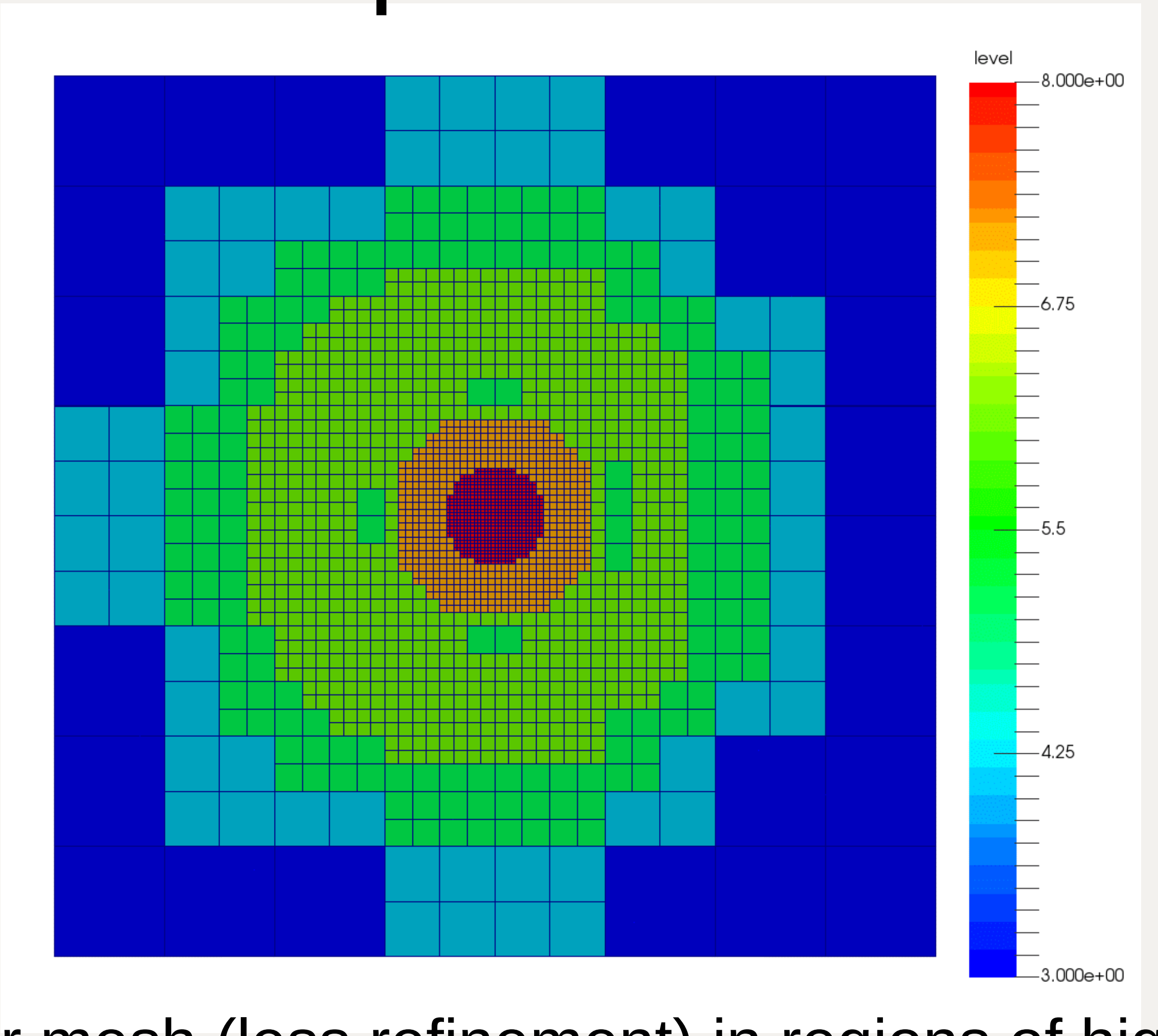
- NSU3D grids replicated (x4)
- p=1 off-body using p4est on 2048 processors for 156 hours
- ¼ degree time step, 6M cells, 48M degrees of freedom

Variable p-order with AMR



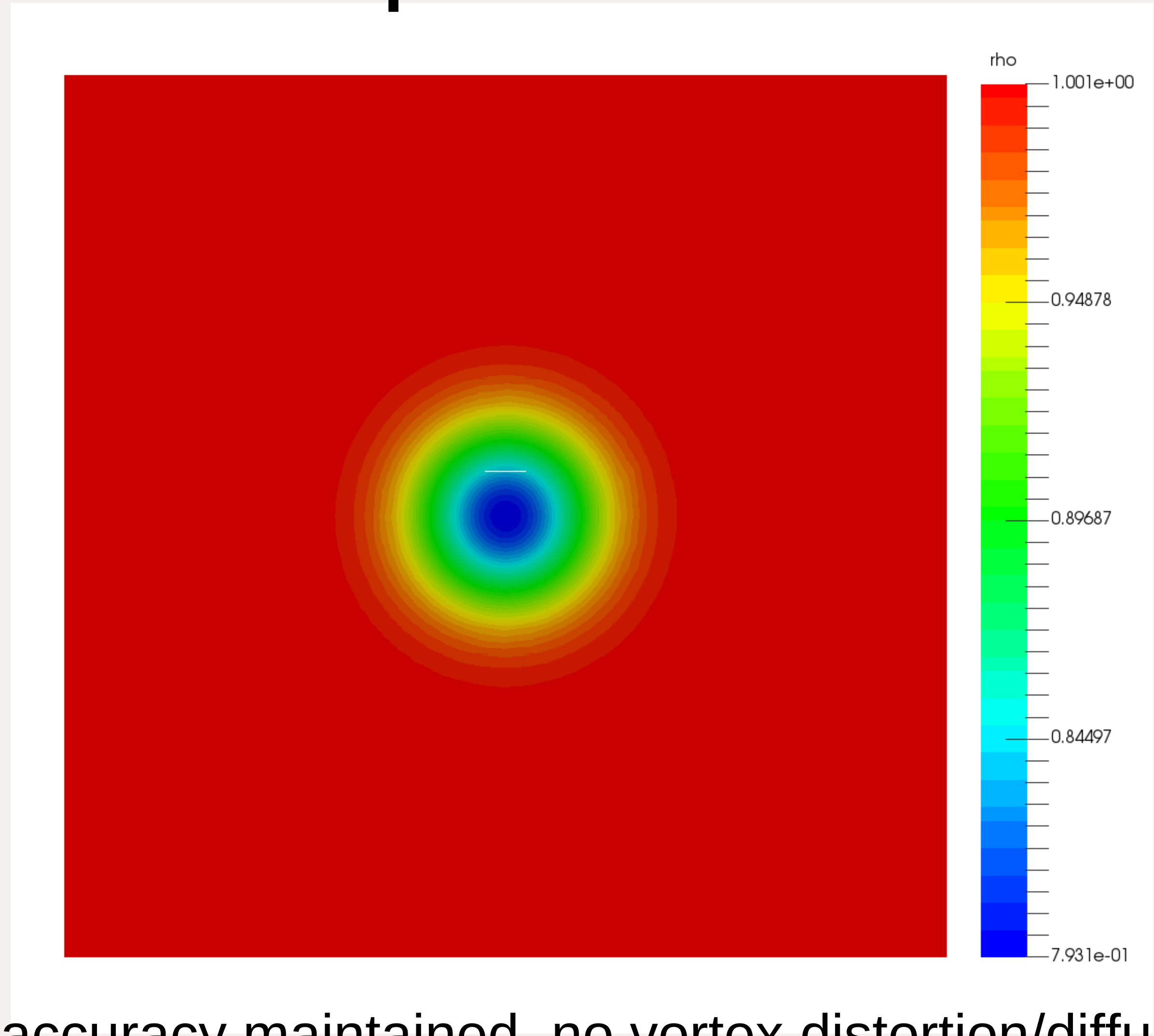
- Prescribed p-order distribution (inherited from parent cells)

Variable p-order with AMR



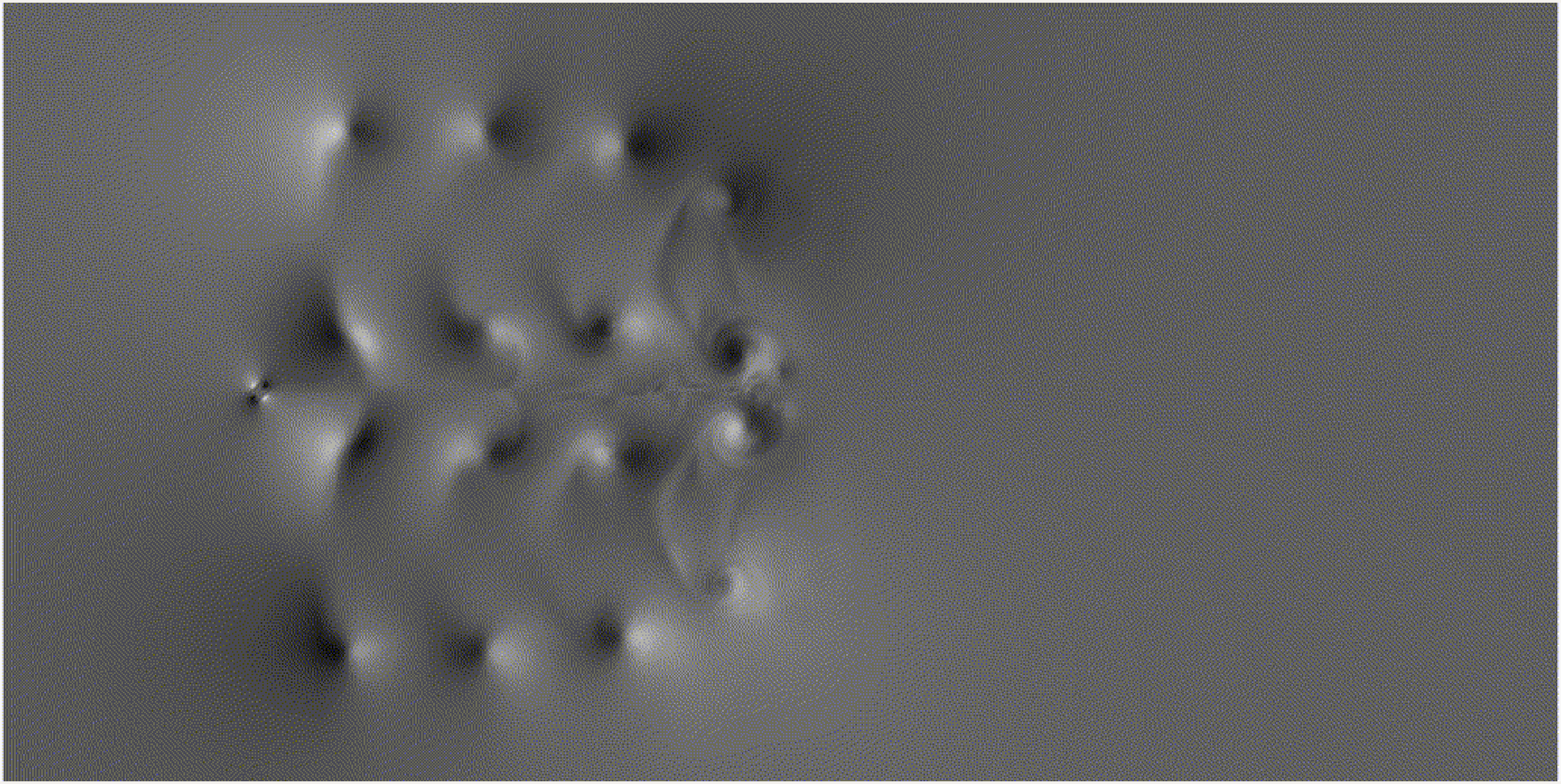
- Coarser mesh (less refinement) in regions of high p

Variable p-order with AMR



- Overall accuracy maintained, no vortex distortion/diffusion

Higher order AMR (fixed p)



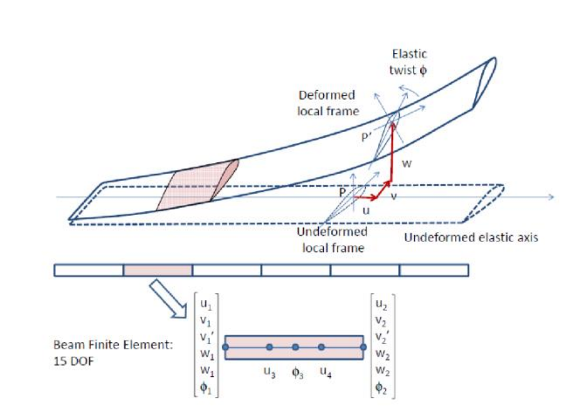
- Single turbine adaptive simulation at p=4 in wake
- High resolution
  - 1M cells, 125M degrees of freedom

Incorporating Additional Disciplines

- Structural dynamics
  - Brick/shell FEM model
  - Fluid-structure interface (FSI)
  - Mesh deformation
- Acoustics
  - FWH far-field method
- Atmospheric inflow/coupling
  - Mesoscale
  - Regional/Continental scale

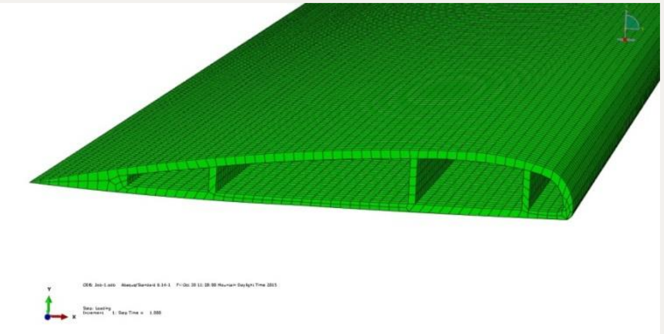
Structural Analysis

Beam model

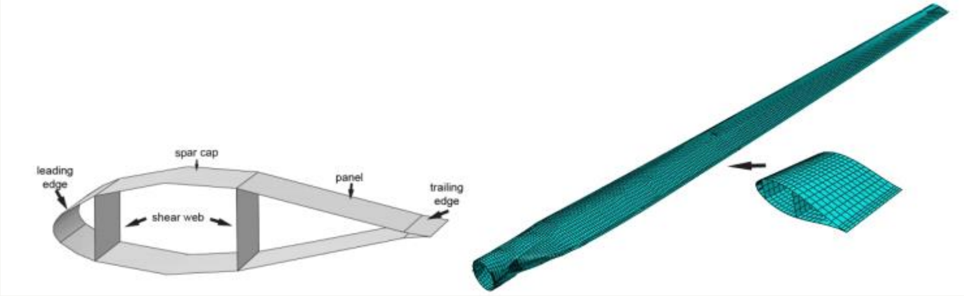


- Hodges-Dowell type finite element beam model
  - 15 degrees of freedom (flap, lag, axial and torsion)

Brick/Shell FEM model



- Brick and shell finite element models
  - In-house developed FEM code
    - Validated against industry standards (Abaqus)
    - Enables tight multidisciplinary coupling
- **Solved via direct inversion**
  - **MUMPS parallel direct solver library (INRIA)**



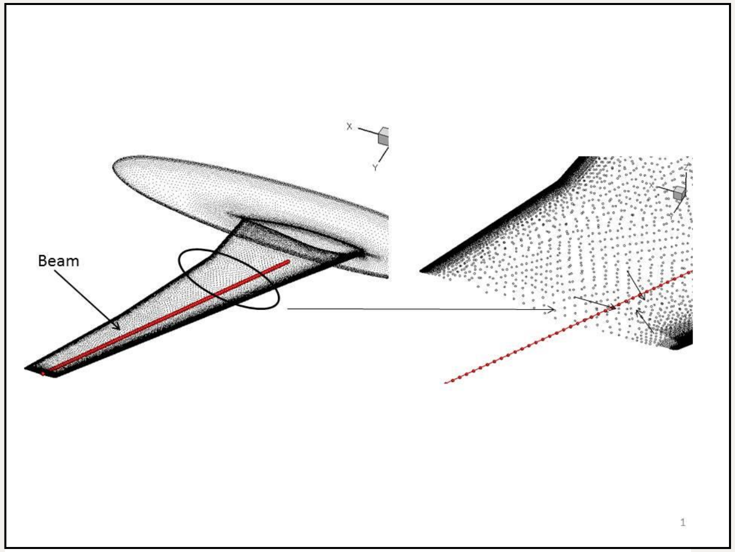
Fluid-Structure Interface (FSI)

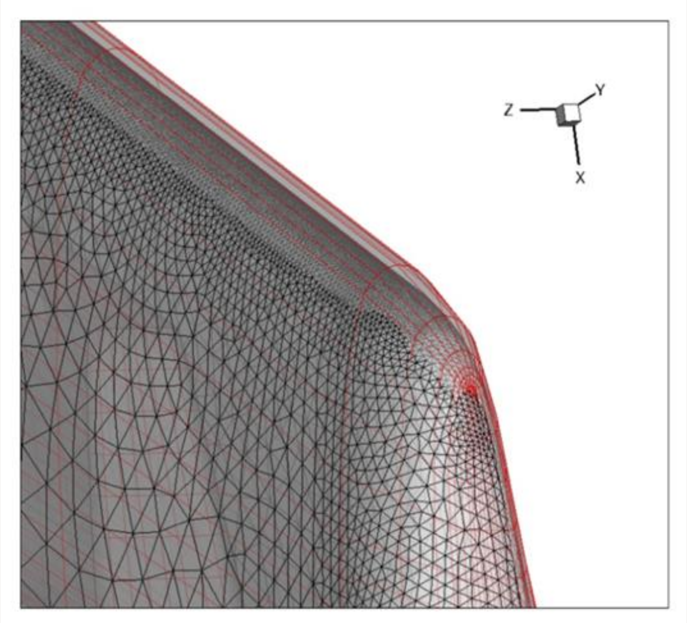
- Cloud of surface points associated with beam/FEM model
  - Must allow for mismatched surfaces
- Forces projected onto structural finite-element shape functions

$$F_{beam} = [T(Q)]F_{cfd}(x,u)$$

- Displacements projected back to CFD surface points using transpose

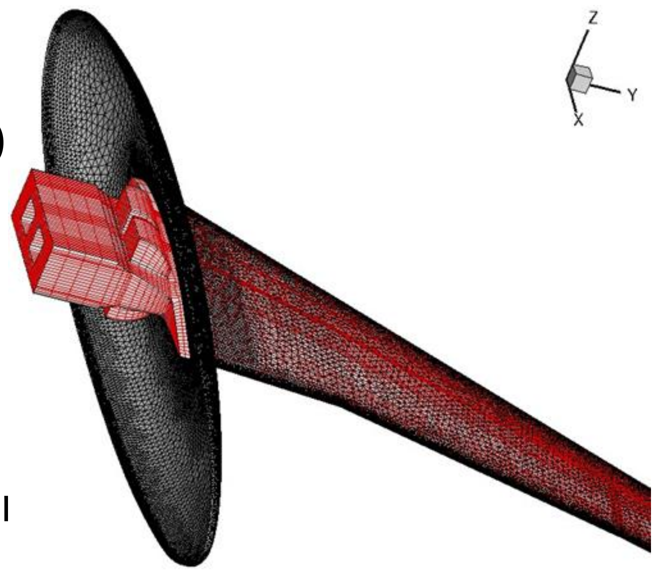
$$x_{surf} = [T(Q)]^T Q$$





CFD/CSD Coupling Time Integration Methodology

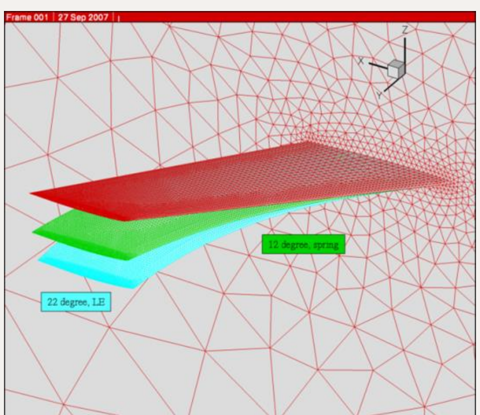
- **Outer loop over physical time steps**
  - **Coupling iterations per time step :**
    - Flow:
      - Implicit BDF2 Newton iterations (GMRES)
      - Linear agglomeration multi-grid
    - FSI (Fluid to structure)
      - Explicit assignment
    - Structure:
      - Implicit BDF2 newton iteration (direct inve
    - FSI (Structure to fluid)
      - Explicit assignment
    - Mesh deformation:
      - Line implicit multigrid

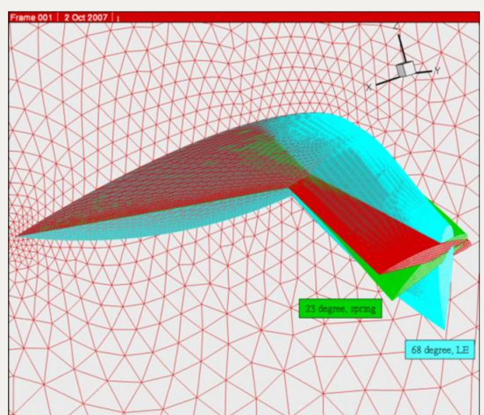


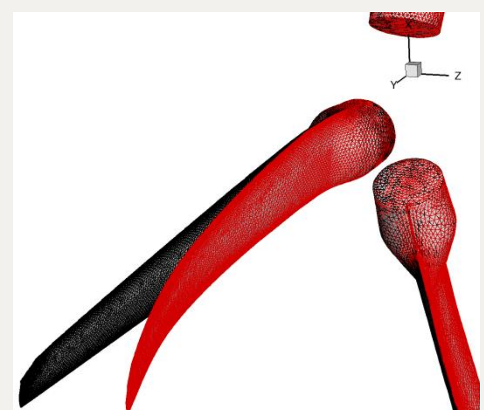
Mesh Deformation

- Propagates surface displacements to interior mesh
  - Deflections from structural model at each time step (x^n)
  - Design shape changes (D)
- Based on linear elasticity analogy
  - (more robust than spring analogy)
- Solved using line-implicit agglomeration multigrid (analogous to flow solver)

$$G(x^n, x^n_{surf}, D) = 0$$



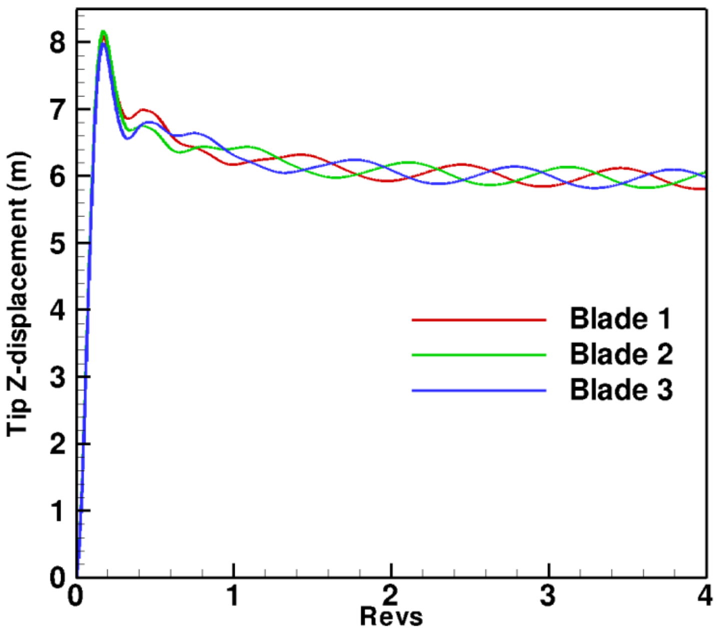




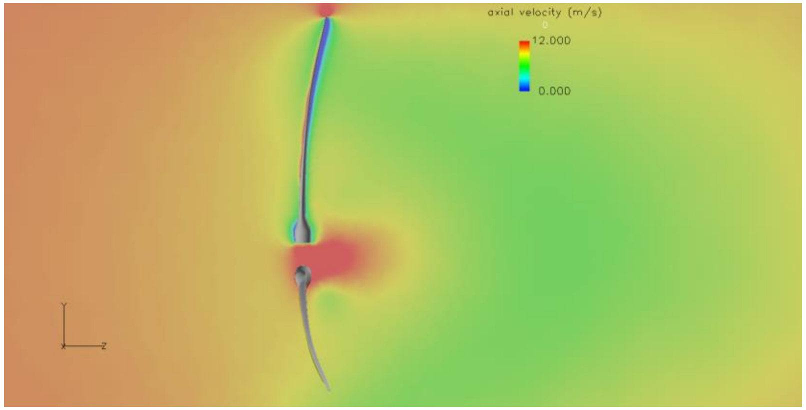
Flexible Rotor
NREL 5MW (63m radius)

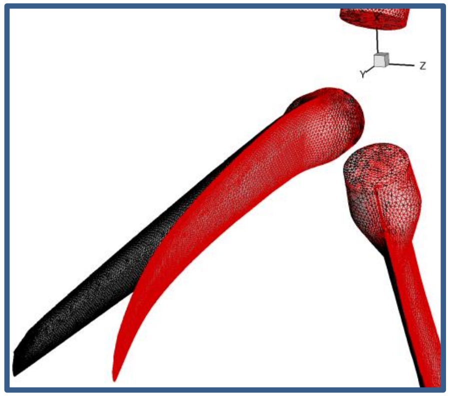
Blade tip vs time

(m)



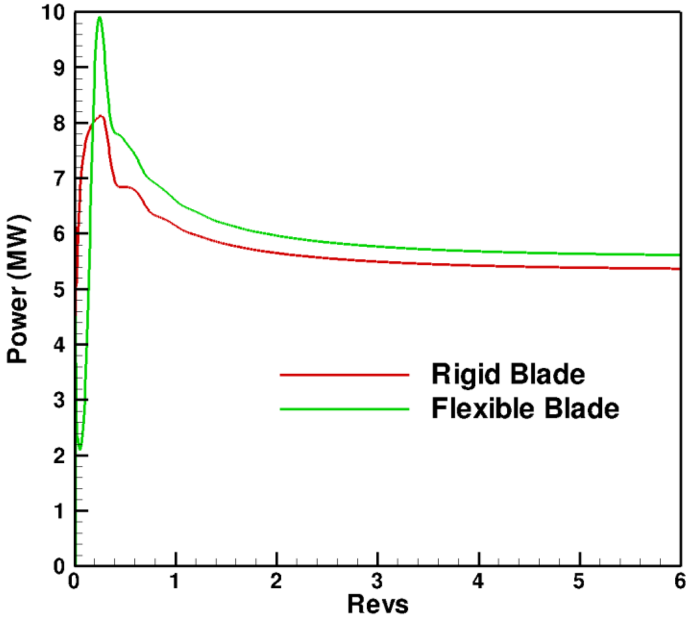
Instantaneous Axial Velocity



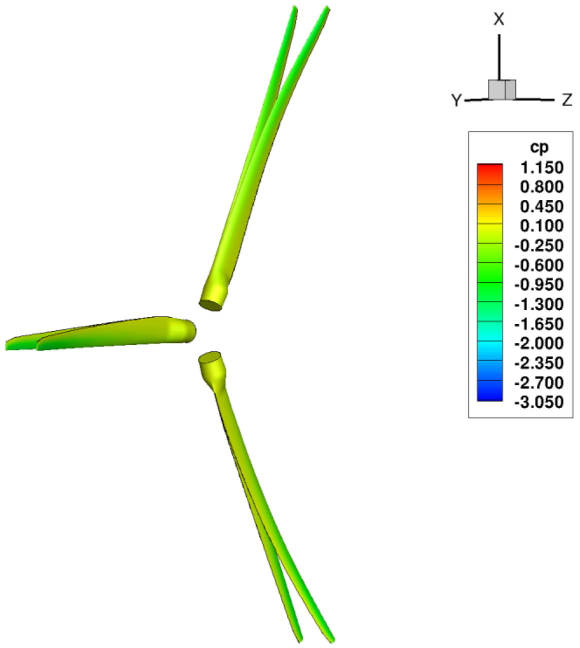


- Blade flaps to high values, but converges to average 5.93m deflection with expected behavior
  - Periodic, 120^0 phase shift between 3 blades

NREL 5MW Performance Prediction



Power vs time



Surface Cp for rigid/flexible blades

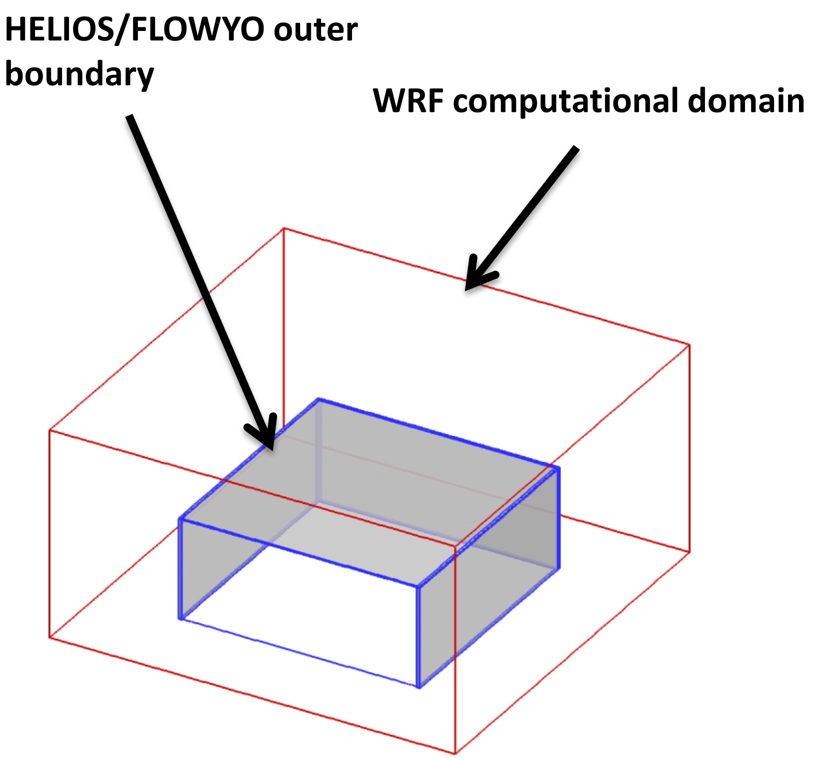
- Aeroelastic simulation predicts 5% higher power output
- Final value approximately 5.8MW (still decreasing)

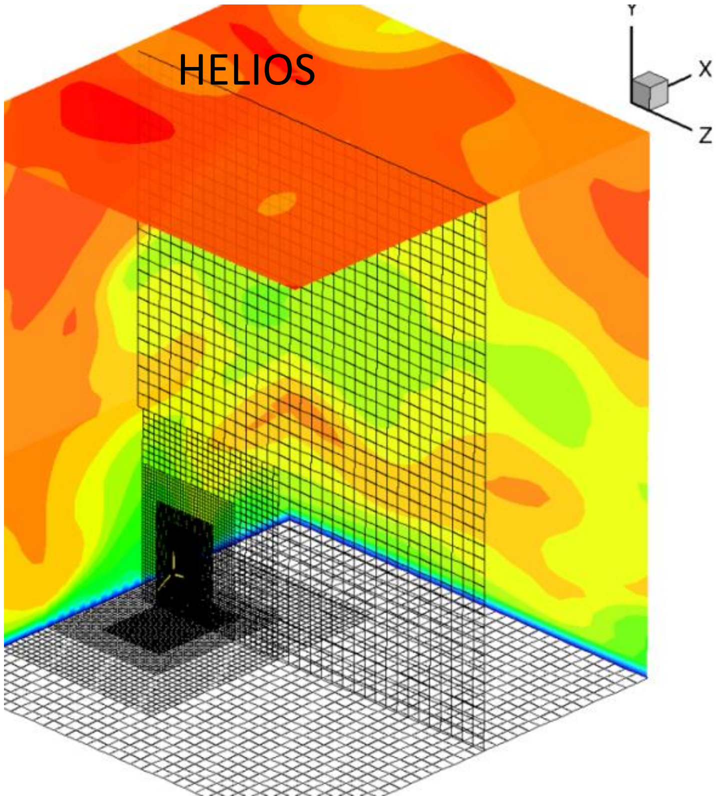
Atmospheric Turbulence Inflow Coupling Interface

Mesoscale flow computed using Weather Research and Forecasting (WRF) model
Synthetic inflow also possible using the Mann (1984) model

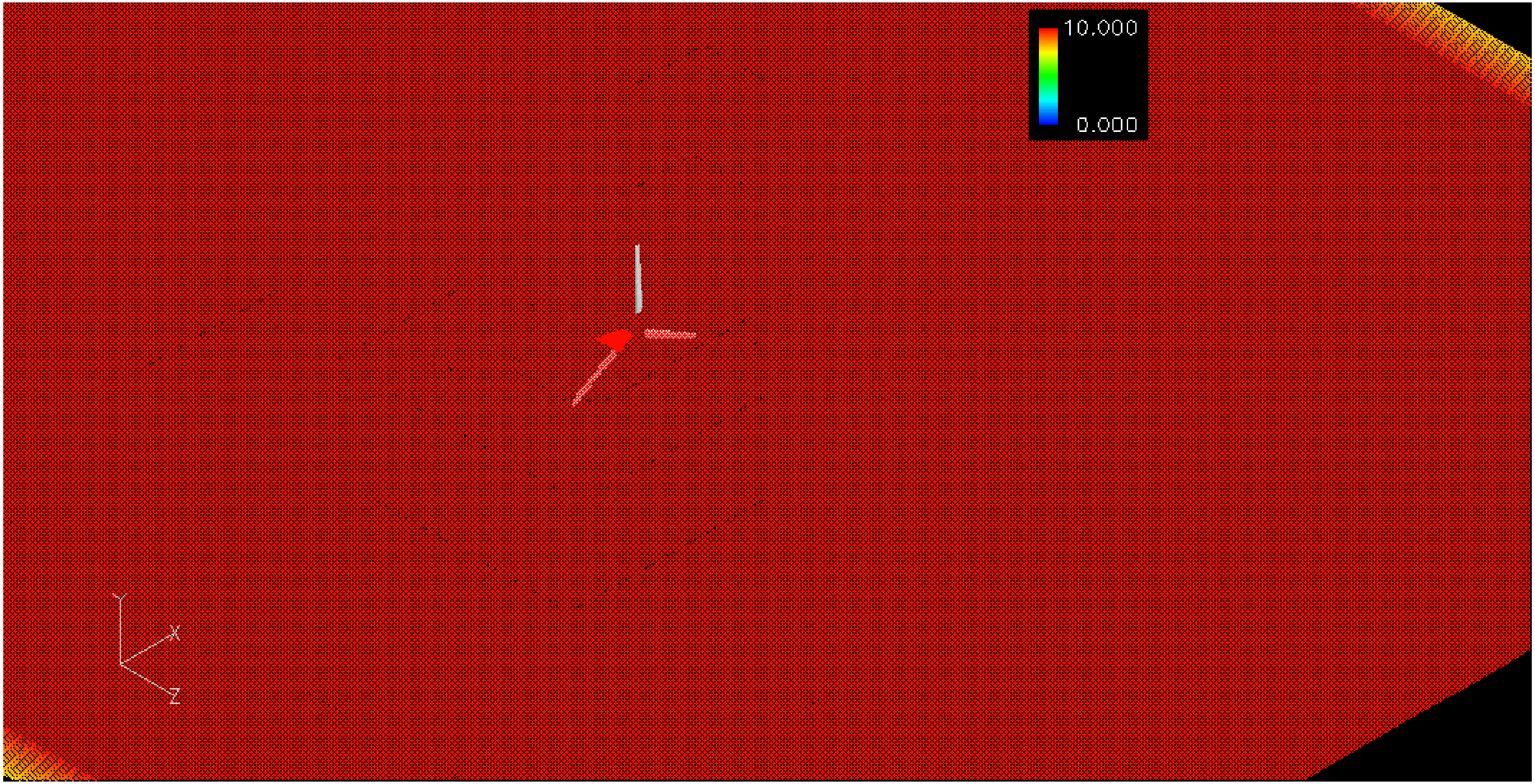
Precursor meso-scale simulations and one-way coupling

Transfer velocity and SGS turbulence quantities at the interface of the micro-scale solver (HELIOS, FLOWYO (actuator-line) ,UWAKE (free-vortex wake)) using an overset grid approach details in Gopalan et al. AIAA 2013



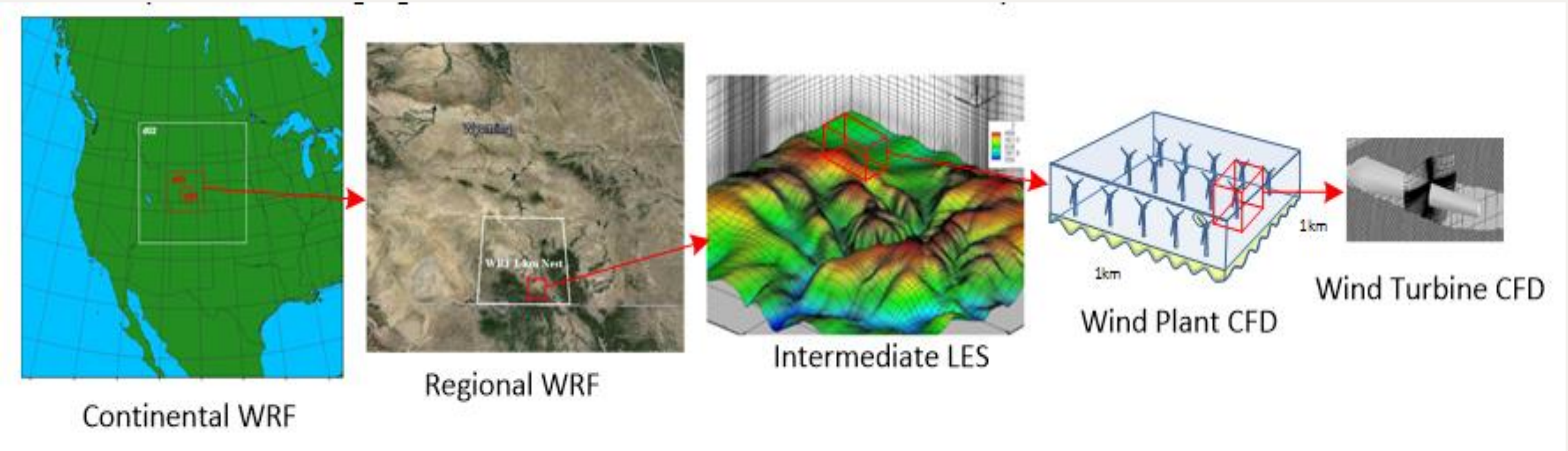


Sexbierum turbine (turbulent inflow)
HELIOS simulation



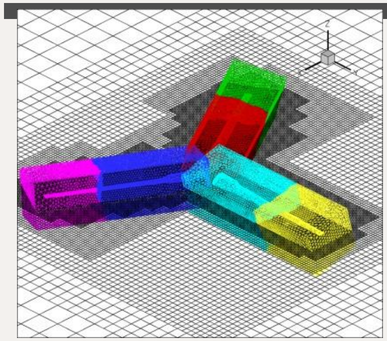
Atmospheric turbulent inflow:
One way coupling WRF (Weather Research Forecast) → HELIOS

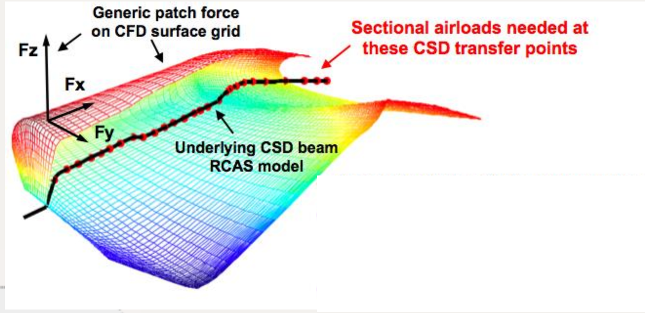
Multiscale Atmospheric Inflow

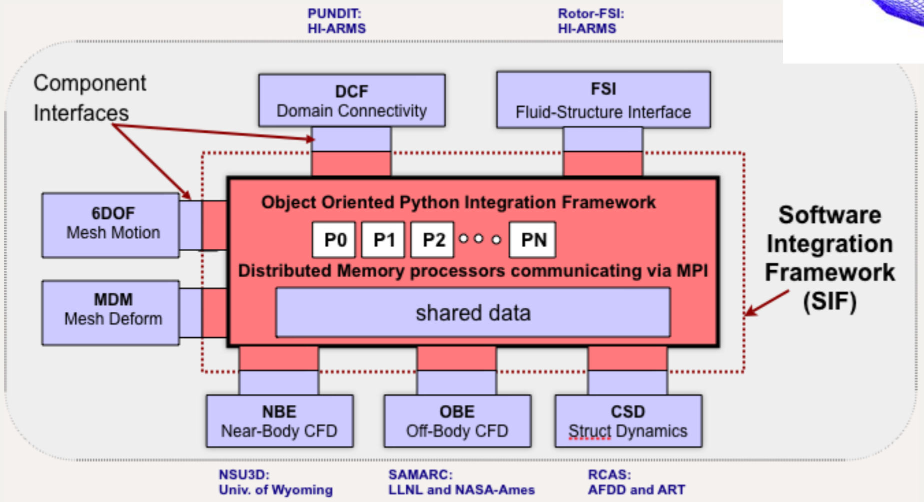


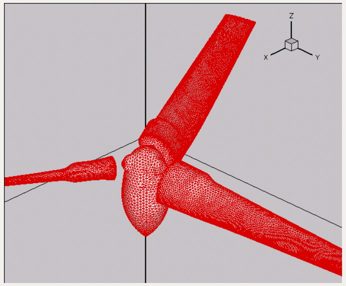
- Weather Research Forecasting (WRF) model at continental to regional scales
- Atmospheric boundary layer LES over complex terrain (OpenFoam) (M. Stoellinger (UWyoming))
- One way coupling only

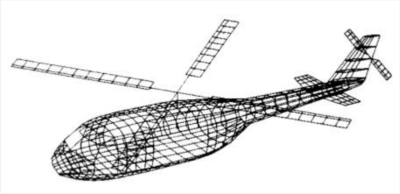
Multidisciplinary Software Components

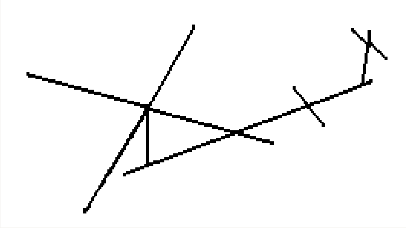


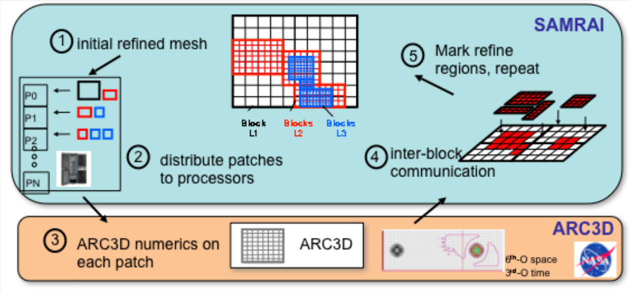




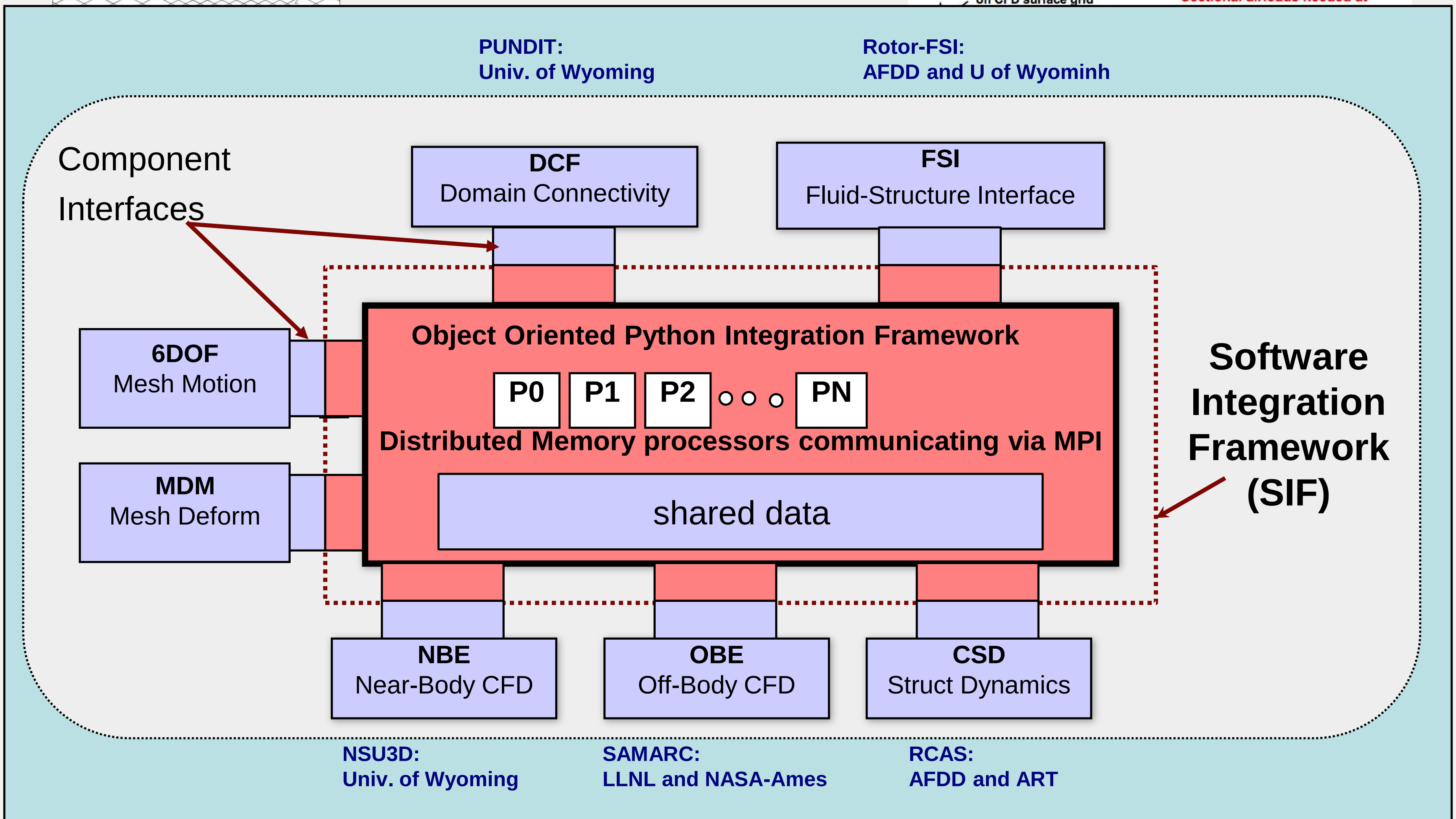




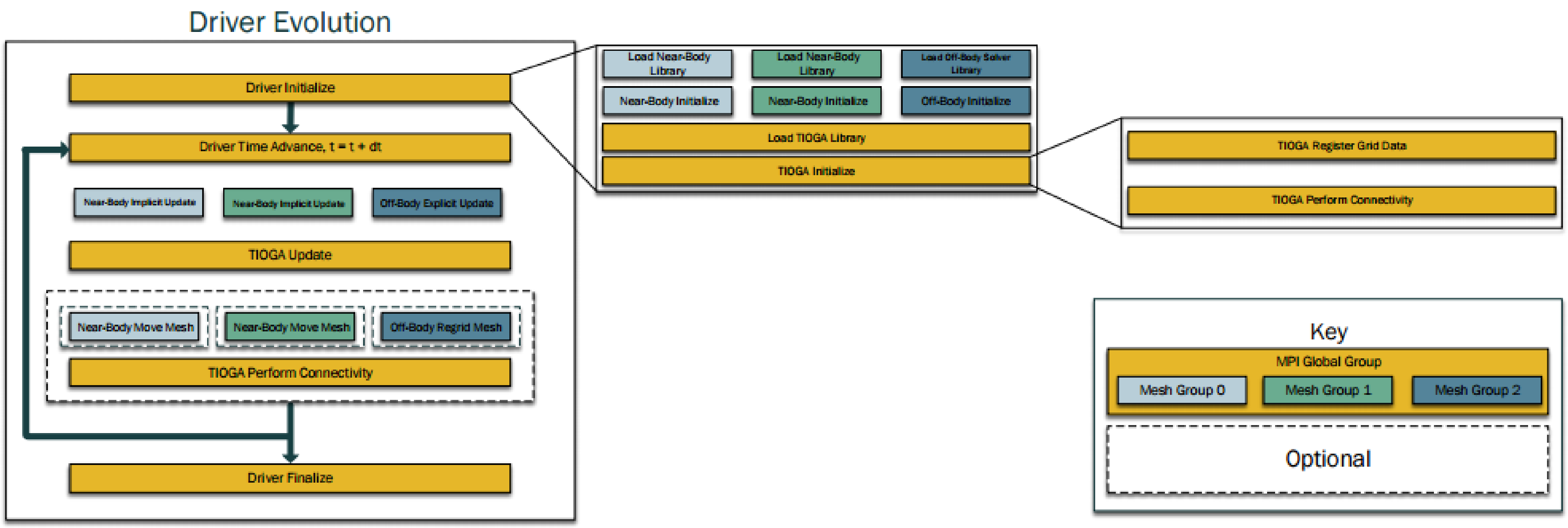




Multidisciplinary Software Components



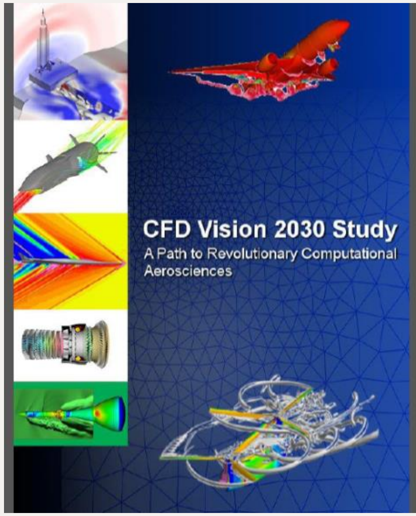
C++ Modular Driver



- ISO-C binding
- Extensible
- All codes run simultaneously
  - Load balancing pros and cons

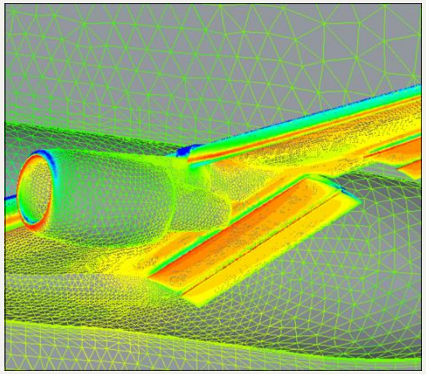
Requirements for Complex Simulations

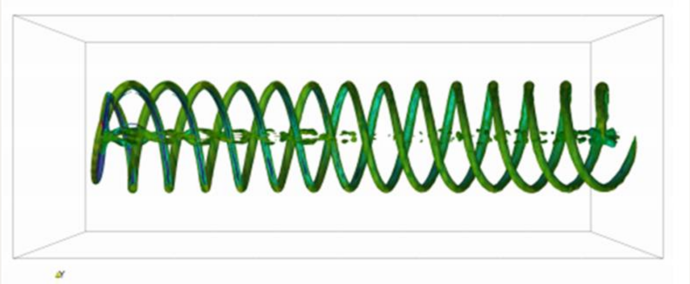
- Wind energy application similar to aircraft/propulsion certification problem
  - – Large scale (HPC)
  - – Time critical (HPC)
  - – Multidisciplinary (Software complexity)
    - CFD, Structures, Acoustics, Controls …
  - – Verifiable and Validated
    - provable accuracy
  - – Knowledge extraction

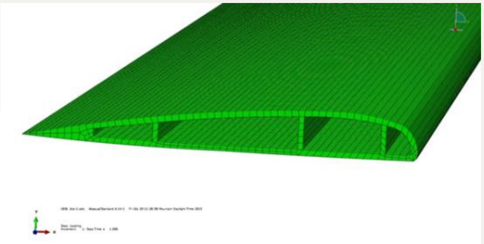


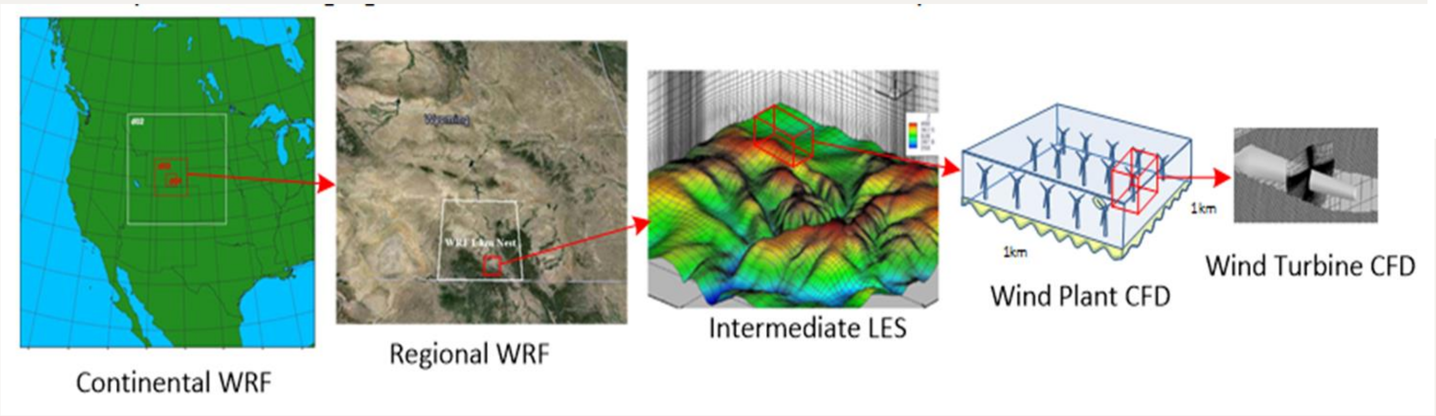
Required Components

- Physics models
  - Near body solver (unstructured)
  - Off body solver (Cartesian)
  - Structural model
  - Controls
  - Acoustic model
  - Atmospheric Boundary layer
  - Regional/Continental Weather model



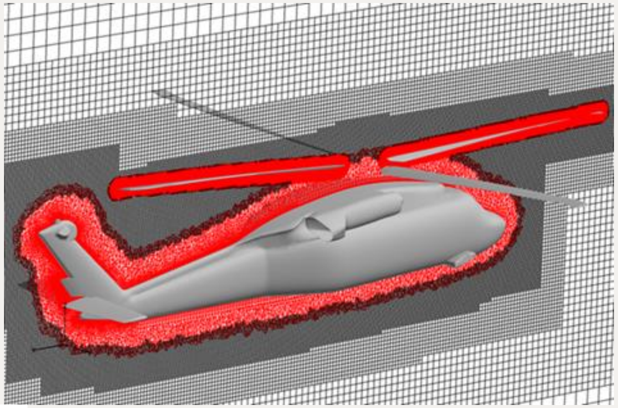


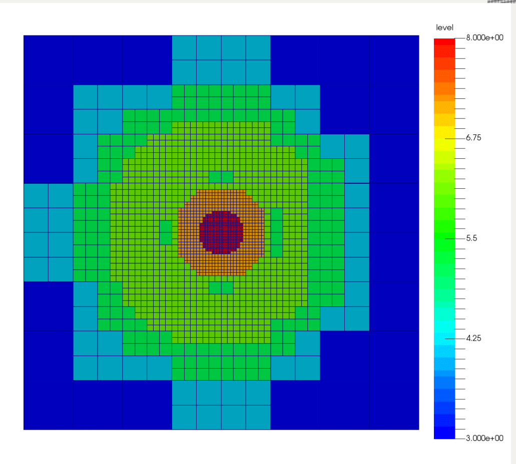


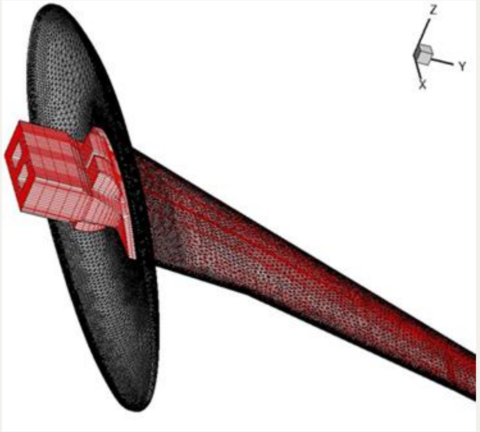


Required Components

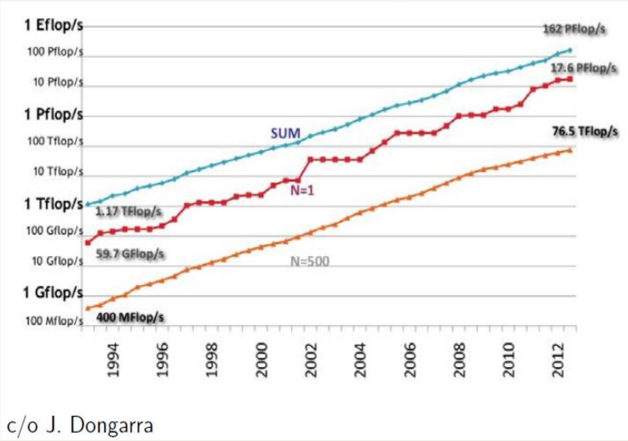
- Physics models
- Enabling glue
  - Adaptive AMR
  - Overset
  - FSI
  - Mesh deformation
- Performance/HPC
  - BLAS
  - MPI/OMP
  - Solver libraries
- Pre/Postprocessing
  - Initial mesh generation
  - Visualization (in-situ)





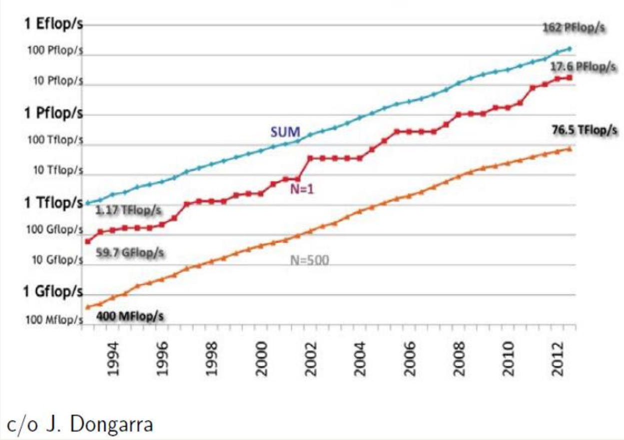


Slow Progress



- 6 orders of magnitude increase in computational power/ 20 yrs
- More complex and accurate simulations
  - 1M to 100M grid points (single discipline)
    - Overset Symposium presentations
    - Drag Prediction Workshop (2001-2016)
- Not keeping pace with other fields
  - Promise of higher-order methods unrealized
  - Adaptive meshing seldom used
  - Software complexity

Slow Progress

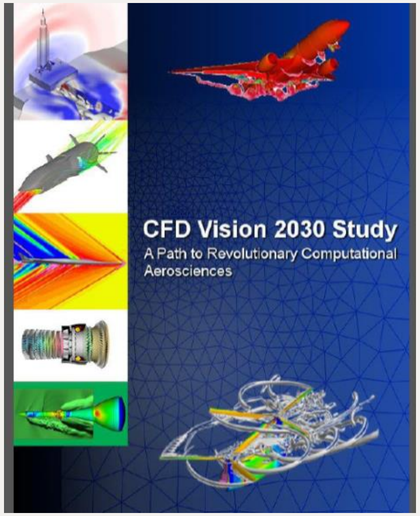


- 6 orders of magnitude increase in computational power/ 20 yrs
- More complex and accurate simulations
  - 1M to 100M grid points (single discipline)
    - Overset Symposium presentations
    - Drag Prediction Workshop (2001-2016)

- Not keeping pace with other fields
  - Promise of higher-order methods unrealized
  - Adaptive meshing seldom used
  - Software complexity

Achieving the Vision

- A community effort:
  - Advances in basic algorithmic techniques
    - Papers, publications, demonstrations
  - Harnessing HPC
    - Paradigms: MPI/OMP, OpenACC, CUDA, BLAS, etc.
  - Shared software at appropriate level
    - – Overset
    - – AMR
    - – Solvers
    - – In-situ visualization
    - Multiple approaches/implementations
      - – One size never fits all



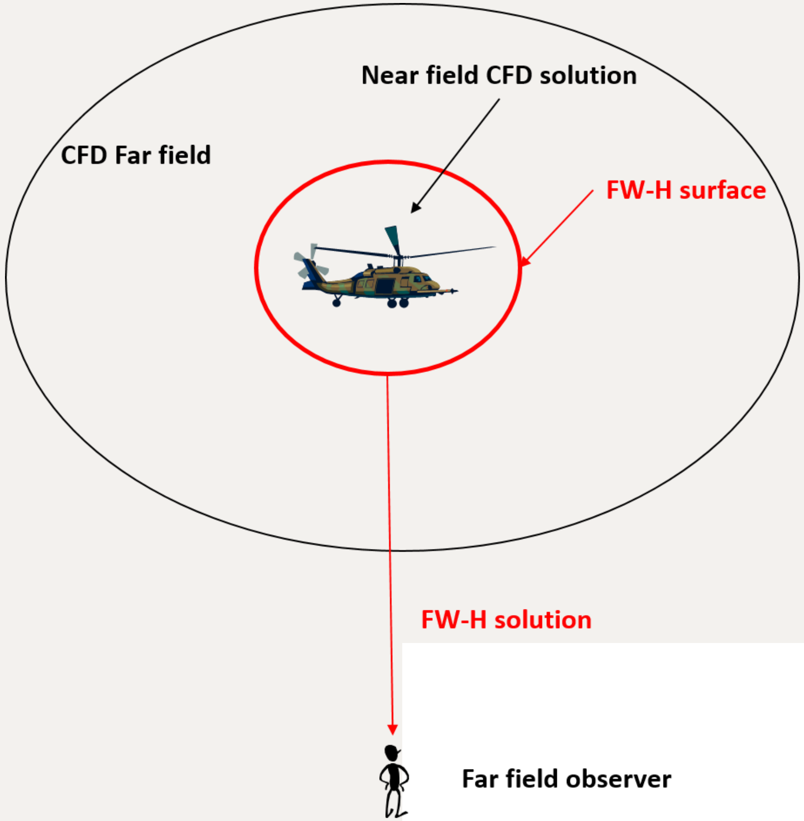
Acknowledgements

- Jay Sitaraman (PGA LLC), M.Stoellinger and R. Fertig (U of Wyoming)
- Dr. Michael Brazell, Dr. Behzad Ahrabi
- Andrew Kirby, Enrico Fabiano, Evan Anderson, (PhD Students)

Acoustic Propagation

Ffowcs-Williams and Hawkings (FW-H)

FW-H: Formulation 1A with source-time dominant algorithm. Linear pressure interpolation at the observer. Quadrupole term neglected.



$$U_i = \left(1 - \frac{\rho}{\rho_o}\right) v_i + \frac{\rho u_i}{\rho_o}$$

$$L_i = p' n_j + \rho u_i (u_n - v_n)$$

$$U_n = U_i n_i$$

$$M_r = M_i r_i$$

$$L_r = L_i r_i$$

$$L_M = L_i M_i$$

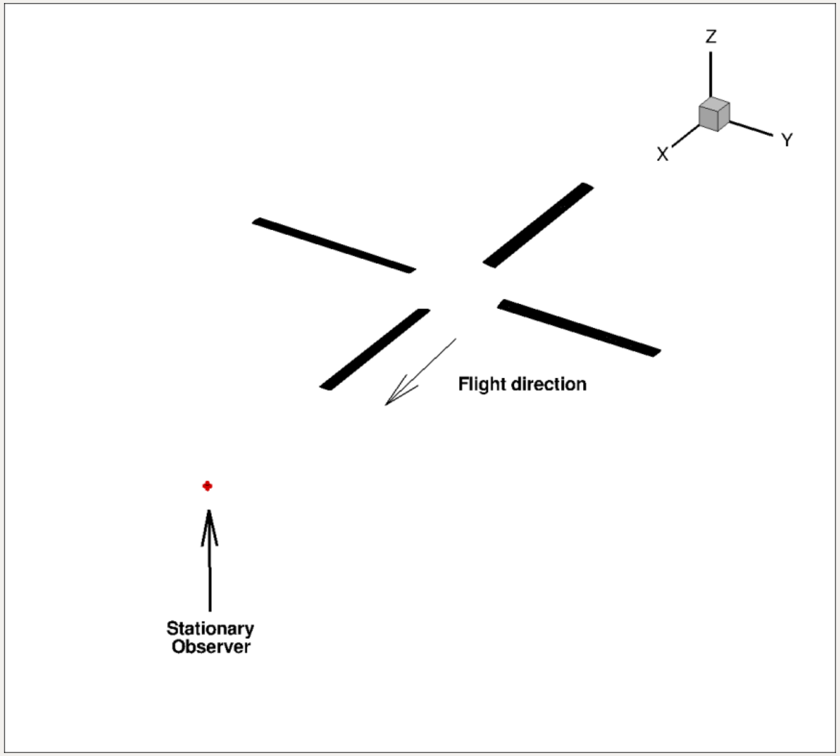
$$K = r\dot{M}_r + c_o (M_r - M^2)$$

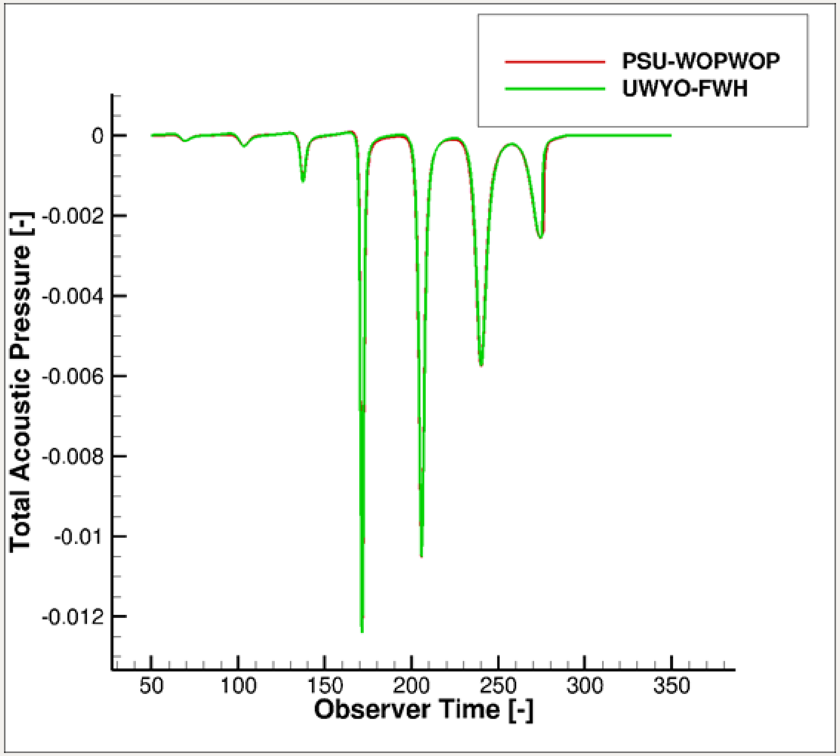
$$4\pi p'(\mathbf{y},t) = 4\pi p'_T(\mathbf{y},t) + 4\pi p'_L(\mathbf{y},t)$$

$$4\pi p'_T(\mathbf{y},t) = \int_{f=0} \left[\frac{\rho_o (\dot{U}_n + U_{\dot{n}})}{r(1-M_r)^2}\right]_{ret} dS + \int_{f=0} \left[\frac{\rho_o U_n K}{r^2(1-M_r)^3}\right]_{ret} dS$$

$$4\pi p'_L = \frac{1}{c_o} \int_{f=0} \left[\frac{\dot{L}_r}{r(1-M_r)^2}\right]_{ret} dS + \int_{f=0} \left[\frac{L_r - L_M}{r^2(1-M_r)^2}\right]_{ret} dS + \frac{1}{c_o} \int_{f=0} \left[\frac{L_r K}{r^2(1-M_r)^3}\right]_{ret} dS$$

Acoustic Problem Validation





Hart-II rotor in trimmed forward flight.

$$M_\infty = 0.095 \; - M_{tip} = 0.638$$

Stationary observer in the plane of the rotor one radius ahead.

Solid wall acoustic integration.