11th Symposium on Overset Composite Grids and Solution Technology

Wright Patterson Air Force Base
Dayton, OH USA

Oct 17 2012

Helios Solver Developments Including Strand Meshes

TECHNOLOGY DRIVEN. WARFIGHTER FOCUSED.

Collaborators:

Jayanarayanan Sitaraman, *U. Wyoming*
Aaron Katz, *Utah State*
Robert Haimes, *MIT*

Presented by:

Andrew Wissink

U.S. Army Aeroflightdynamics Directorate

Aviation and Missile Research, Development and Engineering Center
Moffett Field, CA

Presentation Outline

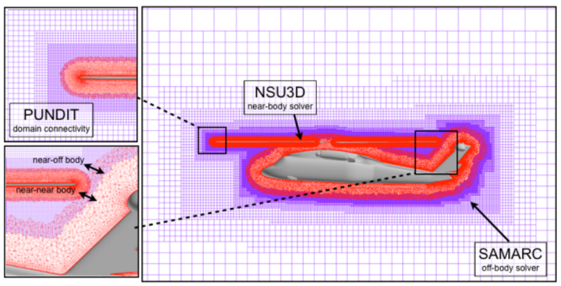
- Overview
- New capabilities
- Sample Results
- Strand Solver Developments
- Summary

Version 3 *Rainier*

Helios Software

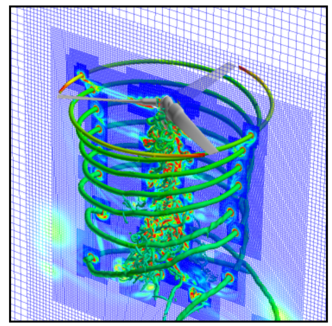
Helios Helicopter Overset Simulations

Dual Mesh Paradigm



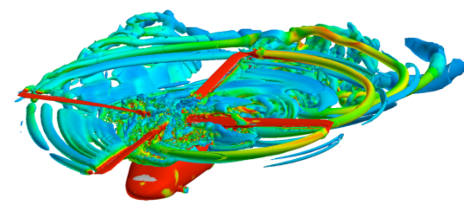
Unstructured Near-body
Cartesian off-body

Adaptive Mesh Refinement



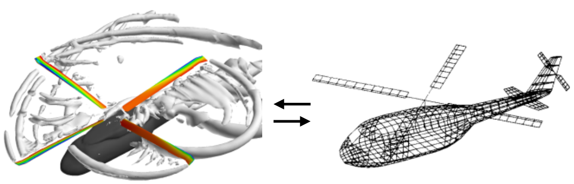
To resolve wake

Moving Body Overset



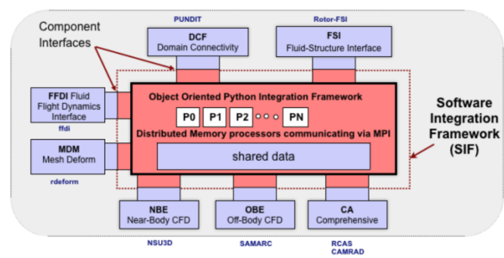
Rotor-Fuselage and Multi-rotor moving mesh support

CFD/CSD Coupling



RCAS and CAMRAD Structural Dynamics and Trim coupling

Advanced Software Infrastructure



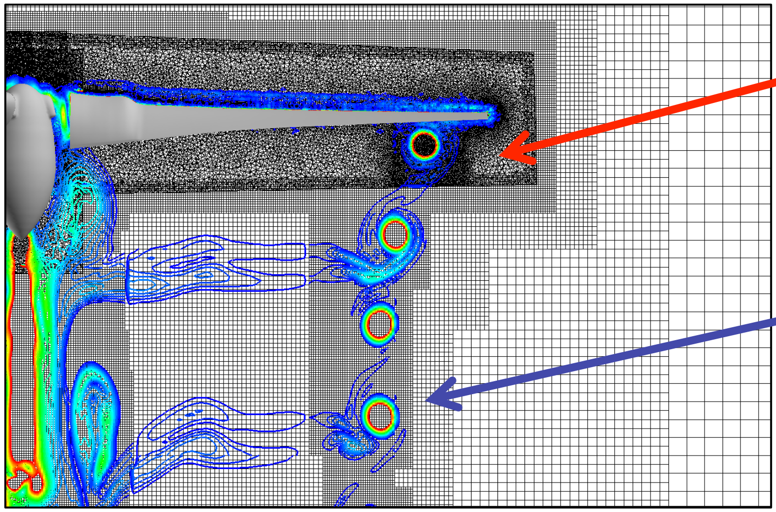
Python-based infrastructure readily supports addition of new software

High Performance Computing

Runs on HPC hardware with focus on parallel scalability

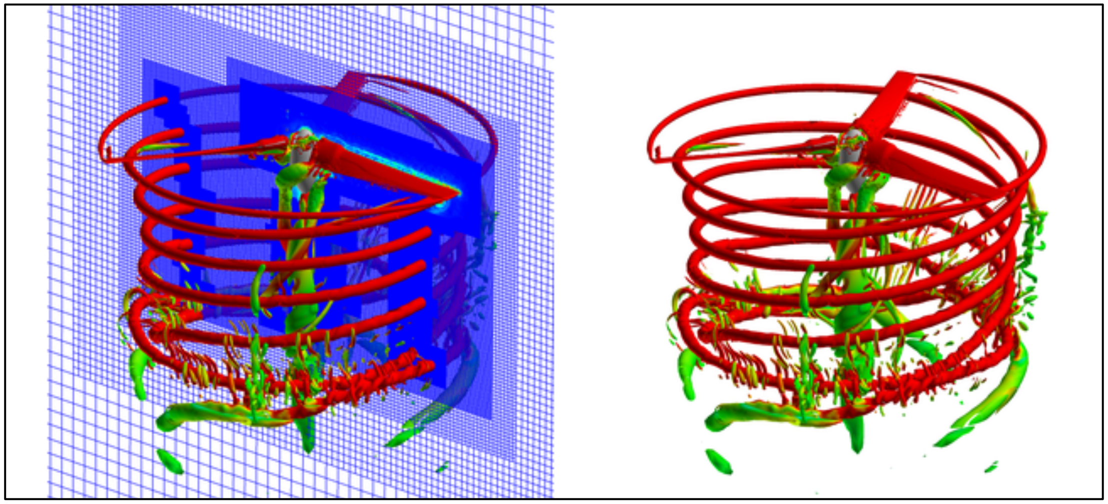
Supports high-fidelity rotary-wing simulation by government and industry
Developed & maintained by a team at Army AFDD.

Dual Mesh CFD Paradigm



Unstructured "near-body"
- Resolve near-wall viscous flow
- Complex geometries

Cartesian "off-body"
- Computationally efficient
- High order accuracy
- Adaptive Mesh Refinement

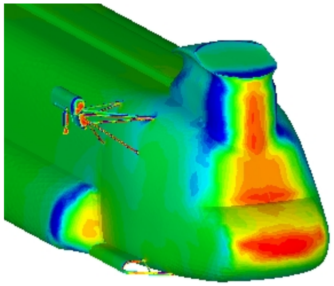


Effective for time-dependent/ moving-body applications

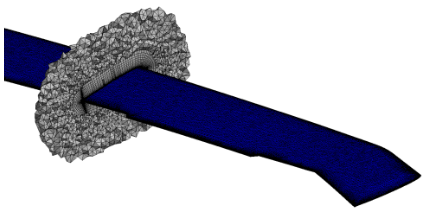
CFD Components

- **Near-body NSU3D flow solver**
  - *Developed by Mavriplis at Univ. of Wyoming*
  - General unstructured – tets, prism, hex
  - Reynolds-averaged Navier-Stokes
  - Spalart-Allmaras turbulence model
  - 2nd-Order vertex-based spatial discretization
  - 2nd-Order BDF time integration

fuselage



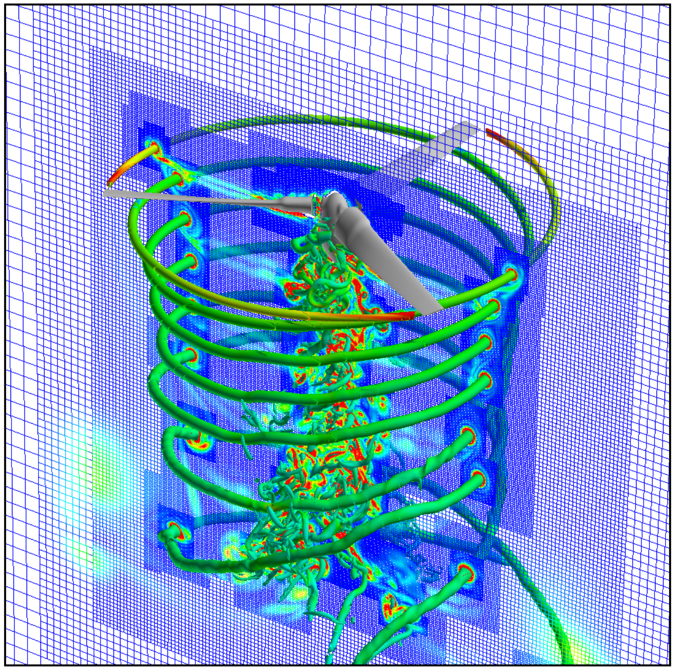
rotor blade



NSU3D
near-body

- **Off-body SAMARC flow solver**
  - Couples LLNL SAMRAI with NASA Ames ARC3D
  - Block structured Cartesian
  - 5th-Order spatial discretization
  - 3rd-Order explicit Runge-Kutta time
  - Automated AMR

SAMARC
off-body



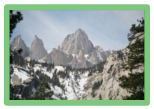
- **Overset Communication**
  - *PUNDIT*
  - Automated implicit hole cutting

RDECOM

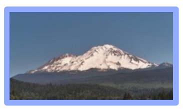
Helios Released Capabilities

AMRDEC

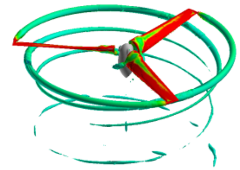
- **Version 1 *Whitney***
  - Dual-mesh unstructured/Cartesian
  - Fixed off-body
  - Isolated rotor/fuselage



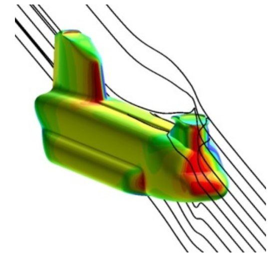
- **Version 2 *Shasta***
  - Adaptive Mesh Refinement (AMR)
  - Rotor-Fuselage



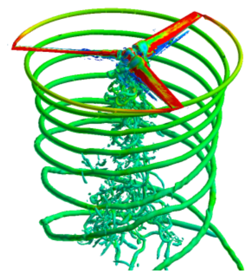
- **Version 3 *Rainier***
  - Automated AMR
  - Multi-rotor
  - DES Turbulence modeling
  - Coviz

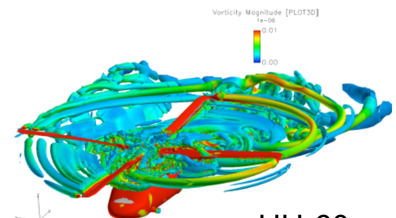


TRAM

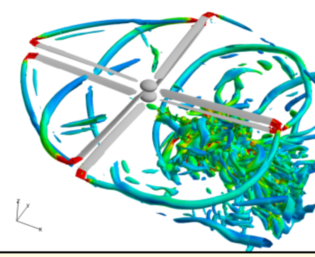


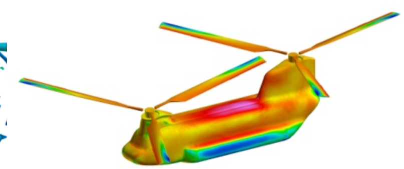
CH-47



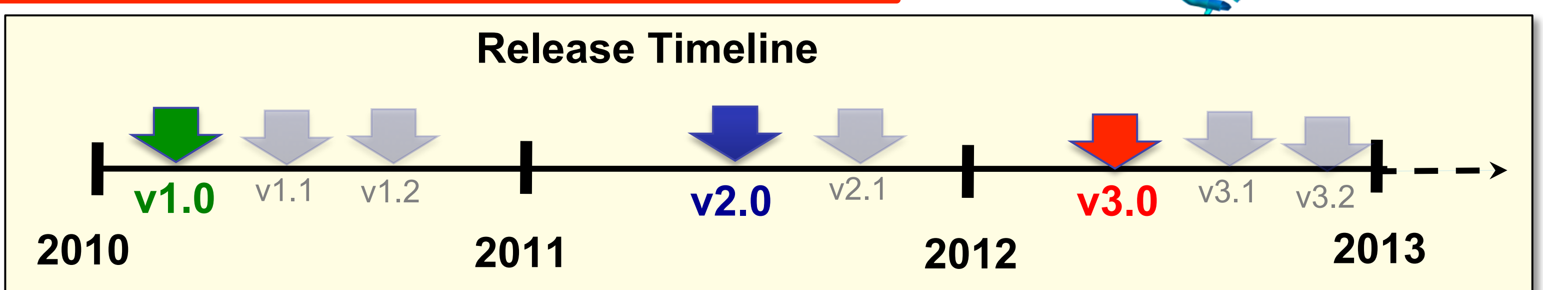


UH-60



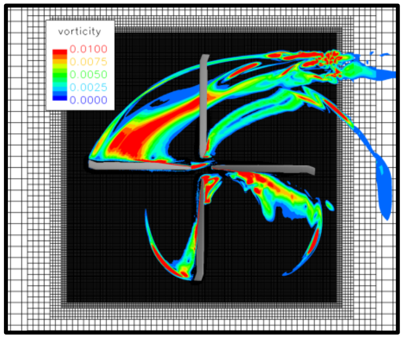


CH-47



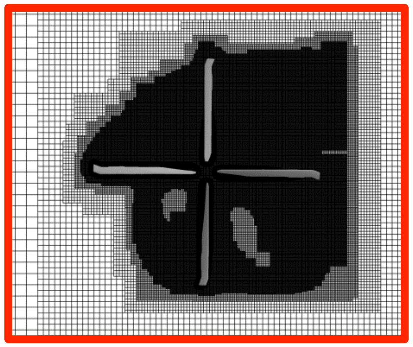
Automated Off-Body AMR

- **Why not simply refine to vorticity or Q-criterion?**
  - User dials in feature and quantity to adapt to (ω_{adapt} or Q_{adapt})

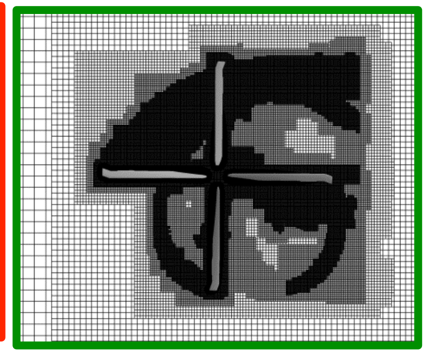


fixed

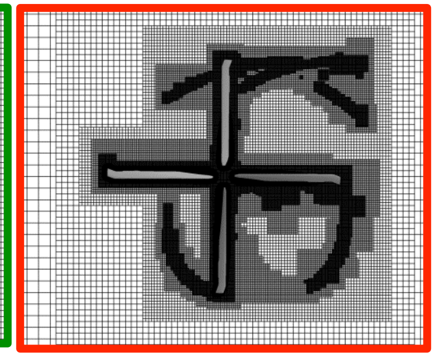
UH60 high speed forward flight



ω_{adapt} low (too many points)



Good



ω_{adapt} high (too few points)

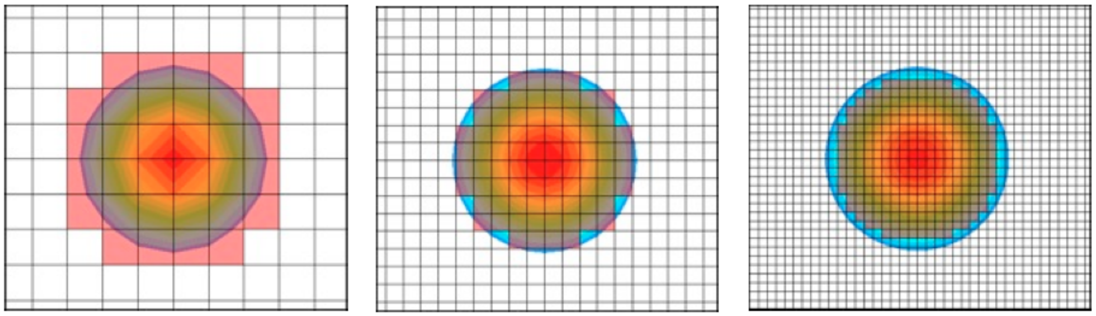
→ ***Choosing appropriate refine criteria requires considerable user-expertise and tuning***

case	C_T/σ	R	chord	Refine Criteria
UH-60 high speed	0.084	322 in	22 in	ω_{adapt}= 0.0045
V-22 hover	0.14	228 in	22 in	ω_{adapt}= 0.005
HART BVI descent	0.06	79 in	4.8 in	Q_{adapt}=0.0005

Manually determined through trial and error…

Non-dimensional Feature Detection Algorithm

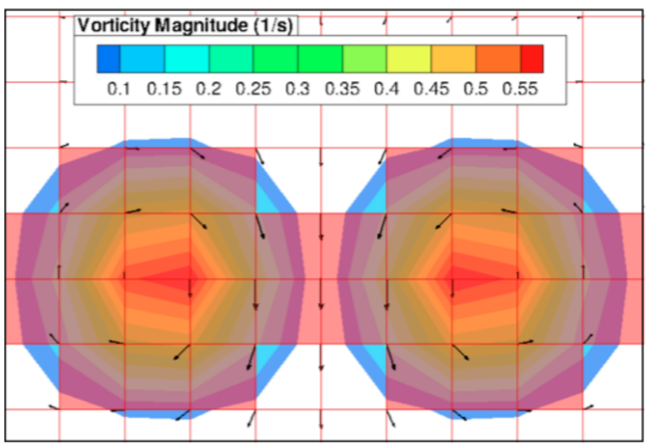
- **Define *f* as Q-criteria normalized by shear strain**
- **Refine where f > 1, otherwise don't refine**



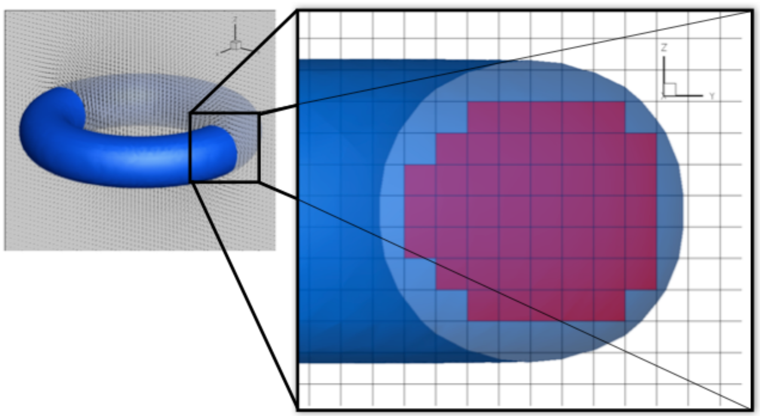
$$Q = \frac{1}{2}(||\Omega||^2 - ||S||^2), \quad f = Q$$

$$\boxed{f = \frac{1}{2}\left(\frac{||\Omega||^2}{||S||^2} - 1\right)}$$

Co- and Counter-Rotating



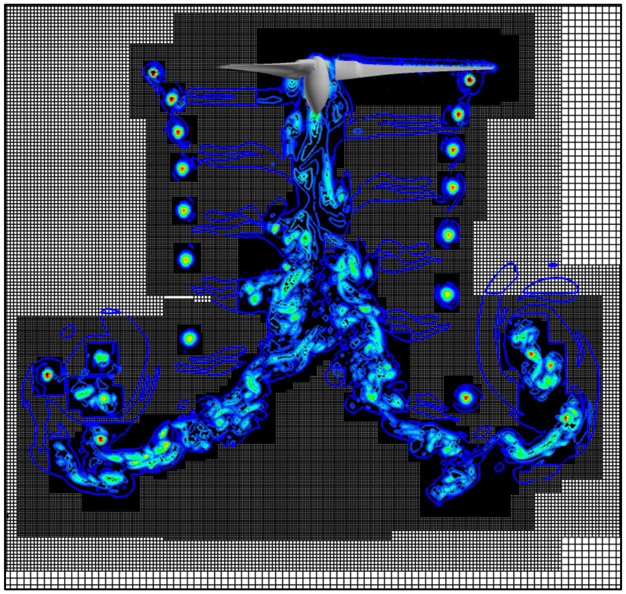
Three-Dimensional Ring Torus

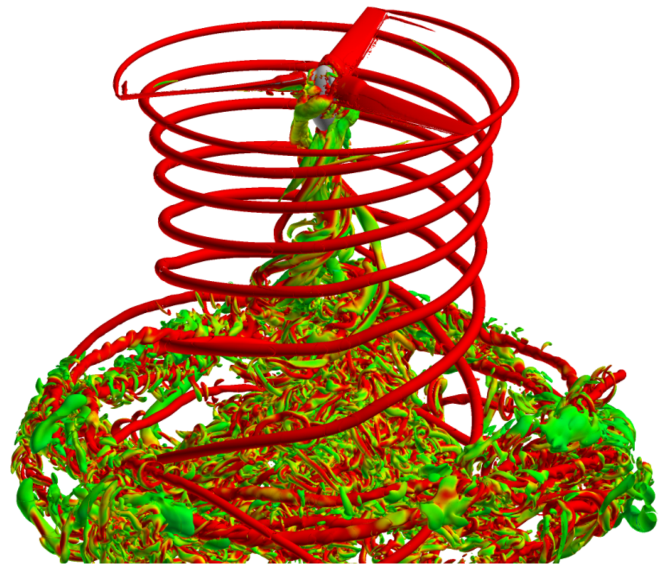


Kamkar, S. J., A.M. Wissink, V. Sankaran, A. Jameson, "Feature-Driven Cartesian Adaptive Mesh Refinement for Vortex-Dominated Flows," *Journal of Computational Physics*, Vol. 230, No. 16, July 2011, pp. 6271-6298.

Automated Off-Body AMR

TRAM Rotor





M_{tip}=0.625
θ=14^o coll

feature detection with *f* = 1 used for off-body refinement

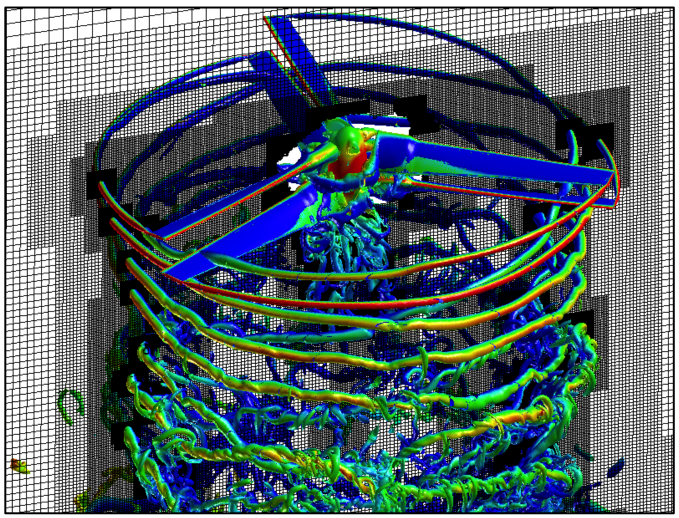
	Figure of Merit	Difference	Mesh Points
Experiment	0.78	-	--
Computation	0.773	-0.9% (+/- 0.2%)	86 M

- **Finest mesh resolution applied to all regions of swirling flow**
- **Richardson-error refinement cutoff also available**

Helios Multi-Rotor Support

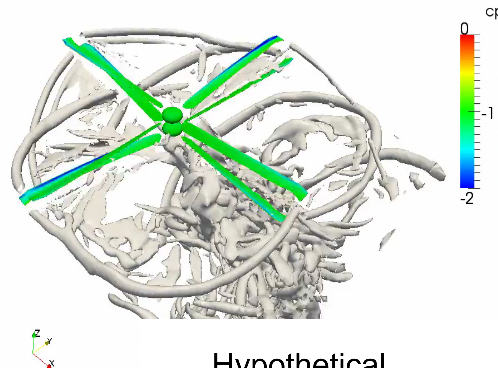
- **Current and future rotorcraft utilize multiple rotors**
- **Version 3 fully supports multi-rotor vehicle configurations**
  - Multiple motion files (specified control angles)
  - CSD coupling with comprehensive analysis codes

Hover



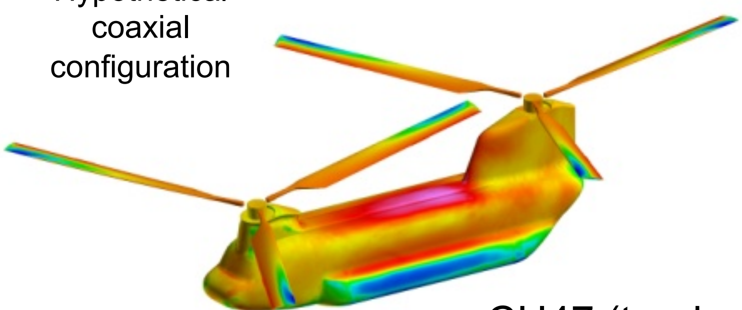
Hypothetical coaxial TRAM

Forward flight



Hypothetical coaxial configuration

Sikorsky X2

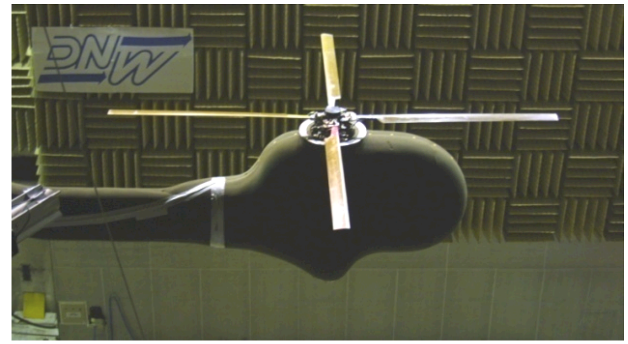


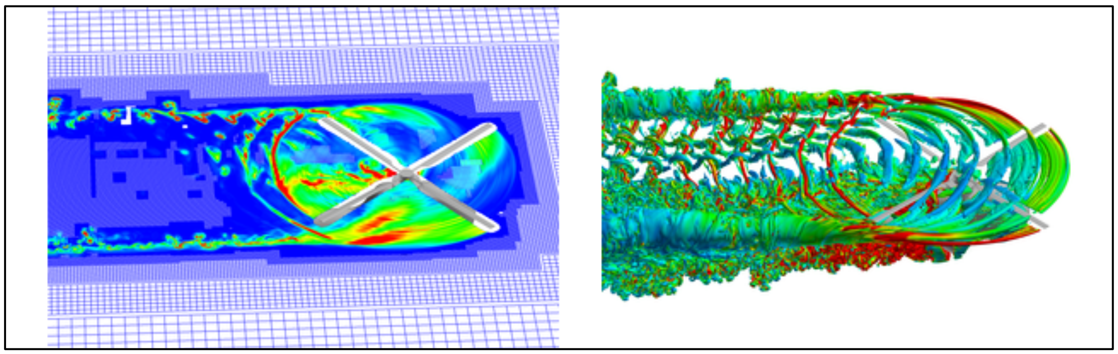
CH47 (tandem)

What Can We Do?

An Assessment of Current Capabilities

- **Helios v3 applied to HART II case**
  - 40% scaled Bo105 model rotor & “fuselage” tested in DNW windtunnel
  - Wake-based PIV measurements of vortex locations and strength

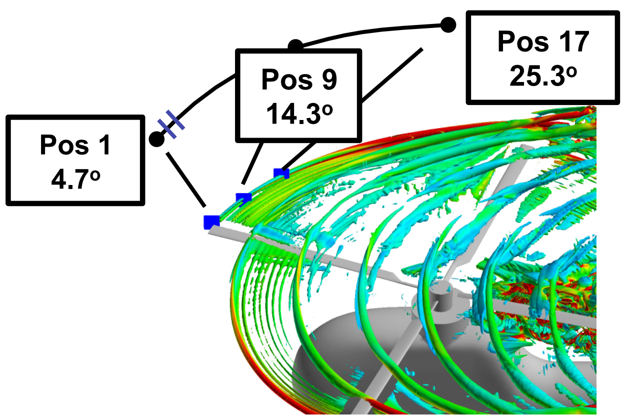




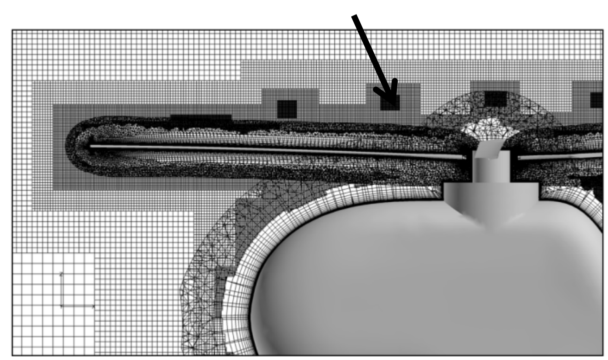
Helios simulation

Near-body: **3M nodes**
Off-body: **200M-300M nodes**
Requires ~4 days on 256 procs

Compare PIV-measured vorticity to computation



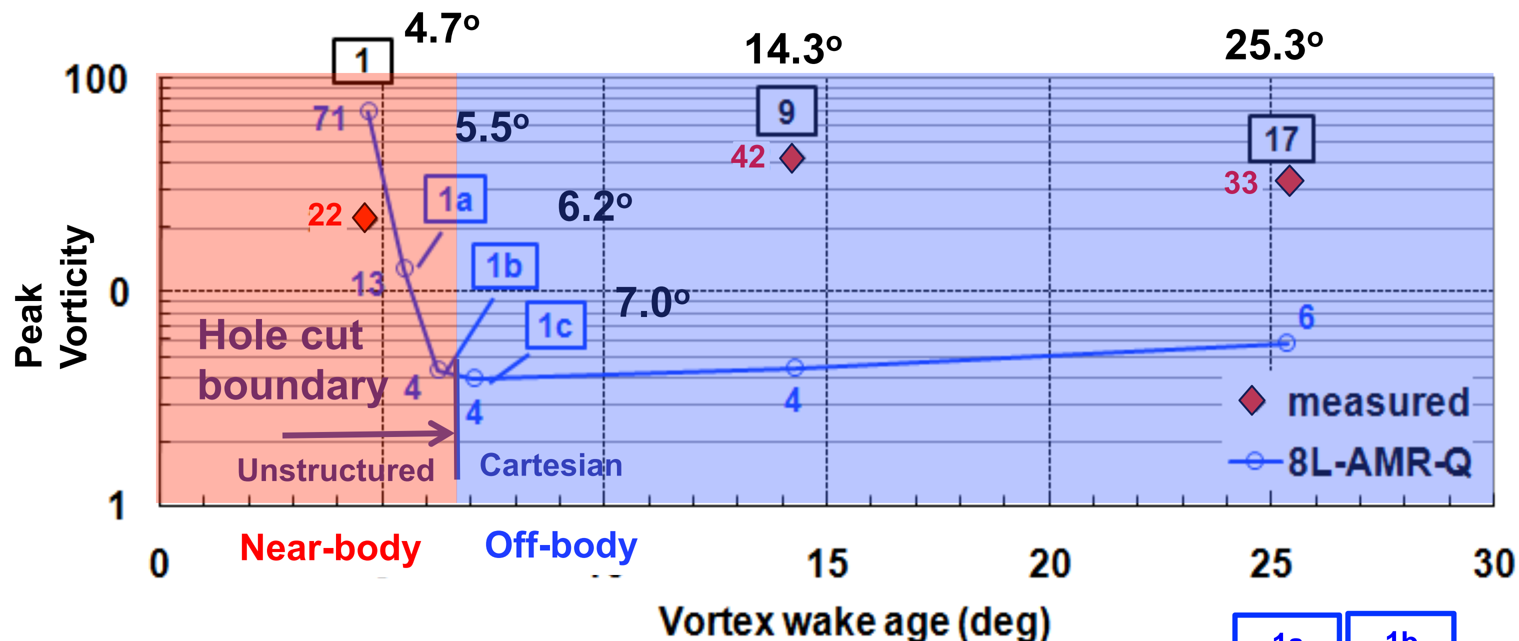
0.025c spacing



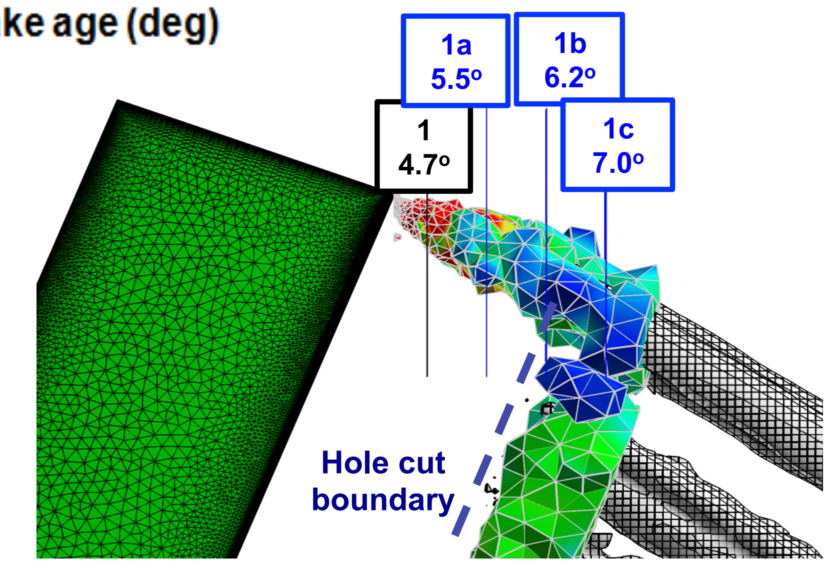
Highly-resolved Off-body Grid

Tip Vortex Dissipation

Near-body to Off-body



- Vorticity dissipates most quickly in near-body mesh
- Strength preserved in off-body mesh



Presentation Outline

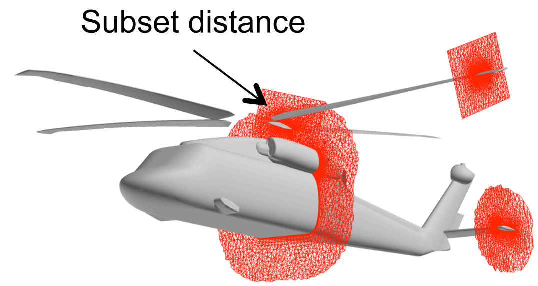
- Overview
- New capabilities
- Sample Results
- **Strand Solver Developments**
- Summary

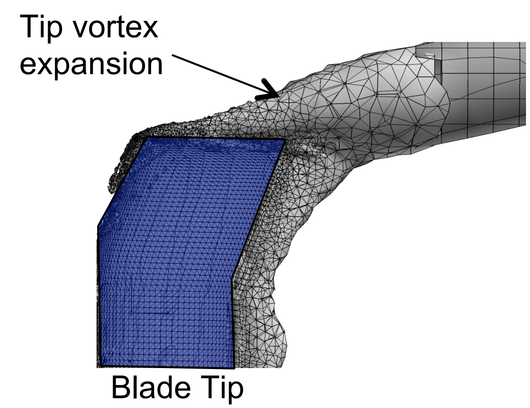
Helios

Version 3 *Rainier*

Current Bottlenecks

- **Near-body unstructured grid generation**
  - Not straightforward for the typical design engineer
  - Fixed near-body subset distance
  - Not adaptive



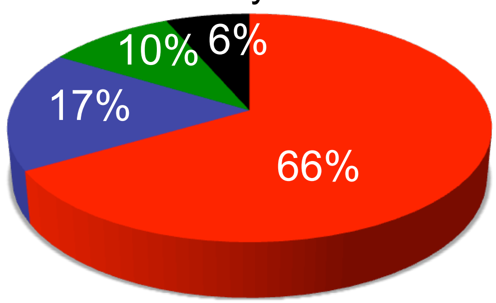


- **Lower order near-body solver**
  - unstructured solver limited to 2nd-O (off-body Cartesian is 5th-O)

- **Computational Cost**

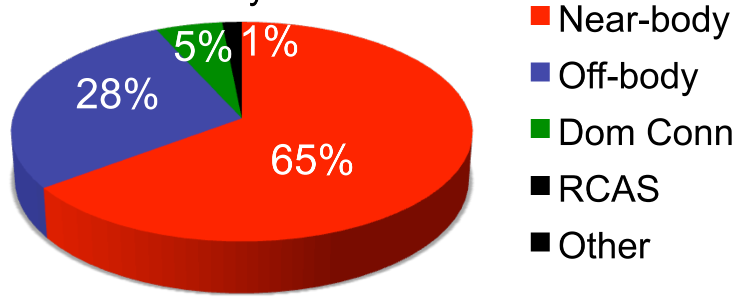
Gridpoints

Near-body: 9.4M
Off-body: 34.1M



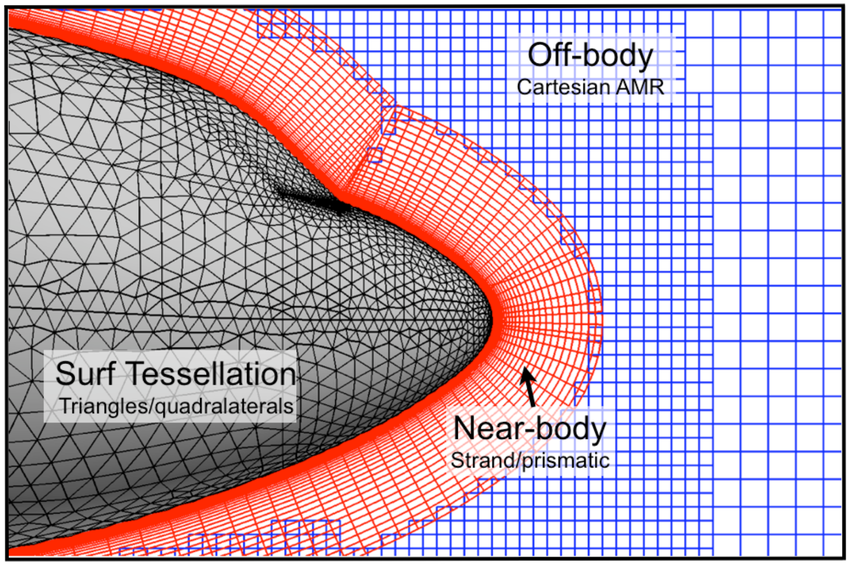
TRAM Hover - **128 procs**
137 hrs (**8 revs/6 days**)

Near-body: 15.4M
Off-body: 146M

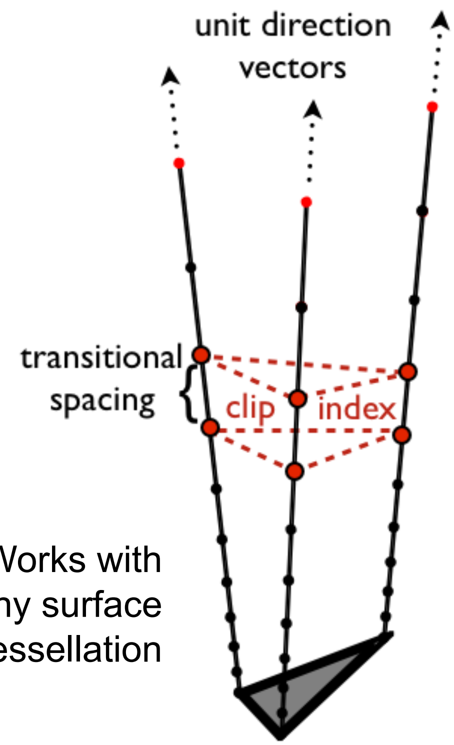


UH60 Fwd Flight – **512 procs**
28.5 hrs (**1 rev/1.2 days**)

Strand/Cartesian Approach



- **"Strand" near-body grid**
  - Straight line segments grown directly from surface tessellation
  - Transitions from viscous spacing at surface to Cartesian off-body
  - **Automatic** viscous mesh generation



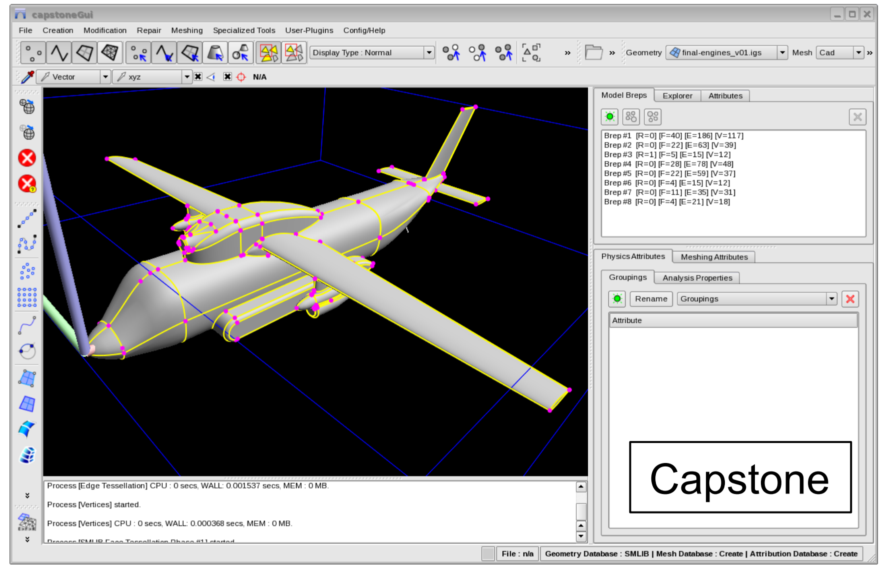
- **Concept introduced and studied in earlier works**
  - Introduced by Meakin et al, AIAA-2007-3834, 2007
  - Validated using Helios, Wissink et. al, AIAA-2009-3972, 2009
  - Specialized strand solver, Katz et al, *J. Comp. Phys.*, 2011
- **Recent work focuses on combining these developments into a usable 3D code**

Mesh Generation
Directly from CAD

Step 1:

Generate Solid Geometry

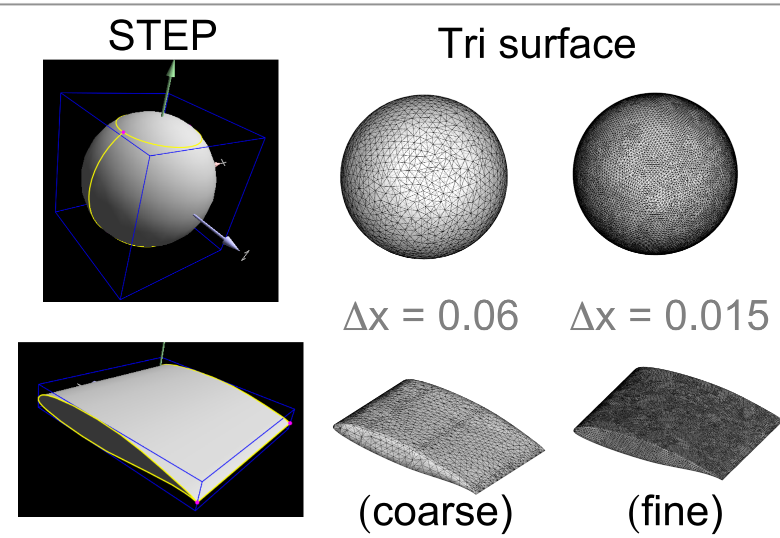
- Available in modern CAD packages e.g. STEP files
- CREATE products that produce STEP include DaVinci, Capstone



Step 2: (automatic)

Tessellate Surface

- Supply desired resolution: **Δx**
- Parameters control resolution in high curvature regions, small protrusions

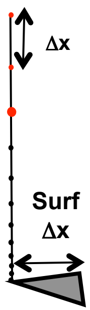


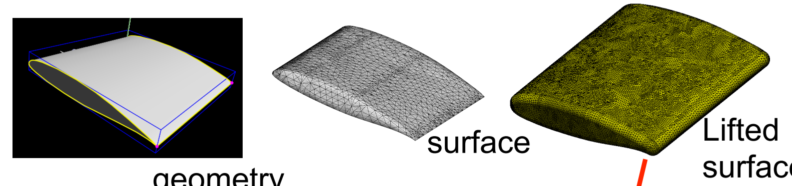
Mesh Generation
Directly from CAD (cont)

Step 3: **(automatic)**

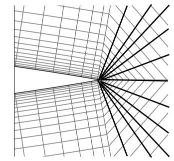
Near-body Strand Generation

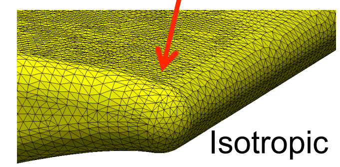
- Surface Δx determines strand length
- Construct "lifted surface" with isotropic spacing
- Lifted surface defines multiple strands at convex edges/nodes
- Apply smoothing at concave corners



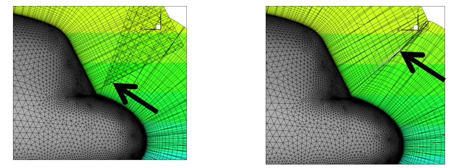


Multiple strands
at **convex** corners
- R. Haimes





Smoothing at
concave corners
- W. Chan

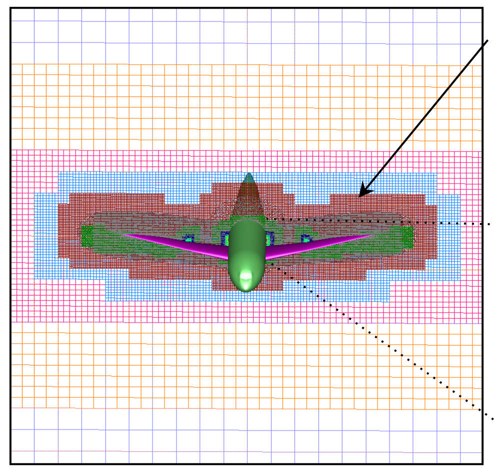


Push collapsed strands away from surf

Step 4: **(automatic)**

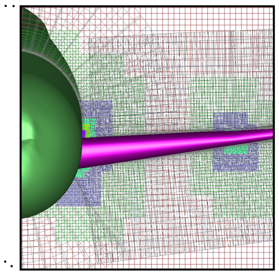
Off-body Cartesian Generation

- Refine to lifted surface
- Telescope to resolve clipped strands



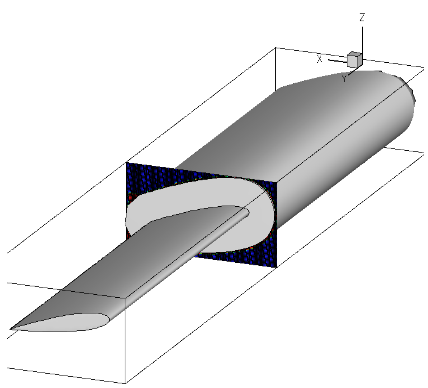
Initially refine to
lifted surface

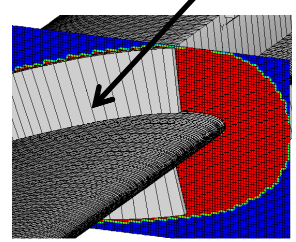
locally refine to resolve
clipped strands

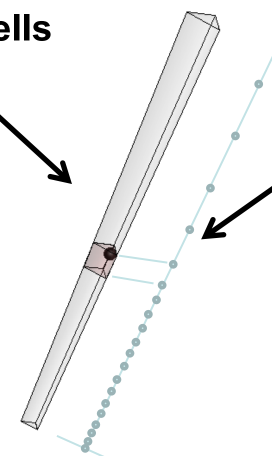


Domain Connectivity

- **OSCAR: Automated Implicit hole cutting**
  - Donors identified for every point of every mesh
  - All donor searches completed locally (since entire strand mesh description available to every processor)
- **Inverse maps accelerate donor search**
  - Holds indices of strand “super cell”
  - Logarithmic inverse map for on-strand index search

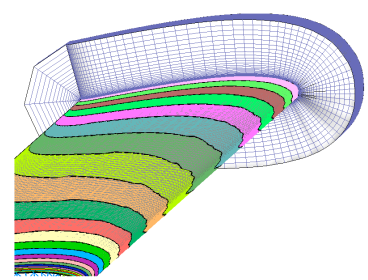






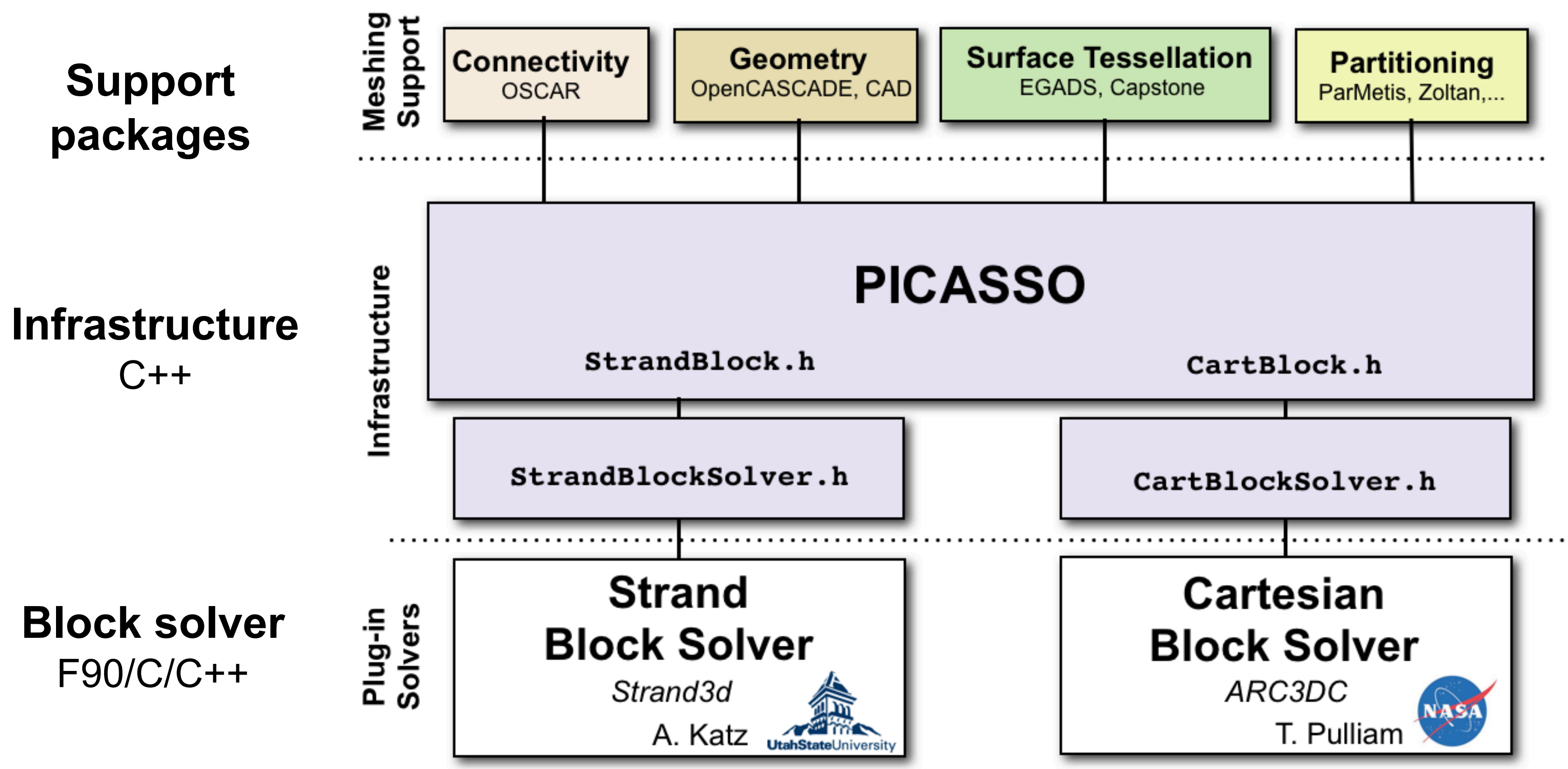
Inverse map:
red: potentially contains super cells
blue: outside strand domain (no donors)

- **Surface-based partitioning enhances scalability**



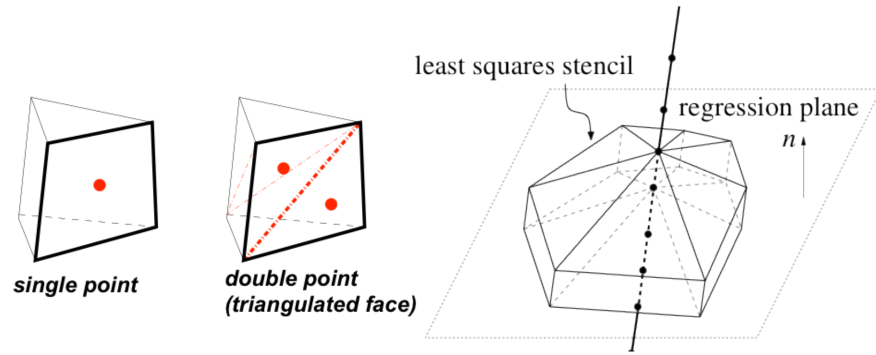
PICASSO

- **Parallel Infrastructure for Cartesian And Strand Solvers**
  - Provides strand grid generation and adaptation with interfaces to surface meshing and geometry packages
  - Facilitates easy integration of new flow solvers, extensible.

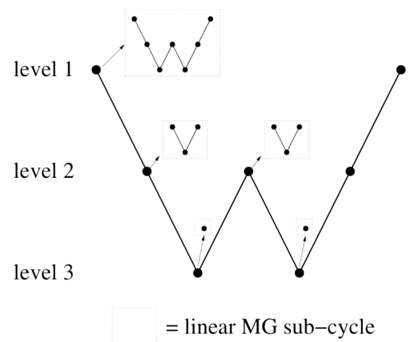


Flow Solver

- **Strand solver – A. Katz**
  - Cell-centered
  - Reynolds-averaged Navier-Stokes
  - No turbulence model (yet)
  - Accommodates both prisms & hexes
  - 2nd-Order spatial discretization
  - Implicit psuedo-time marching
  - ***Details in AIAA-2012-2779***



Green-Gauss
surface integration



Nested nonlinear multi-grid
- outer: FAS nonlinear MG
- Inner: nested linear subcycles

numerical flux:

$$\hat{\mathcal{F}} = \frac{1}{2}\left(\mathcal{F}(Q_L) + \mathcal{F}(Q_R)\right) - \mathcal{D}(Q_L, Q_R),$$

CUSP:

$$\mathcal{D}(Q_L, Q_R) = \frac{1}{2}\alpha^* c(Q_R - Q_L) + \frac{1}{2}\beta(\mathcal{F}(Q_R) - \mathcal{F}(Q_L)).$$

nodal projection gradients:

$$\frac{\partial Q}{\partial x} = \sum_i a_{xi} Q_i$$

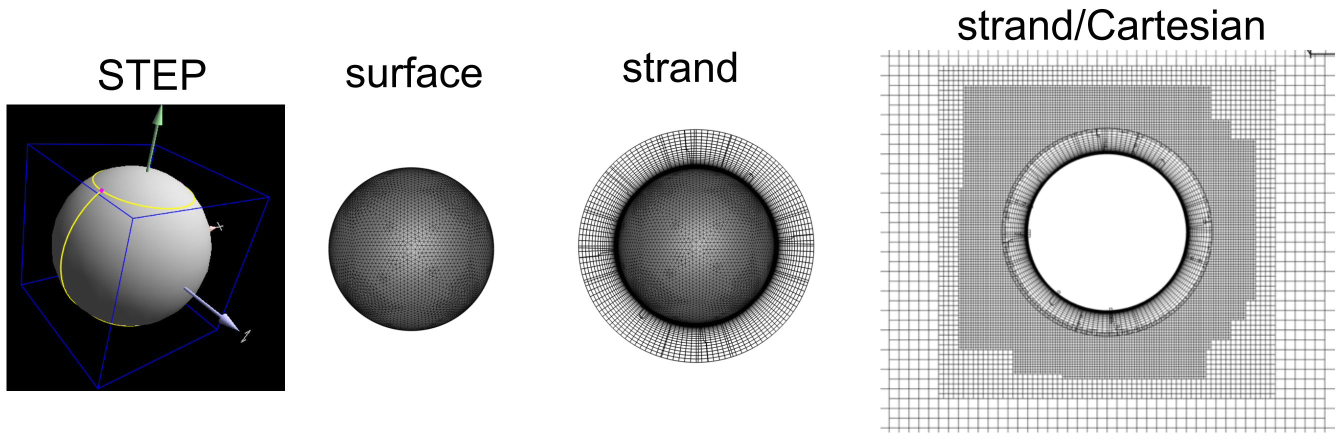
- **Cartesian Solver**
  - SAMARC solver used in Helios
  - ***Details in AIAA-2010-4554***

PICASSO Results

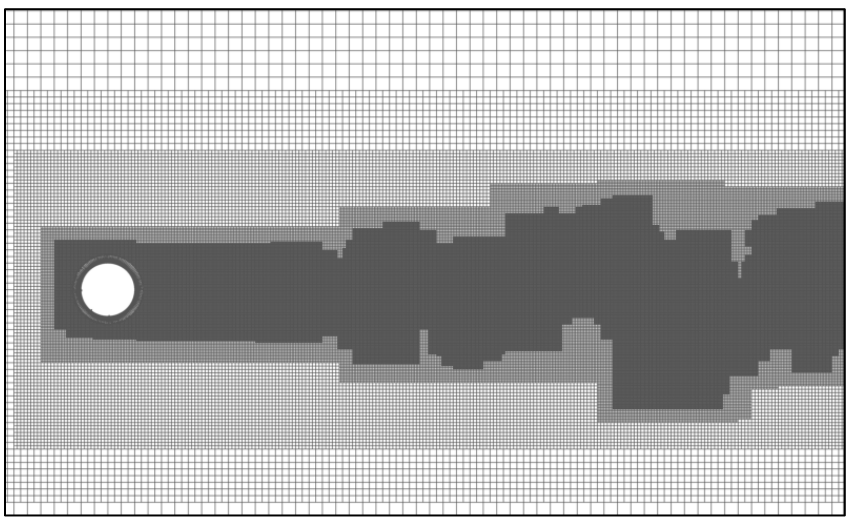
Flow Over Sphere

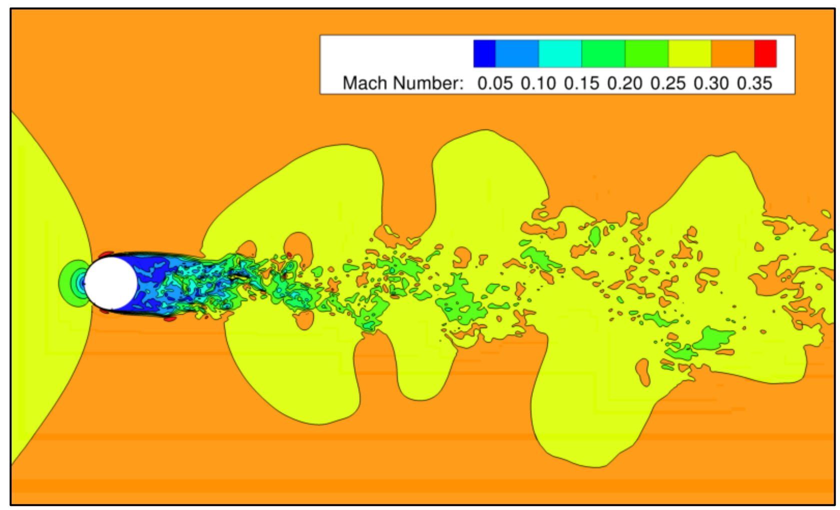
Re = 1000

Inputs:
- STEP file
- Wall spacing
- Flow conditions



Automatic





Solution after 25000 steps

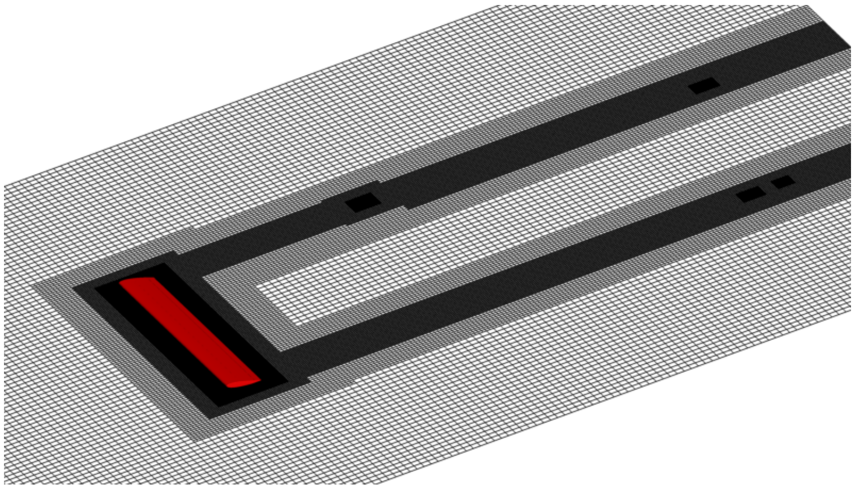
PICASSO Results

NACA 0015 Wing

Flat tip
Aspect ratio = 6.6
M=0.1235
α= 12^{o}
Re = 5000

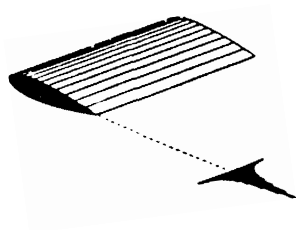
PICASSO

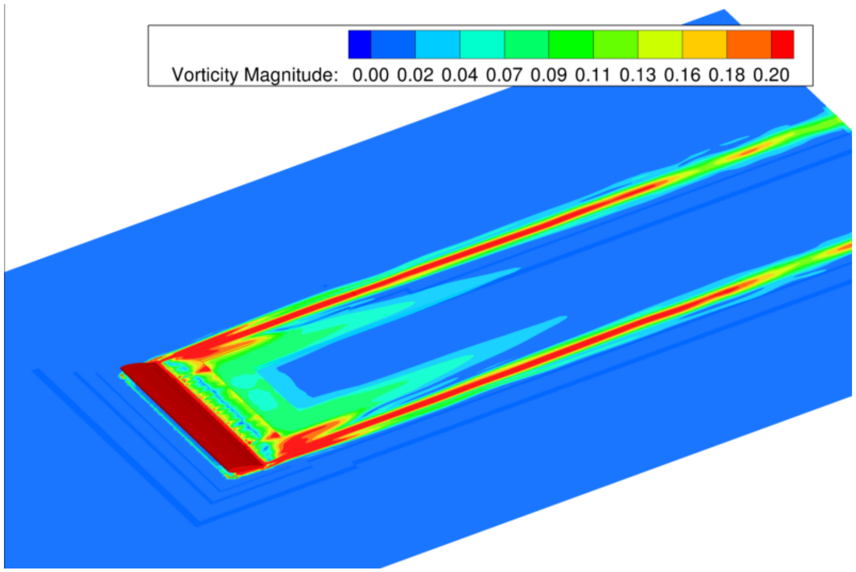
32 procs



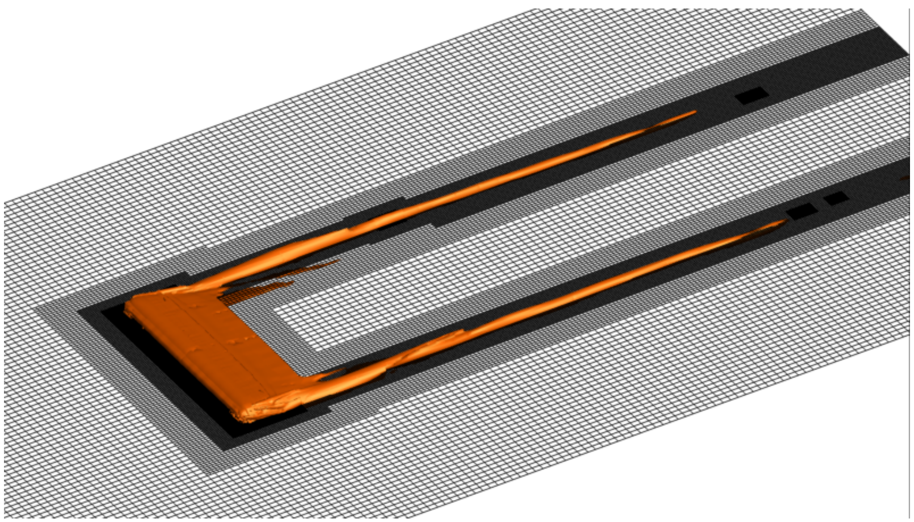
Mesh

Experimental Results
McAlister, Takahashi





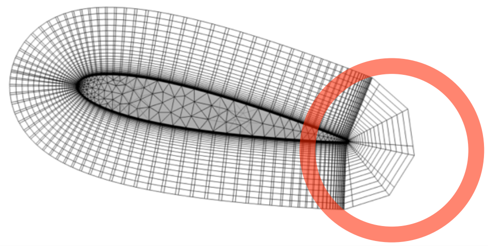
Vorticity contours



Vorticity iso-surface

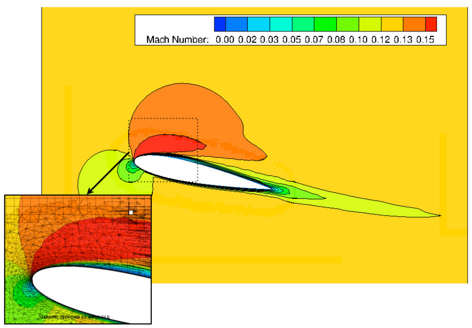
PICASSO Results
NACA 0015 Wing (cont)

Flat tip
Aspect ratio = 6.6
M=0.1235
α= 12^o
Re = 5000



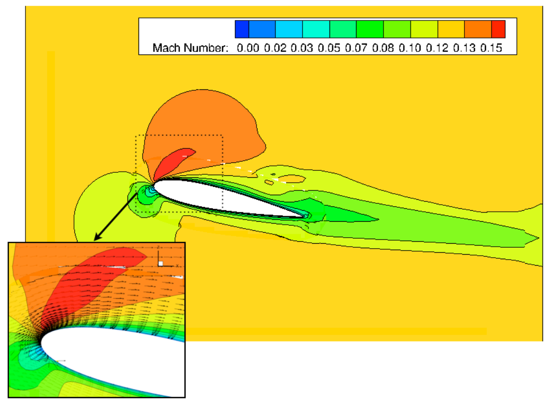
Trailing edge flow should be improved by multi-strand capability

Targeted for Jan 2013

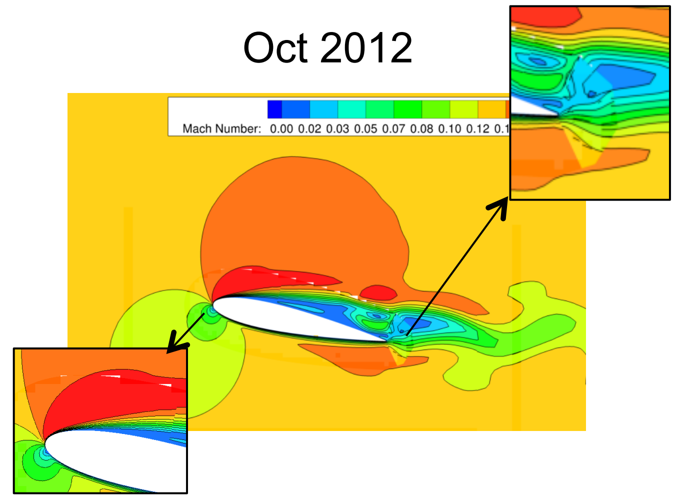


Helios Results
unstruct mesh, NSU3D

June 2012
AIAA-2012-2916



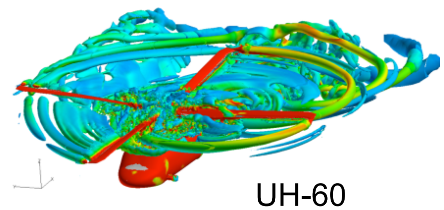
Oct 2012

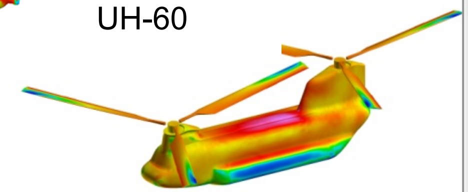


PICASSO Results
strand mesh, Strand3d

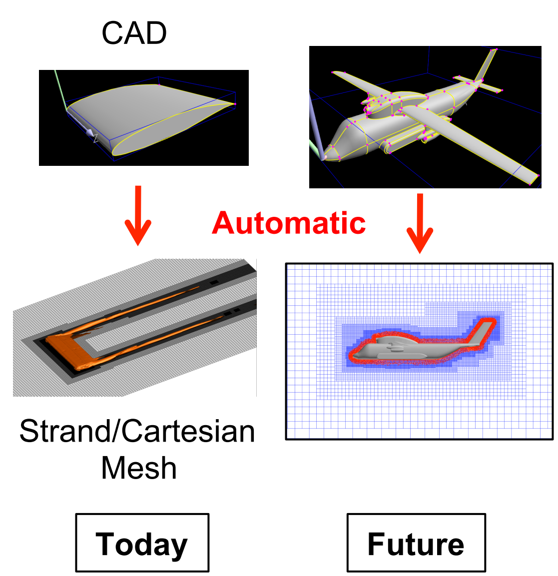
Summary

- **Helios is a high-fidelity rotorcraft overset CFD/CSD analysis tool intended for DoD acquisition engineers**
- **Current CFD bottlenecks**
  - Unstructured mesh generation non-trivial
  - Much of the wake dissipation occurs in near-body mesh
  - Near-body solver dominates computation time





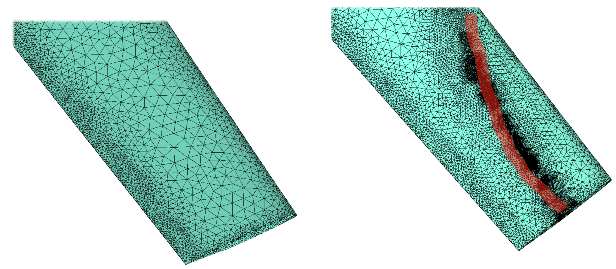
- **Strand technology a promising new approach**
  - Automatic viscous mesh generation directly from CAD solid geometry
  - Parallel flow solver that exploits structure in normal direction
  - Scalable domain connectivity
- **First dedicated strand solver implemented in PICASSO**
  - Demonstrated automatic CAD-to-mesh for simple geometries
  - Actively pursuing more complex cases



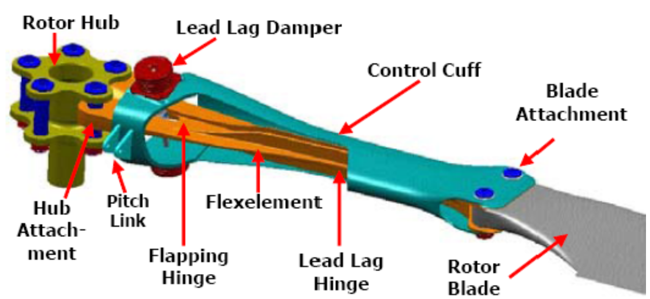
Future

- **Application of Helios v3 to candidate Joint Multi-Role acquisition designs**
- **Strand improvements**
  - Capability to handle multiple strands
  - Turbulence models
  - More complex geometries
- **Higher order algorithms**
- **Near-body Mesh Adaptation**
- **Higher fidelity structural dynamics**
  - Current CSD uses 1D beam elements
  - 3D dynamics finite element solver needed for composite blades/hubs
- **Scalability to 1000's processors**

Joint Multi-Role Concepts



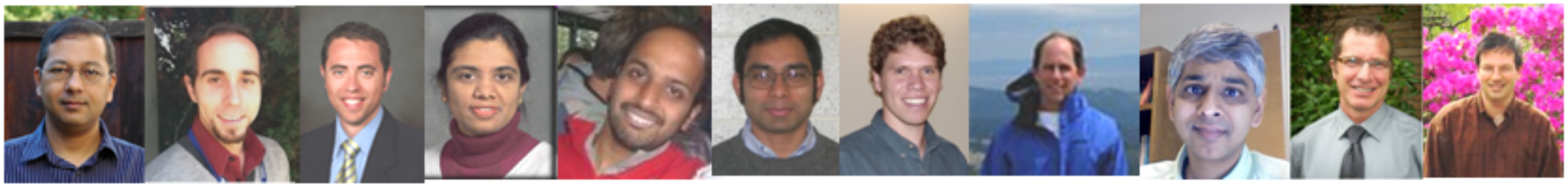
Surface Adaptation



3D structures

Acknowledgements

Helios development team



Jain Burgess Katz Jayaraman Sitaraman Datta Wissink Potsdam Sankaran Strawn Mavriplis

Supported by the CREATE-AV (air vehicles) element of the Computational Research and Engineering for Acquisition Tools and Environments (CREATE) Program sponsored by the U.S. Department of Defense HPC Modernization Program Office.