Introduction to the Glenn Icing Computational Environment (GlennICE)

Thomas Ozoroski

NASA Glenn Research Center – Icing Branch

Thermal Fluids Analysis Workshop, Cleveland, OH

8/26/2024

Agenda

- Background and Introduction
- GlennICE Workflow Overview
- ONERA M6 Example Case Walkthrough
- CRM-HL Example Case
- ARIES-I Example Case
- Future Developments
- Additional Projects and Research
- How To Obtain GlennICE

What I Hope to Convey and Showcase Today

- Provide a high-level overview of aircraft icing
- Discuss what sort of research we are conducting in the Icing Branch at NASA Glenn
- Highlight some of our previous research and showcase a few of our facilities
- Provide an overview of our new computational tool GlennICE
- Walk you through an example case using GlennICE
- Highlight some research we are actively engaged with and some examples
- Talk about where we are investing money in tool development

BACKGROUND AND INTRODUCTION

Introduction to Icing

Background

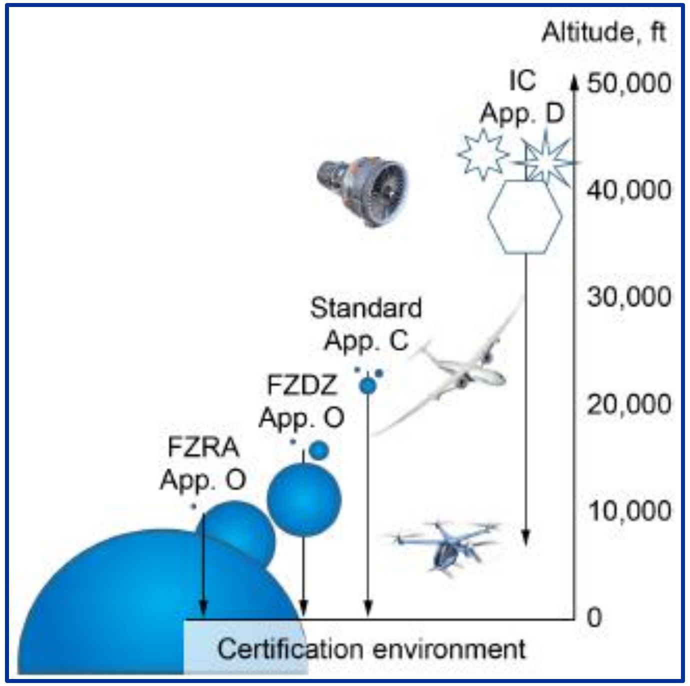
- Ice formation and its resulting performance impact is a complex problem that is interdisciplinary by nature.
- Due to this complexity, icing tools are only validated over a narrow range of vehicle geometries and icing conditions.

Goals

- Improve the state of the art of icing tools to be applicable over a larger range of conditions, including high-lift configurations, novel vehicle designs, Freezing Drizzle (FZDZ), Freezing Rain (FZRA), and Ice Crystals (IC).

Impact

- Enable icing tools to be utilized earlier in the design process to reduce design cost, as well as increasing safety and efficiency.
- Note that novel vehicle and engine designs have an unknown risk to the newly codified Appendix D and O icing conditions (FZRA, FZDZ and IC), and carry a regulatory burden that existing tube and wing designs with a proven flight history do not carry.
- Absent of advancements in validated simulation techniques for icing, the costs associated with proving the safety of novel vehicle and engine designs can be extremely large or even economically prohibitive for market entry.

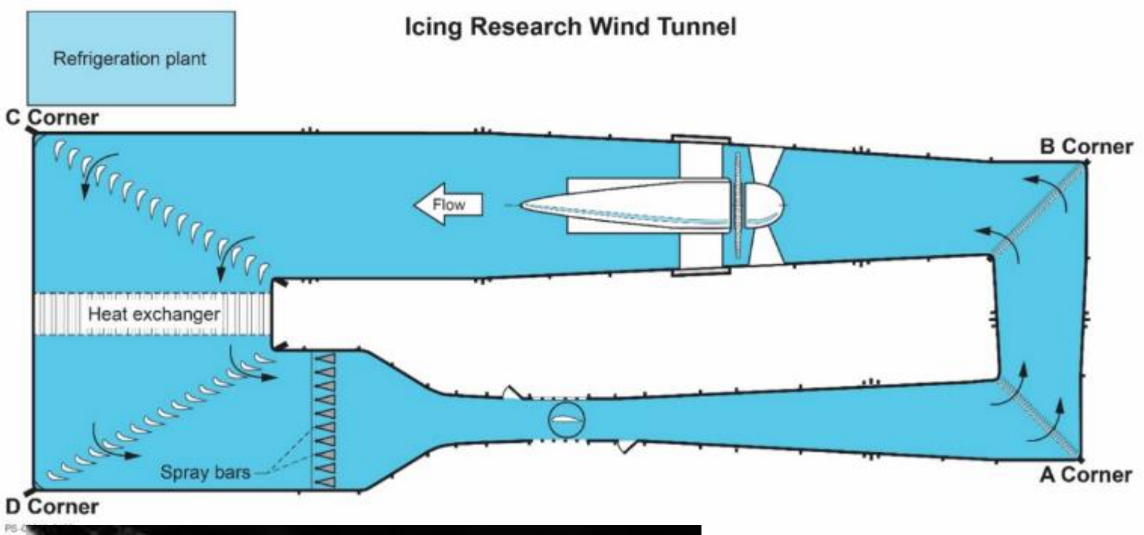


The various icing certification environments and the altitude at which they occur.

[1]See Appendix B *History of Icing Certification Rules* of AC 25-28 for more detail.

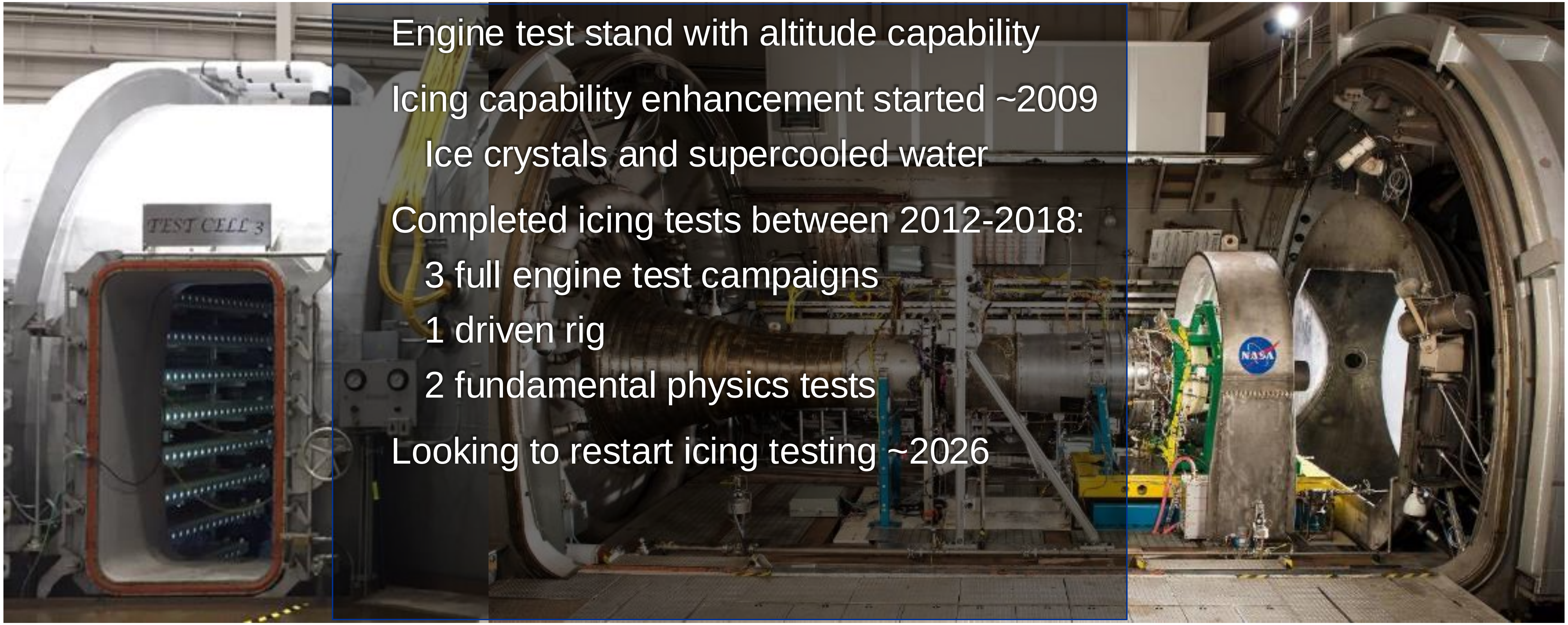
Icing Research Tunnel (IRT)

- Recirculating Icing Wind Tunnel
- 6 x 9 ft (1.8 x 2.7 m) test section
- Constructed in 1943
- Underwent numerous upgrades in its history
- Heavily utilized by NASA, industry, DoD, …
- Designed for super cooled water icing
- Now some limited ice crystal capability



https://www.nasa.gov/directorates/armd/aetc/icing-research-tunnel/

PSL: Propulsion Systems Lab



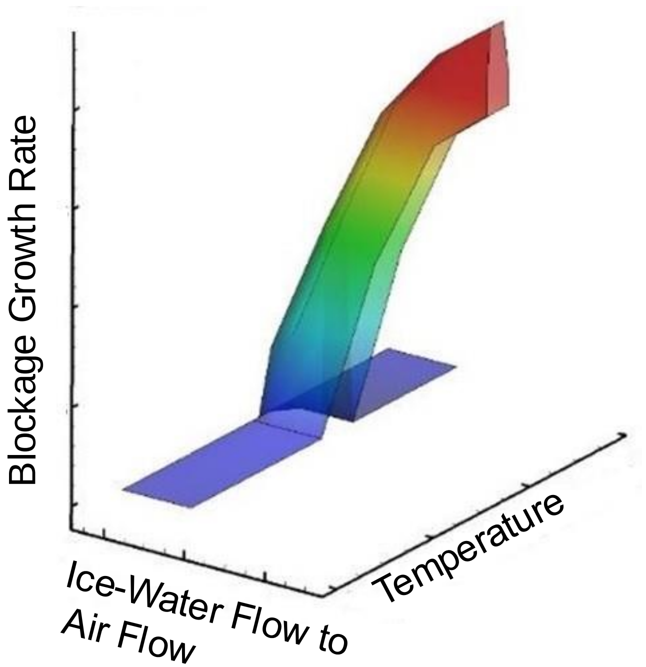
- Engine test stand with altitude capability
- Icing capability enhancement started ~2009
  - Ice crystals and supercooled water
- Completed icing tests between 2012-2018:
  - 3 full engine test campaigns
  - 1 driven rig
  - 2 fundamental physics tests
- Looking to restart icing testing ~2026

https://www.nasa.gov/directorates/armd/aetc/propulsion-systems-laboratory-psl-facility/

NASA Ice Crystal Icing Tools

COMDES-MELT

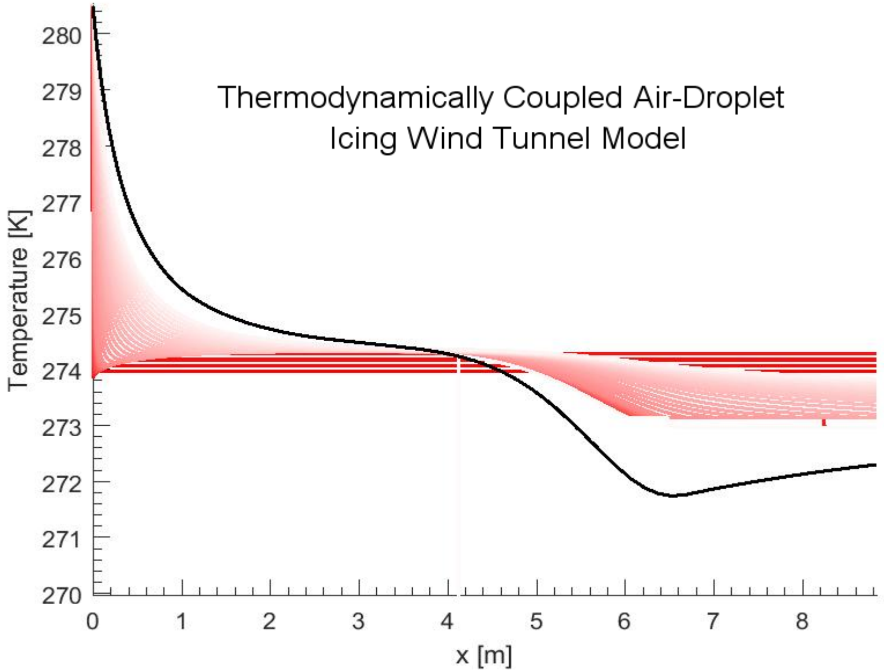
Mean-line compressor analysis code coupled with an ice crystal thermodynamic state code, which is used as a turbofan engine icing risk analysis tool



https://software.nasa.gov/software/LEW-20027-1

TADICE

Icing wind tunnel simulation tool (1D) that models the thermodynamic interactions between the water/ice particles of an icing cloud and air as it flows down the tunnel.

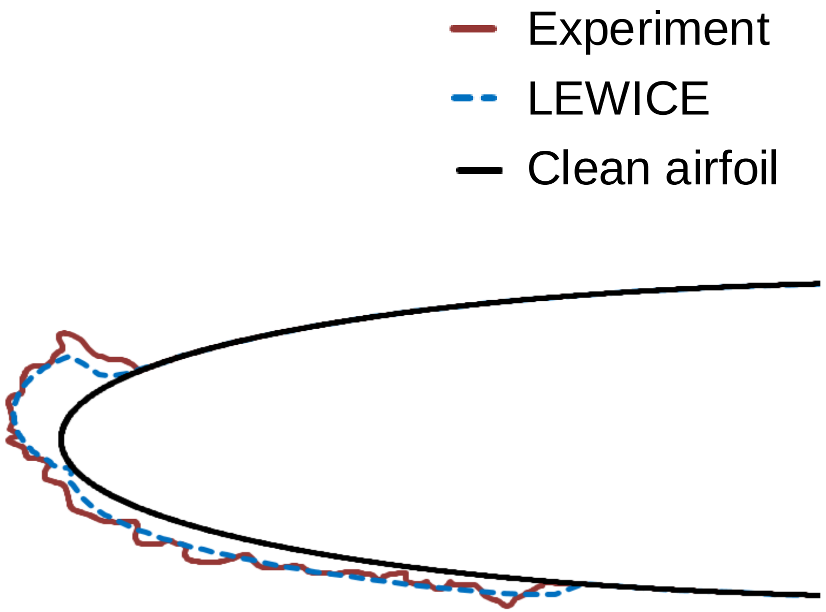


https://software.nasa.gov/software/LEW-19874-1

NASA Icing Tools

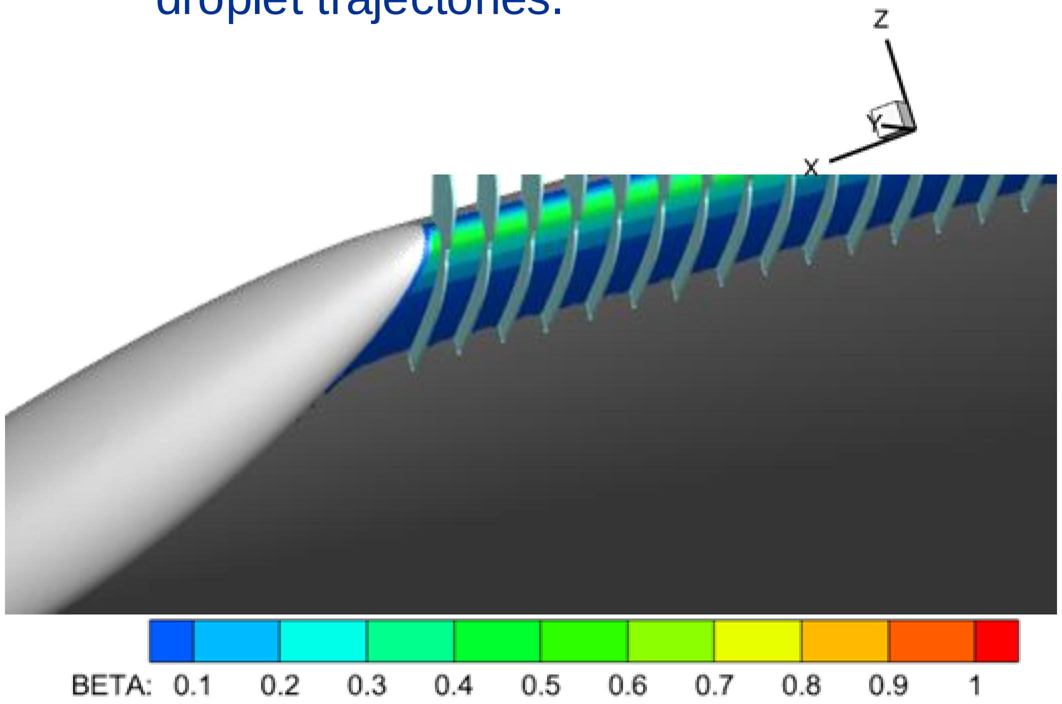
LEWICE (1980s – 2006)

Simulates 2D ice accretion using super-cooled droplets impinging on a body.



LEWICE3D (1993 – 2015)

Simulates ice accretion in quasi-3D on user-specified cut planes using full 3D droplet trajectories.



https://www1.grc.nasa.gov/aeronautics/icing/software/

Why We Are Developing GlennICE

- Historically, icing codes have relied on integral boundary layer methods and stepping back from the stagnation/attachment line
- Our historical codes are not fully 3D
  - LEWICE is 2D
  - LEWICE3D is a quasi-3D code
- LEWICE3D uses a CFD flow solution and computes ice growth along selected cut planes
  - This procedure typically involves lofting 2D ice shapes into a 3D ice shape
- LEWICE3D relied heavily on empirical data and was tuned for transport aircraft
  - The type of problems encountered in new and novel designs really pushes LEWICE3D
- The need was identified for NASA to develop a new fully 3D icing solver
- GlennICE was designed from the ground up using
  - Modern coding practices
  - Designed to accrete ice in a fully 3D workflow
  - Removing empirical reliance to focus on modeling underlying physics from first-principle

GlennICE Overview

- Fully 3D icing simulation tool
- CFD post processor
- Can predict droplet impingement & resulting ice growth on aircraft surfaces
- Lagrangian droplet tracking with adaptive refinement for efficient solution convergence
- Highly efficient parallelization schemes built for modern HPCs
- Built using modern coding practices and standards from the ground up
- Designed and focused on tackling problems relevant to government, industry, and academia

GlennICE

External Icing

Rotational Icing

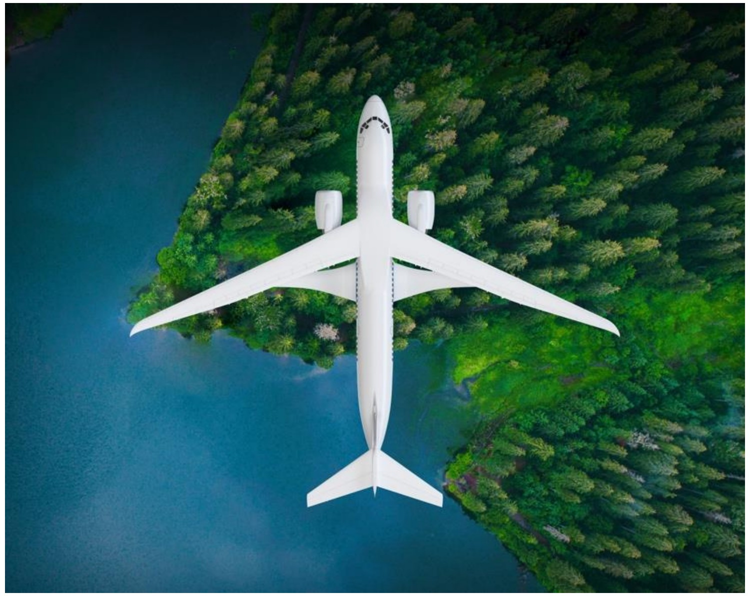
Engine Icing

As of 2018,GlennICE is the foundational code through which NASA will develop and evaluate physical models associated with ice accretion

The Problems We Are Tackling

With an icing code capable of accreting ice on an arbitrary 3D configuration, we can analyze and improve configurations earlier in the design phase

Enduring Safety

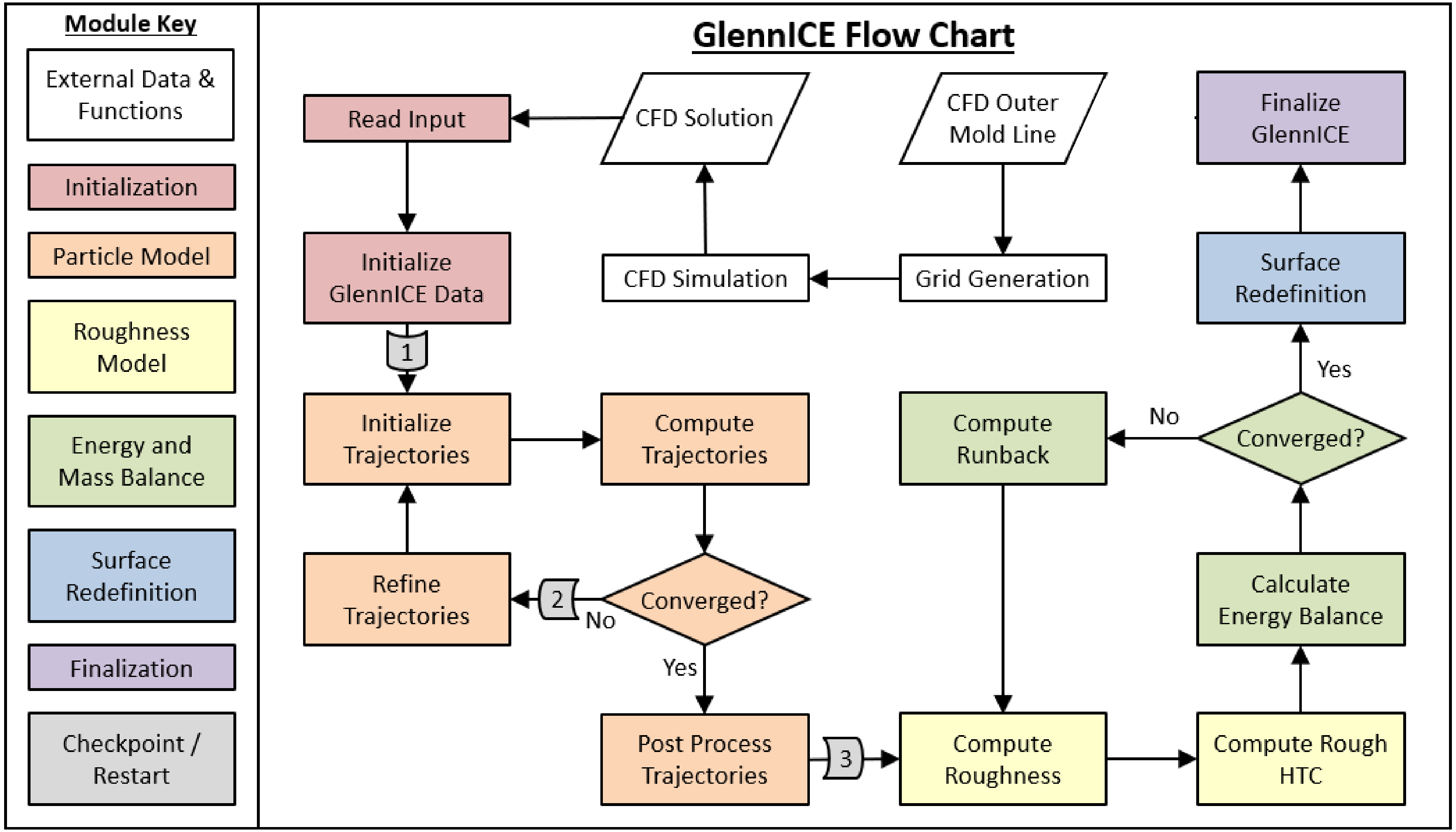


Fuel Efficient Concepts

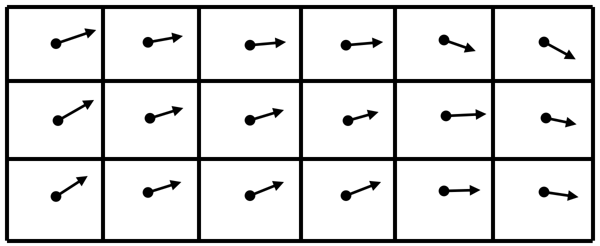
Emerging Markets

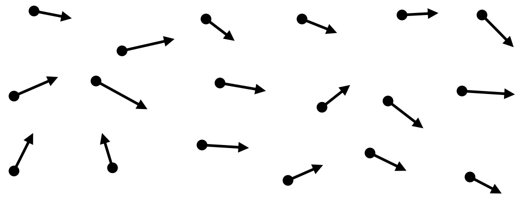
GlennICE WORKFLOW OVERVIEW

Standard GlennICE Workflow



Impingement Computation: Lagrangian vs Eulerian





Eulerian

Pros

- *Domain decomposition.*
- *Ability to couple with fluid flow.*

Cons

- *Susceptible to numerical diffusion*
- *Cloud is viewed as a continuum… i.e. unable to physically capture tangled trajectories*

Lagrangian

Pros

- *Accuracy of solution is decoupled from underlying CFD mesh.*
- *Ease of use as a post processing tool for arbitrary CFD solutions.*

Cons

- *Problem is akin to those seen in ray tracing.*
- *Difficult to decompose the domain for parallel problems.*
- *Many trajectories are required to compute smooth collection for naïve schemes.*

Lagrangian Approach: Node Centered Streamtube Definition

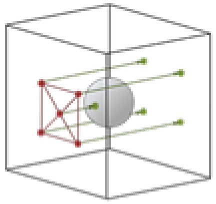
- GlennICE discretizes the streamtube area on the release plane and associates a scalar quantity of water flow rate with each individual trajectory

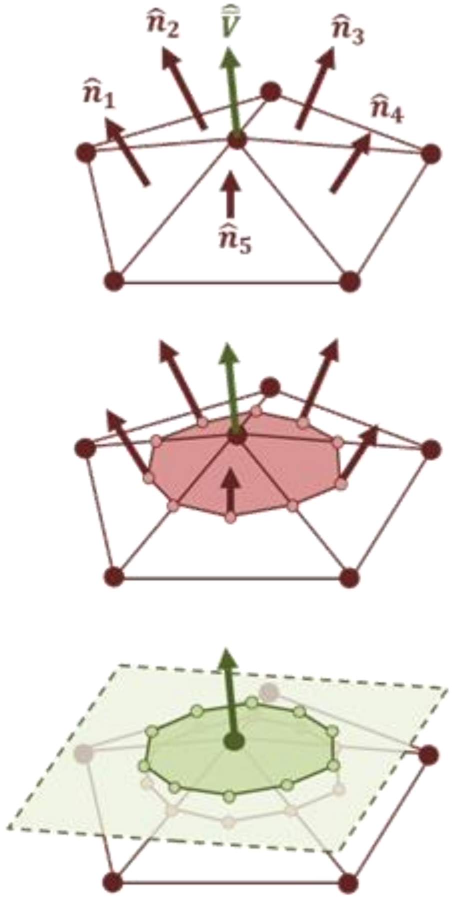
- Allows for trajectory refinement on curved surfaces

- The streamtube area is defined by the equation:

$$Streamtube\ Area = \sum_{i=1}^{\#\ of\ faces} W_i A_i \left(\hat{\hat{V}} \cdot \hat{n}_i \right)$$

- From streamtube area, a water flow rate is assigned to the trajectory

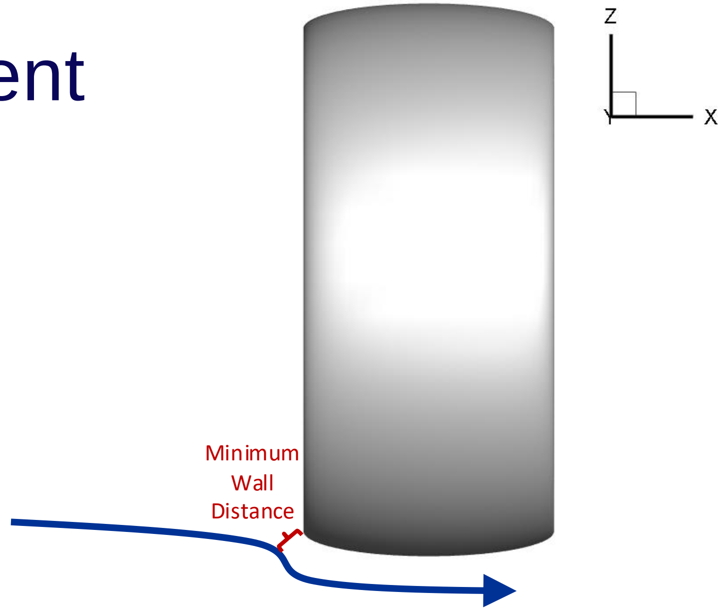


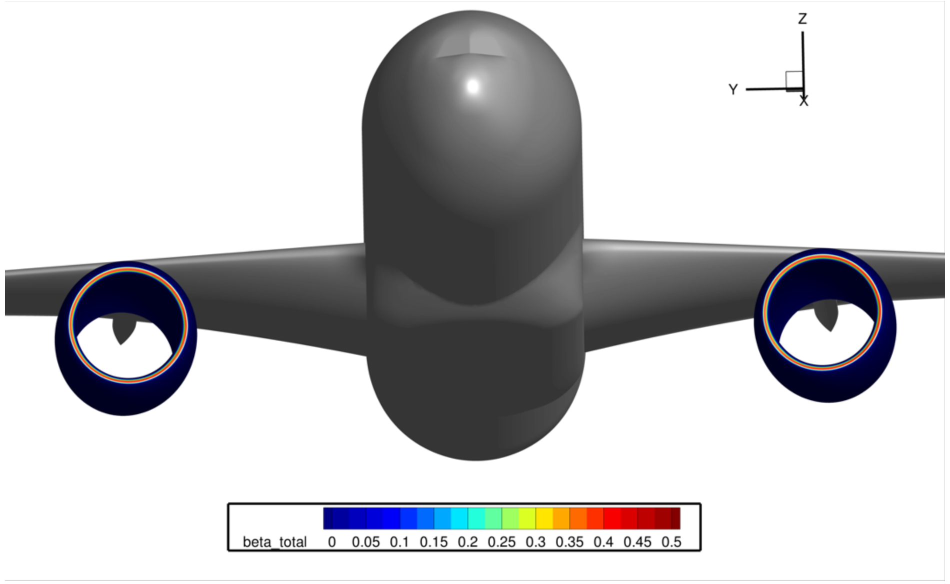


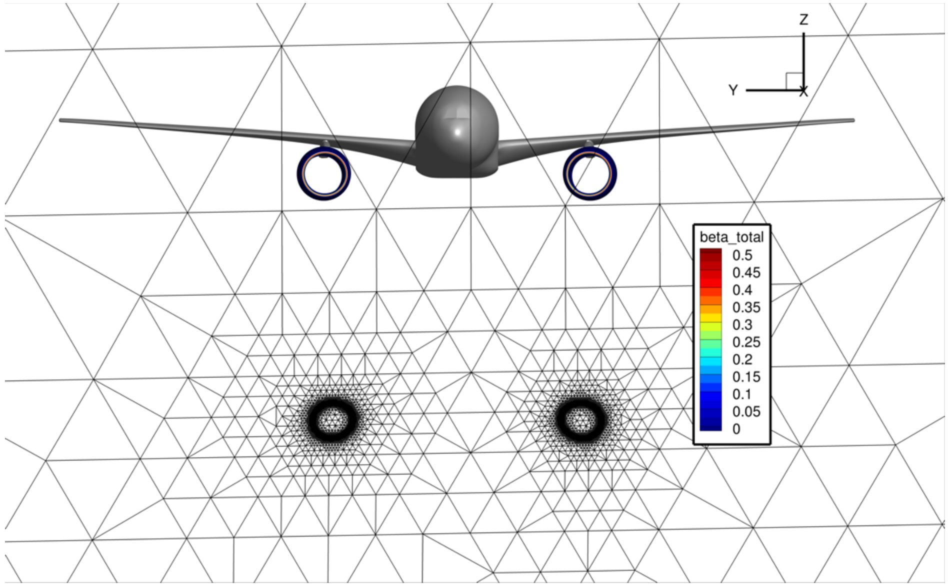
Schematic depicting the computation of stream tube associated to a trajectory on a non-planar inlet

Feature Finding and Automatic Refinement

- GlennICE currently computes wall distance internally based on the Fast Distance method of Wigton [1].
- Users only have to specify the CFD surface they are interested in targeting, and GlennICE will adaptively refine the trajectory release to target these surfaces.
- Adaptively releasing trajectories allows for immense time savings when trying to converge a solution



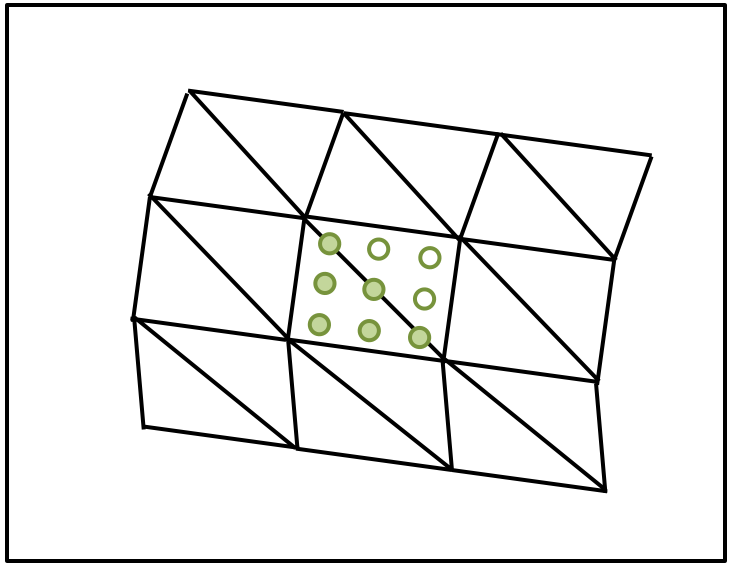




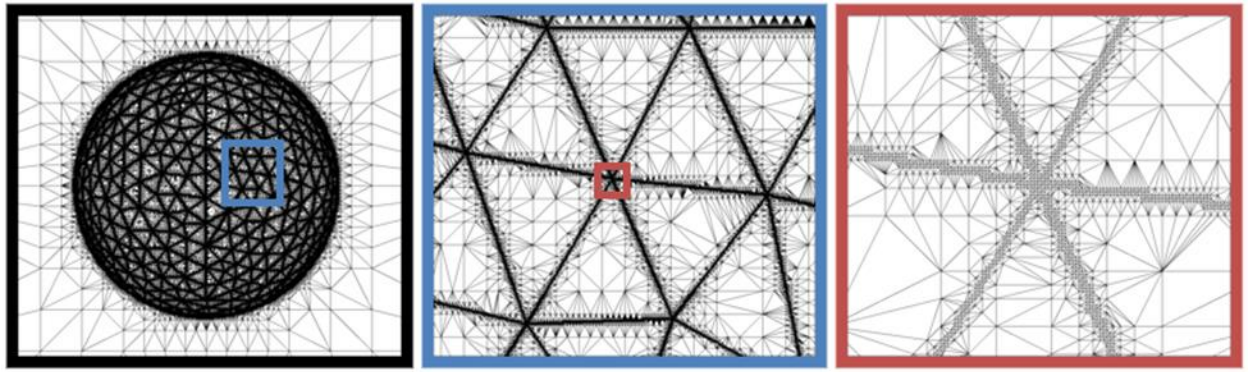
[1] Wigton, Larry "Final Report for first year of prime Contract NAS2-14090 "Research in Computational Aeroscience Applications Implemented on Advance Parallel Computing Systems" 1996

Lagragian Approach - Impinging Water

- A fully generic approach is costly.
- Metrics can be created to identify regions where continuum assumptions are likely valid.
  - Only need to refine trajectories that hit near the edge of a CFD face
- Using these assumptions, unnecessary work can be avoided, greatly improving efficiency.[1]
- The desired result is a hybrid approach that is:
  - Robust in regions where the continuum assumption breaks down.
  - Efficient in regions where the continuum assumption is valid.



Schematic of the Impinging Water Methodology



Depiction of the trajectory and trajectory connectivity at refinement iteration 20 at various levels of zoom. Note that the location of the zoomed images are identified by the colored boxes.

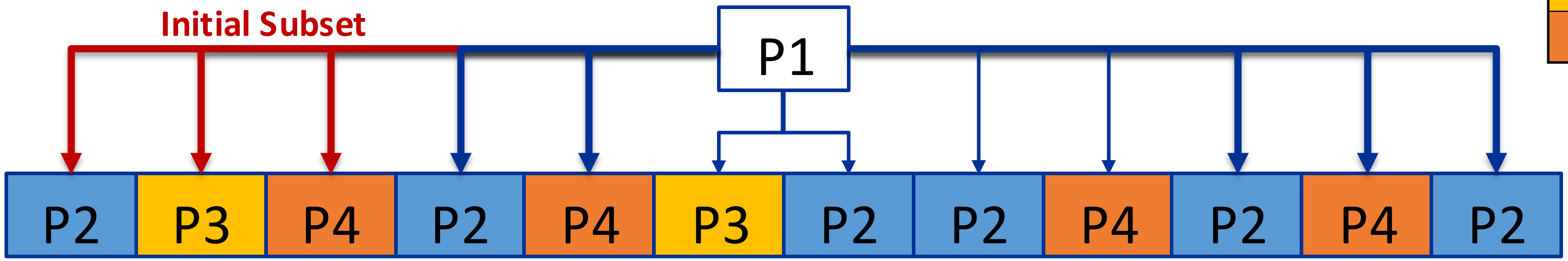
[1]Porter, Christopher E. "A Comparison of Trajectory Refinement Schemes for GlennICE." AIAA AVIATION 2022 Forum. 2022.

Static vs. Dynamic Scheduling

- In a **static scheduling** process, the trajectory routines are ordered to distribute an equal amount of trajectories between the total number of processors.

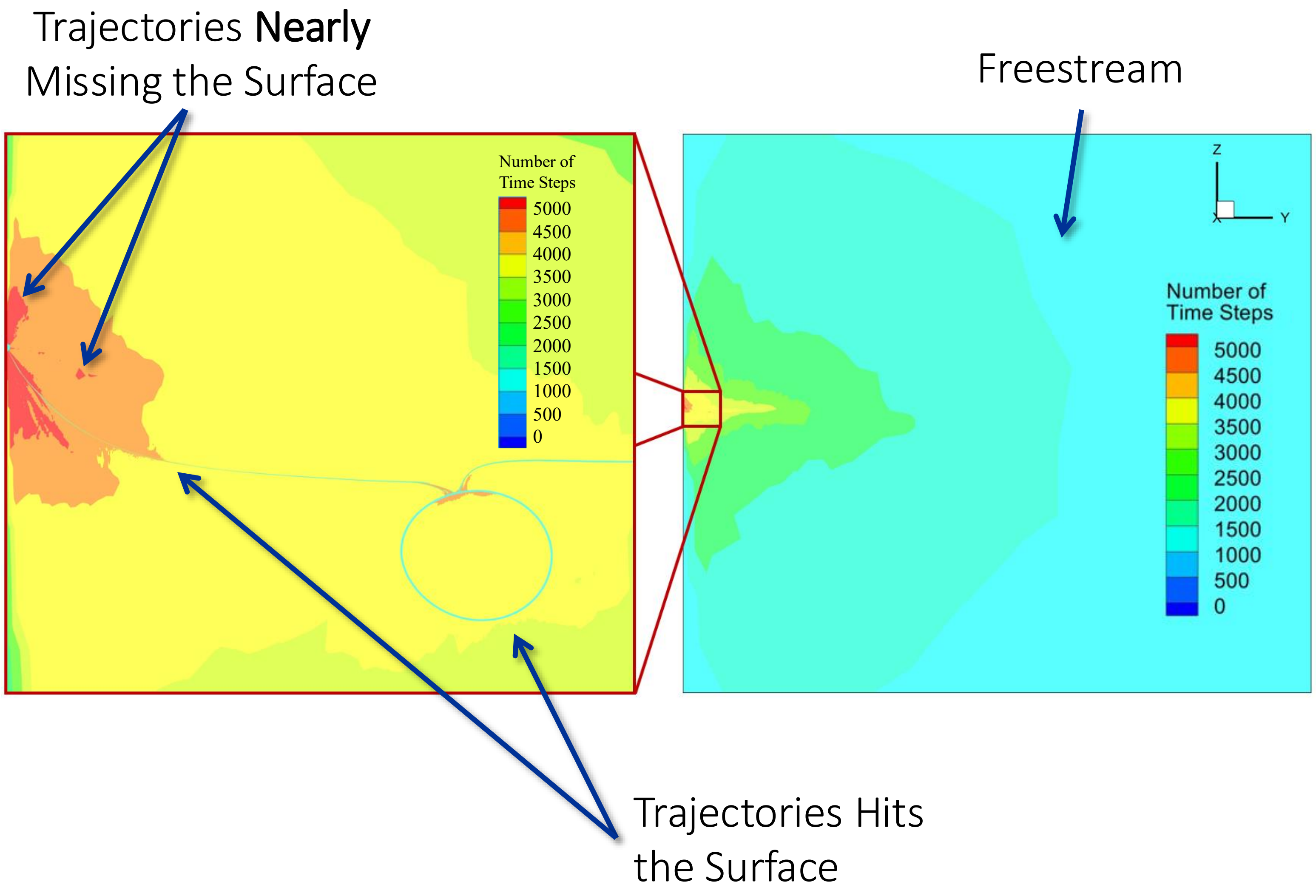
- In a **dynamic scheduling** process, the distribution of trajectories is not predetermined but is managed at runtime based on the current workload and processor availability.

	Number of Trajectories	
Processor	Static Scheduling	Dynamic Scheduling
1	*3*	--
2	*3*	*6*
3	*3*	*2*
4	*3*	*4*



Dynamic Scheduling and Load Management

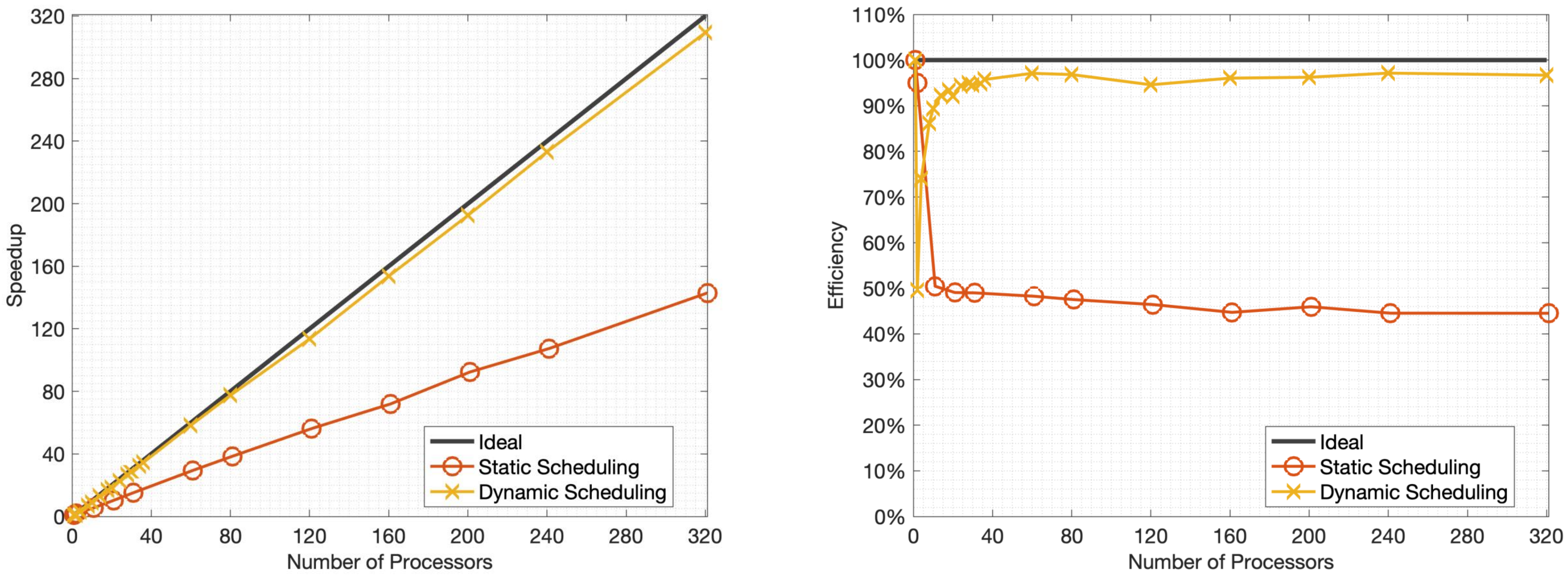
- An important part of the GlennICE software is the efficiency we have built into the trajectory routine
- A computation on a semi-span aircraft can be upwards of 50M+ trajectories
- Efficient computation of these trajectories is a key part of integration into a production workflow
- To achieve this, we manage runtime based on the current workload and processor availability



Speedup and Efficiency

$$\text{Speedup} = \frac{\text{Time } \mathbf{1\ Processor}}{\text{Time } x \text{ Processors}}$$

$$\text{Efficiency} = \frac{\text{Speedup}}{\text{Number of Processors}}$$



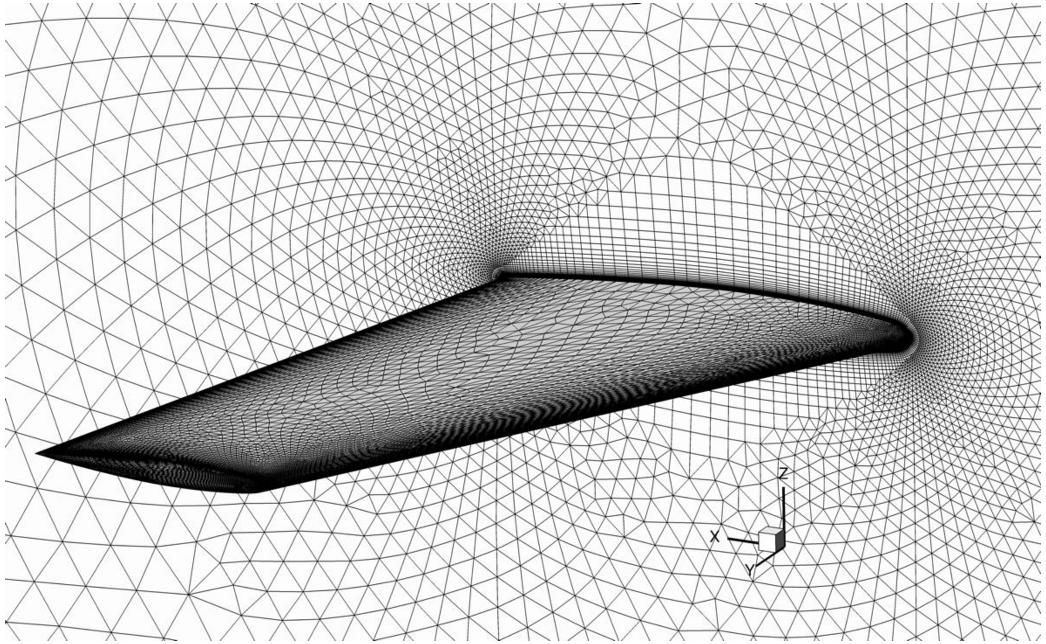
Roughness Augmentation and Heat Transfer

- LEWICE and LEWCE3D relied heavily on 1D empirical data
  - Tuned to transport aircraft
  - Based on IBL methods
  - Not applicable for new concepts or expanding the design envelope
- HTC is a crucial component because it dictates what the ice shape will be
- When a liquid droplet impacts the surface, it releases heat when it freezes
- If there is not enough convective heat transfer, only a portion of the droplet will freeze
  - If there is unfrozen water, it moves along the surface until freezing or running off
- Our three-dimensional methods still rely on empirical assumptions
  - We increase the convective heat transfer to account for the influence of icing roughness
  - Current methods in GlennICE work, but reliance on them is not a long-term strategy
  - We still need to develop methods that are informed by first-principle

ONERA M6 EXAMPLE CASE

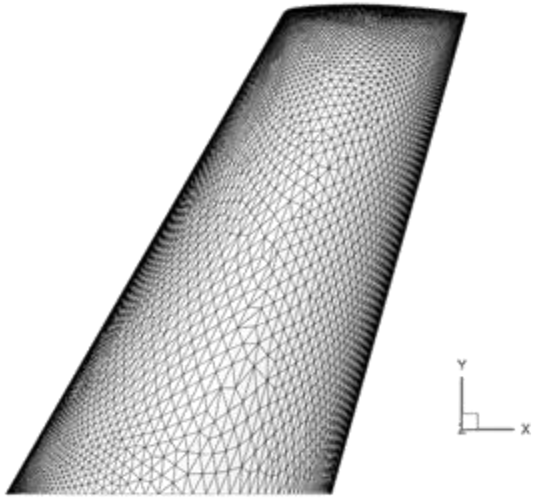
Case Setup

- Coupling between GlennICE and the CFD software is entirely one-way
- GlennICE generates a discretized flow field based on a CFD solution – .szplt or .fvuns
- Two methodologies exist for providing solutions:
  - The 1-Solution – user to provides heat transfer coefficient (HTC) and Adiabatic Wall Temperature (T_{aw})
  - The 2-Solution – requires two CFD solutions run with different isothermal wall temperatures and GlennICE computes HTC and T_{aw} internally
- We accept tetrahedral and mixed-element unstructured solutions



Geometry Description:

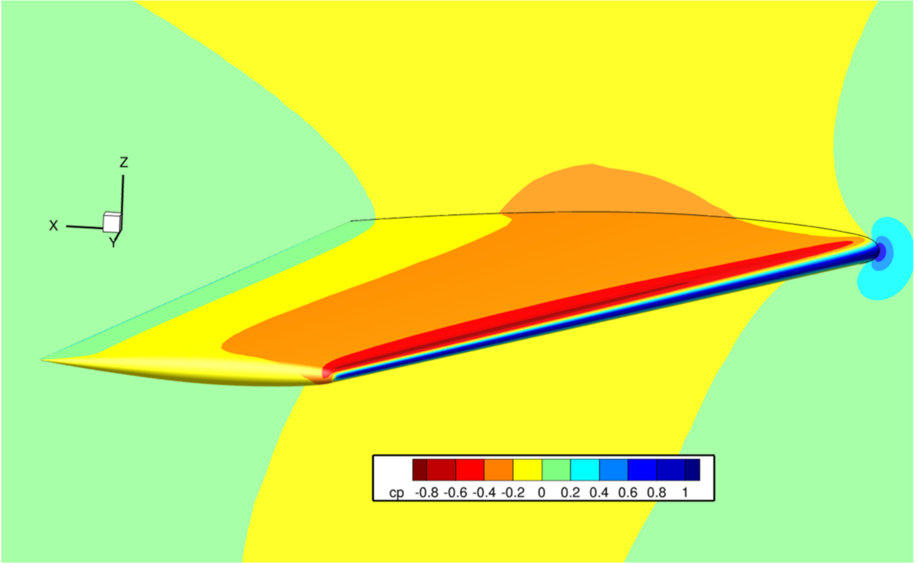
- ONERA M6 geometry
- Grid generated using Heldenmesh
  - 955,027 nodes
  - $y^+ \leq 1$
- Chord length of 1.0 m with a span of roughly 1.5 m
- Cantilevered off of a symmetry plane

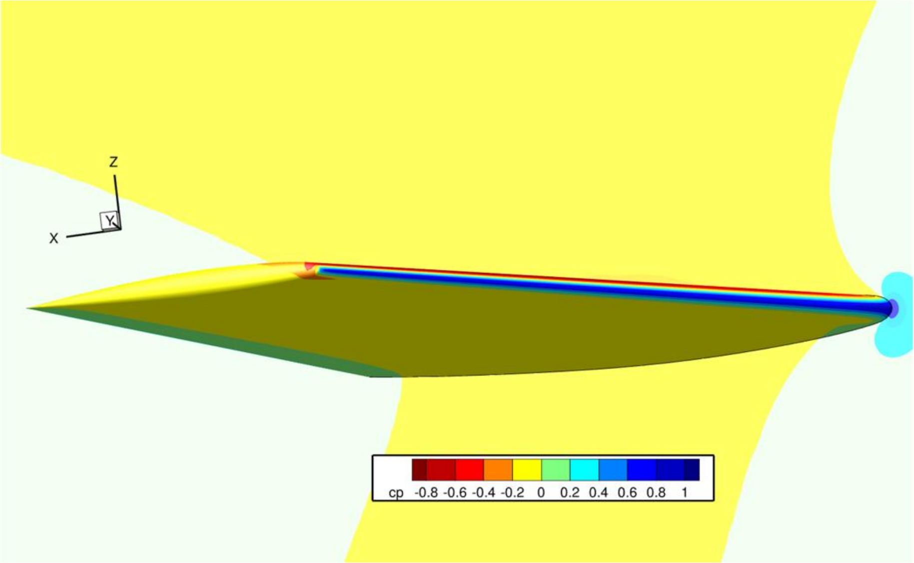


AOA (deg)	U_∞ (m/s)	M_∞	$T_{s,\infty}$ (K)	ρ_∞ (kg/m^3)	a_∞ (m/s)	$Re/C \times 10^6$
2	128.9	0.40	267.15	0.771	322.26	5.3087

CFD Analysis

- Using NASA's FUN3D solver to generate a steady RANS CFD solution
  - SA-neg turbulence model
  - Perfect-gas finite-volume discretization
- Imposing a nondimensional wall temperature of $W_{T,1} = 0.96$ and $W_{T,2} = 1.04$ as a boundary condition
  - $W_{T,1} = 0.96 \rightarrow W_{T,1} = 0.96 * T_{s,\infty} \rightarrow W_{T,1} = 0.96 * 267.15$ K$\rightarrow W_{T,1} = 256.464$ K
- Generate a .szplt file for the volume and boundary for each wall temperature
- 2-Soln Method requires a 'heating' variable, can also be named: Wall Heat Flux, surface heat flux, QWall





AOA (deg)	$U_\infty (m/s)$	M_∞	$T_{s,\infty}$ (K)	ρ_∞ (kg/m^3)	a_∞ (m/s)	$Re/c \times 10^6$
2	128.9	0.40	267.15	0.771	322.26	5.3087

Single-Bin Ice Accretion

- We have our CFD solutions, what now?
- Let us conduct a single-shot accretion of a single droplet diameter
- We choose a spray time of 2700s (equal to 45 minutes)
- Main parameters in the glennice.nml file
  - Convert all CFD solution variables to m-k-s units inside the submission file
  - Input freestream icing conditions shown below
  - Tag the appropriate surfaces for what the inlet and icing surfaces are
  - Specifying convergence metrics
- Typically, we run jobs on NASA Advanced Supercomputing
  - Job parameters through the glennice.nml file
  - Job submission through a .pbs job file (can also be run serially)

Diameter (μm)	LWC (g/m^3)	Spray Time (s)	Icing Temp. (K)	ρ_{ice} (kg/m^3)
20	0.51	2700	267.15	450

Properties of interest for our analysis

What This Actually Looks Like

glennice.nml Input File Parameters:

```
&files
  solution_in = "om6ste_volume_T1.szplt"
  solution_surface_in = "om6ste_boundary_T1.szplt"
  solution2_in = "om6ste_volume_T2.szplt"
  solution2_surface_in = "om6ste_boundary_T2.szplt"
  tecplot_out = "om6ste_tecplot_output.szplt"
  stl_outer_mold_line_out = "om6ste_IceShape.stl"
  volume_restart_file = "om6ste_Volume_Restart.dat"
  refinement_restart_file = "om6ste_Refinement_Restart.dat"
  surface_restart_file = "om6ste_Surface_Restart.dat"
/
&release
  inlet_coverage = "full"
/
&distribution
  diameter = 20
/
&bin_count
  nbins = 1
/
&adaptive_refinement
  fraction_contained_tol = 0.7
  max_face_count = 10000
  pct_converged_limit = 80
/
&surface_count
  number_of_surfaces=4
/
&surface_nml
  category =  "Inlet" , "nocategory" , "nocategory" , "Icing"
/
&freestream
  icing_temperature = 267.150
  twc = 0.51
  time = 2700.0
  ice_density = 450
```

```
&htc_augmentation
  augmentation_type = "McClain_roughness"
  turbulent_htc_augmentation = 3.0
  ideal_rime_limit = 0.004
/
&variable_properties
  ! Scaling density
  map_index = 4
  scale_factor = 0.7710
/
&variable_properties
  ! Scaling u-velocity
  map_index = 5
  scale_factor = 322.2565
/
&variable_properties
  ! Scaling v-velocity
  map_index = 6
  scale_factor = 322.2565
/
&variable_properties
  ! Scaling w-velocity
  map_index = 7
  scale_factor = 322.2565
/
&variable_properties
  ! Scaling pressure
  map_index = 8
  scale_factor = 80068.45220 ! ρ∞ * a∞^2
/
&variable_properties
  ! Scaling temperature
  map_index = 9
  scale_factor = 267.150
/
```

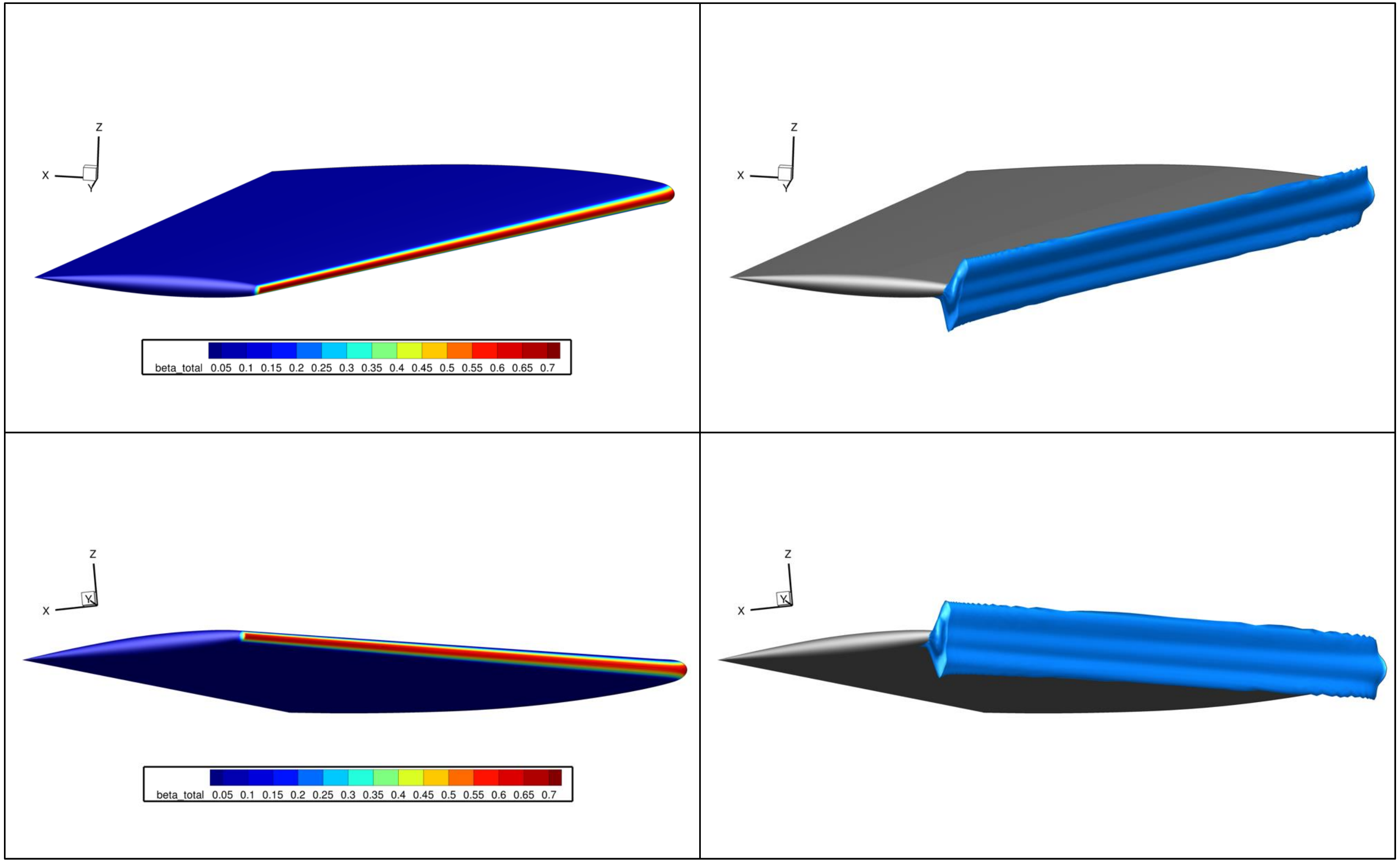
```
&variable_properties
  ! Scaling temperature
  map_index = 9
  scale_factor = 267.150
/
&variable_properties
  ! Scaling heating
  map_index = 10
  spelling = "heating"
  scale_factor = -10000.0 ! W/cm^2 to W/m^2 and a sign change
/
```

Command Line Execution:

```
$ mpiexec -np 128 glennice glennice.nml >
   single_bin_ice_accretion_log.txt
```

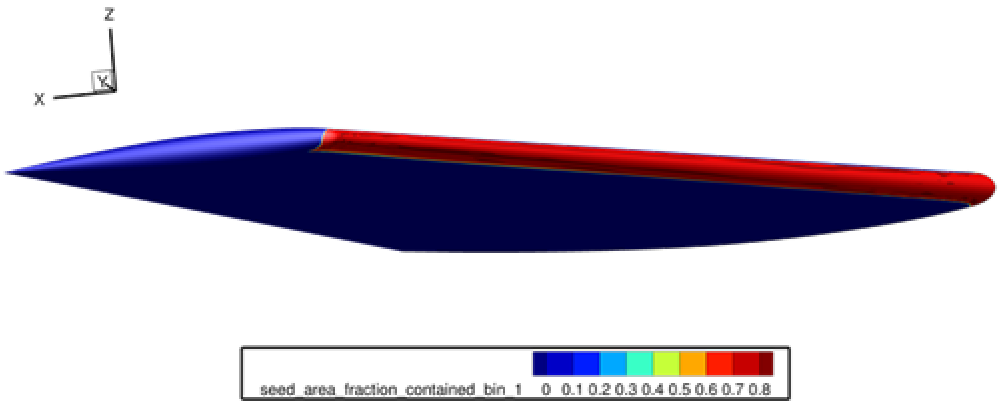
Single-Bin Ice Accretion

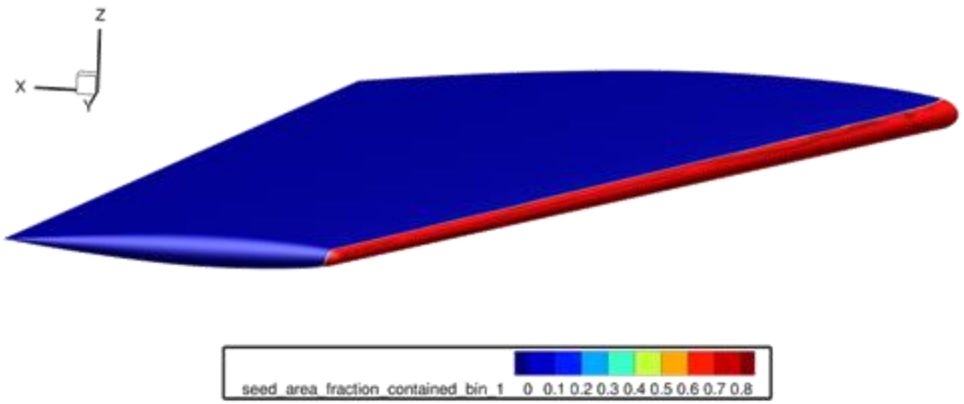
- So we computed our first ice shape: what do we look at?
- Two main files are output
  - The surface data .szplt file
  - The ice shape .stl file
- For a typical user, the key variables are often
  - Collection efficiency (beta_total)
  - Computed ice shape
- Collection efficiency will show us where we get water impinging and how much
- We also compute and print out
  - Convergence metrics
  - Ice Mass, Volume, & Thickness
  - Hit counts
  - Augmentation values
  - Freezing fraction
  - Many more



Changing Convergence Metrics With a Refinement Restart

- Within GlennICE we have different ways to determine whether a job can be considered ‘converged’
- Our recommended procedure is setting a value for pct_converged_limit
  – This allows for the code to stop once we have achieved this limit
- For our first job, we set pct_converged_limit at 80%
  – What happens when we switch this value to be 90% instead
- The knowledge of what this variable actually represents is a little more technical
- A user can think about this as the percentage of all the faces that achieve a criteria that has been set in the glennice.nml file
- In a Eulerian solver, you have a known residual that defines convergence
  – This does not exist for a Lagrangian scheme





What Our Standard Output Will Show

```
Refinement iteration: 16, number release points: 9868242
   Surface 4: "boundary 4 Airfoil"
      Number faces hit: 5274, min hit count: 1, max hit count:
   9589, total hits: 9556528
      Total mass flow: 2.380617939E-03, l2 convergence:
   9.537353706E-05
      Percent of impacted faces converged (by metric):
      fraction_contained_tol:  86.04%, mass_flow_tol:   0.00%,
   total:  86.04%
```

We change the pct_converged_limit to 90% and utilize a refinement restart

```
$ mpiexec -np 128 glennice glennice.nml restartJob=Refinement >
   refinement_restart_log.txt
```

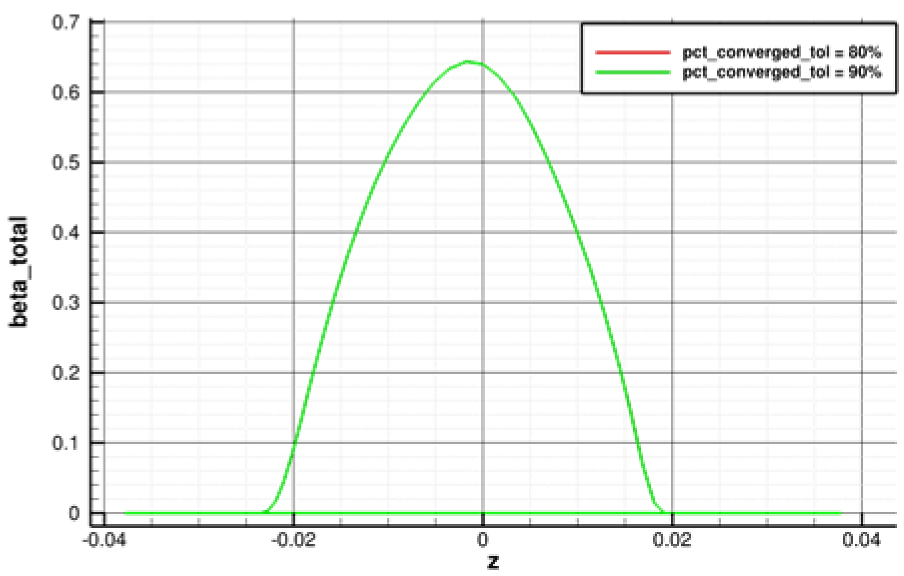
We see our original metric of 80% was met

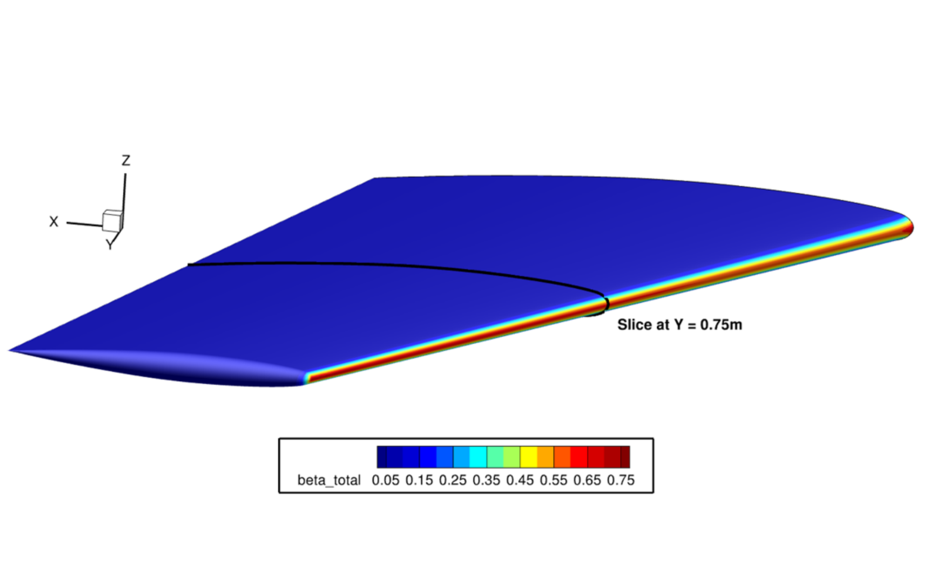
```
Refinement iteration: 17, number release points: 12892366
   Surface 4: "boundary 4 Airfoil"
      Number faces hit: 5288, min hit count: 1, max hit count:
   17762, total hits: 12321239
      Total mass flow: 2.380496000E-03, l2 convergence:
   1.961877346E-05
      Percent of impacted faces converged (by metric):
      fraction_contained_tol:  95.92%, mass_flow_tol:   0.00%,
   total:  95.92%
```

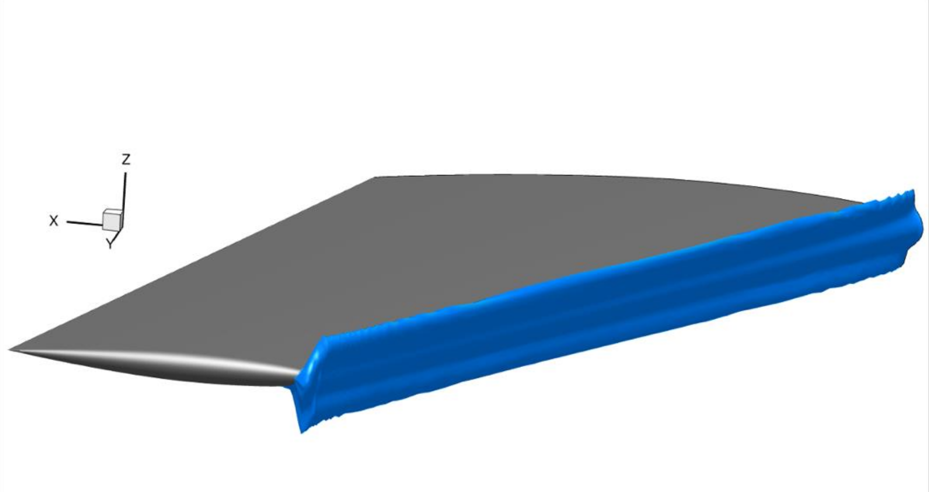
We achieve a fraction_contained_tol above our 90% threshold in the next iteration

Convergence Metric Change Using a Refinement Restart

- So we changed our pct_converged_limit from 80% to 90%, what happened to our results?
- If we look at the ice shape, very little seems to have changed
- Take a slice at Y = 0.75 m
  - We see that collection efficiency is very similar
- So, it 'appears' that nothing discernable happened, so let us take a look at what changed between runs





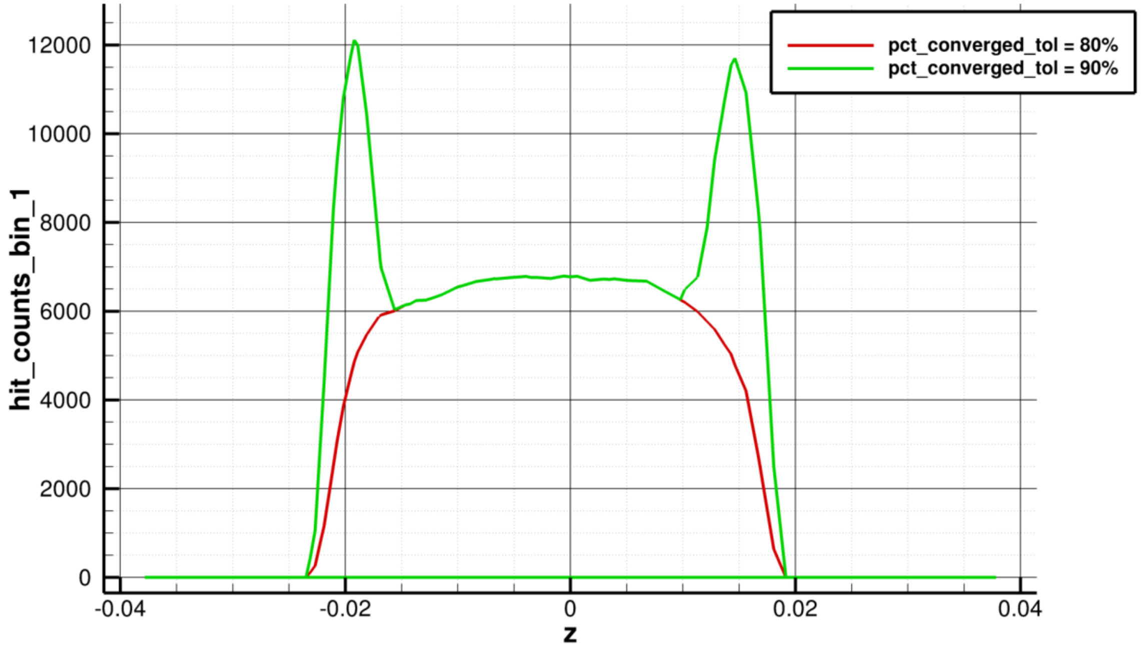


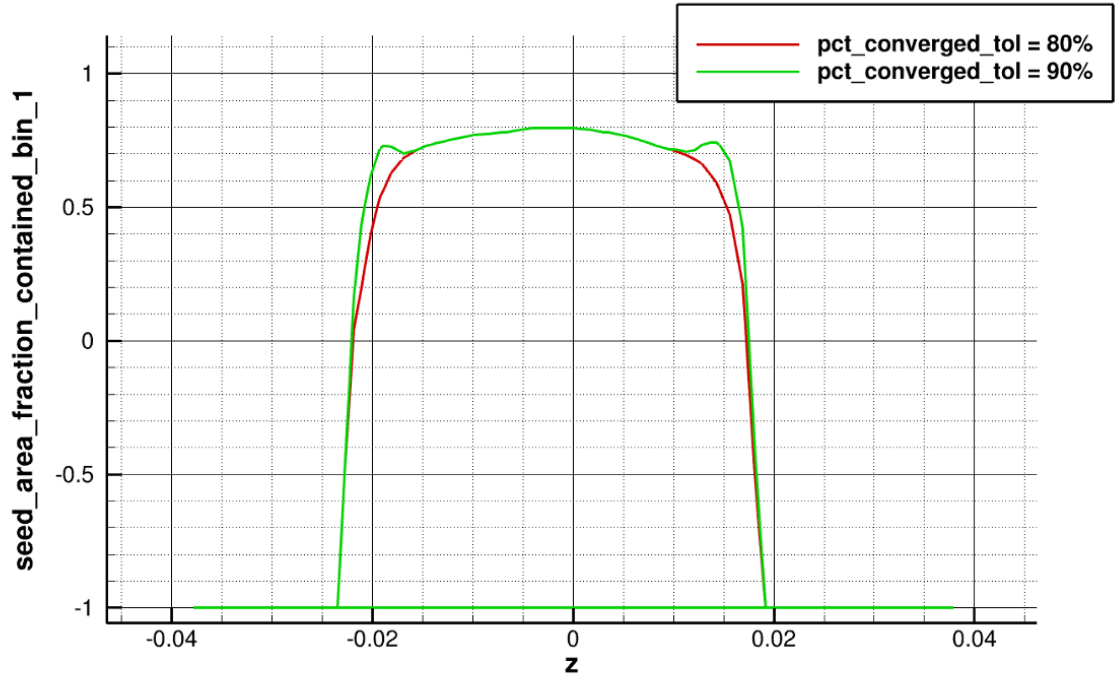
Minimal differences compared to 80%

Convergence Metric Change Using a Refinement Restart

- Our metrics for convergence added trajectories at the impingement limits
- The number of trajectory hits on the surface increased significantly in this region
- Our convergence metrics improved in the impingement limit region

These metrics and the results will not always behave the same for every configuration



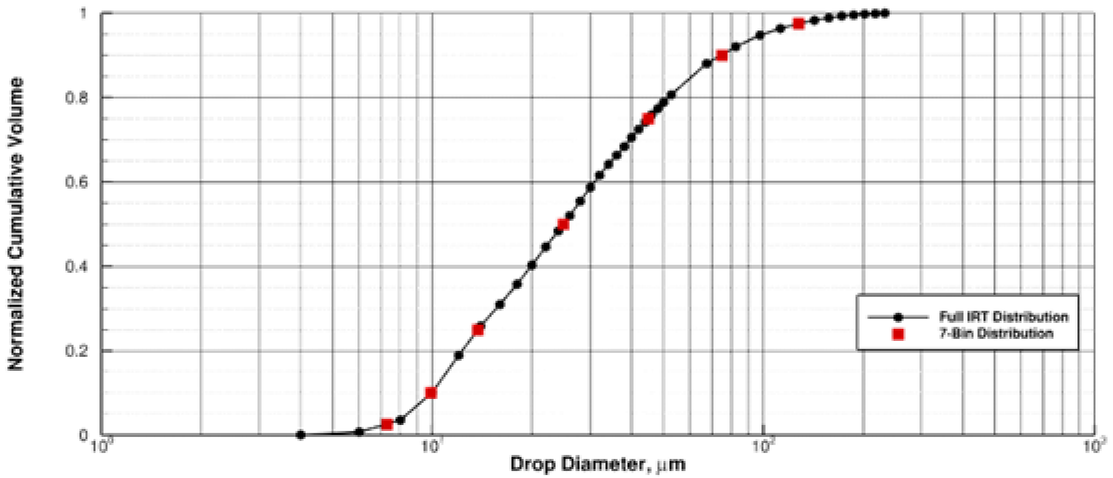


Multi-Bin Ice Accretion

- In reality, droplets are not encountered at a single diameter but in a distribution of droplets
- We can approximate these distributions through binning the diameters and how much of the total volume of water they take up
- We base our accretion parameters off the median volume diameter (MVD) and what we will call the mass fraction
- Our mass fraction says what fraction of the total mass each droplet contributes
  - If a single diameter accretes 800 g of mass and their mass fraction is 0.1, they will end up contributing 80 g of mass to the solution

How this is specified within the glennice.nml file:

```
/
&distribution
  diameter = 6.0, 10.0, 14.0, 20.0, 27.0, 35.0, 44.0
  mass_fraction  = 0.05,  0.1,  0.2, 0.3,  0.2,  0.1, 0.05
/
&bin_count
  nbins = 7
/
```



IRT 7-Bin Distribution for an MVD = 24.9 μm

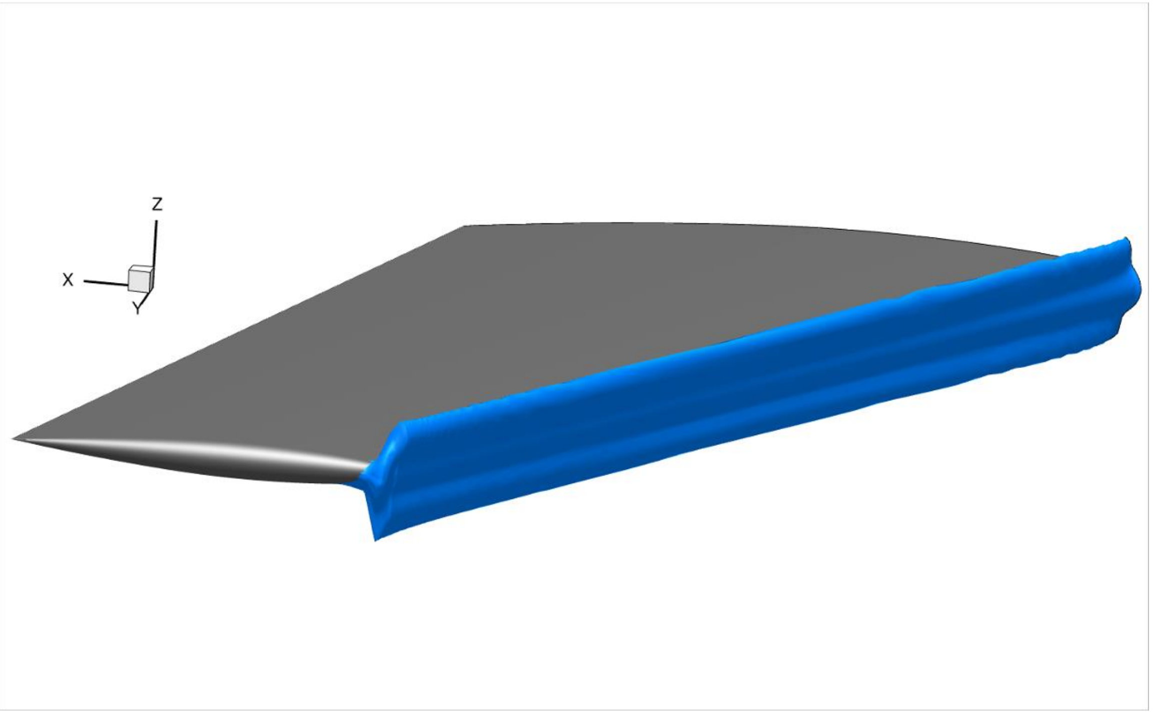
The distribution we will assume for this problem:

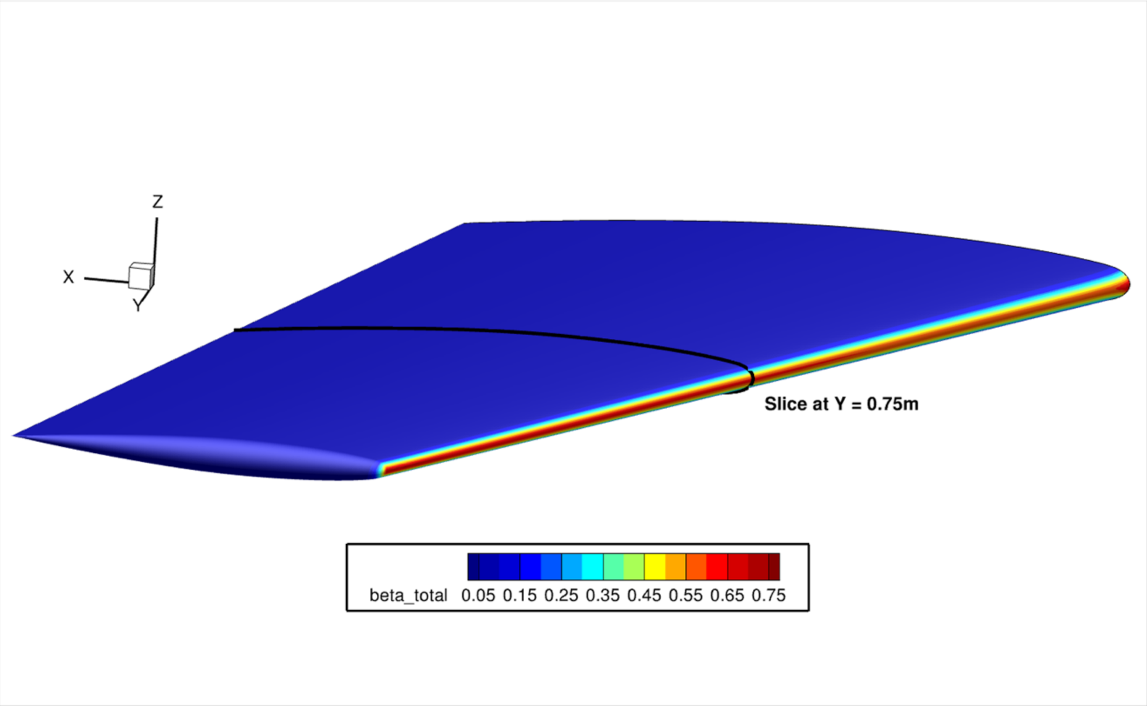
Diameter (μm)	6	10	14	20	27	35	44
Mass Fraction	0.05	0.1	0.2	0.3	0.2	0.1	0.05

Langmuir D MVD = 20μm Distribution

Multi-Bin Ice Accretion

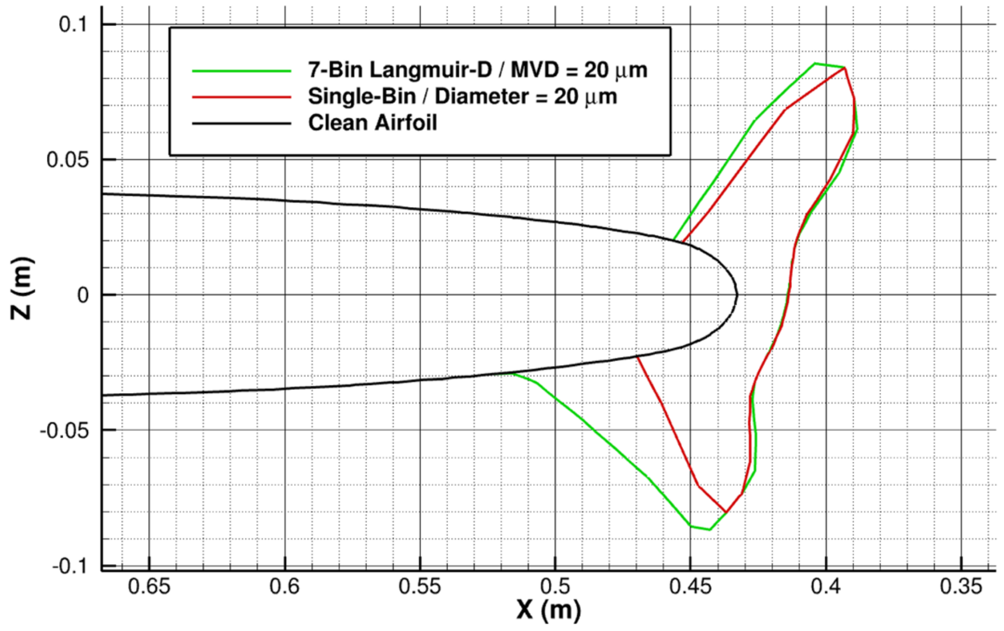
- When you run multiple bins, each droplet diameter will behave and converge differently
  - The path of a large droplet will behave differently than a small
  - Lighter particles move with the flow and turn more easily than heavier particles
- This means that you have to converge each diameter individually
- In the end, we combine all of the individual diameters and account for the combined influence of each

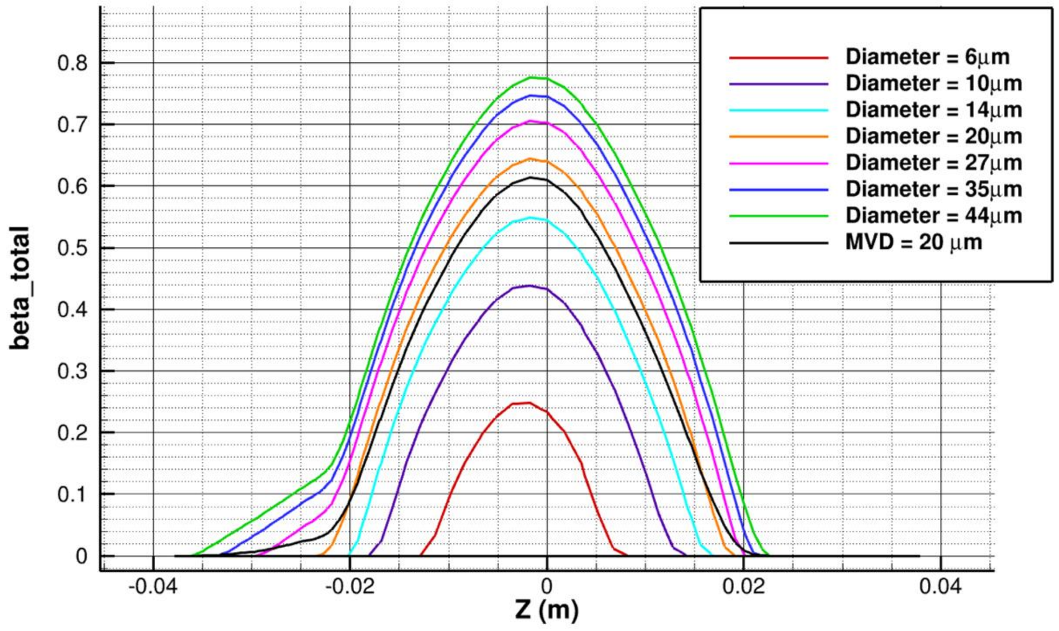




Multi-Bin Ice Accretion

- We see in our slice that a multi-bin accretion has ice forming further back on the airfoil
  - The green line moves further back in the streamwise direction
- We see that the peak water loading on the leading edge is slightly smaller
  - Looking at peak collection efficiency the diameter = 20µm is larger than MVD = 20µm
- We can see the larger ice shape is being driven by the amount of water impinging beyond the diam = 20µm profile shown in orange
- These differences drive the requirements for ice protection systems
  - How big, how far back, how much energy is required, etc.





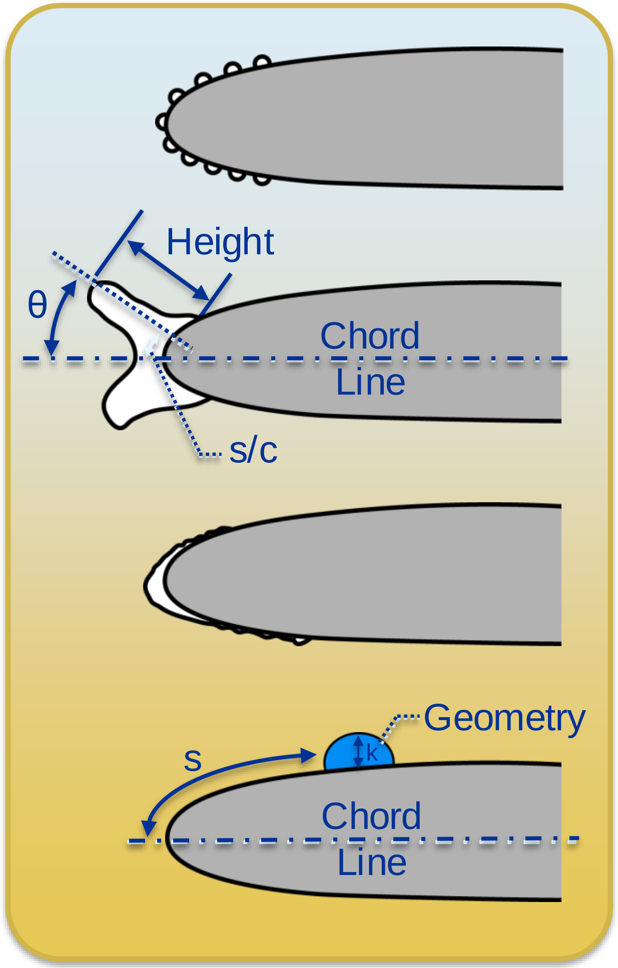
Temperature and Icing

- In icing there is never one condition that you have to certify your aircraft or IPS to – it is always multiple
- Different icing temperatures result in different ice shapes:
  - Warmer temperatures result in glaze or mixed ice shapes (pilots call this clear ice)
  - Colder temperatures result in rime ice shapes
  - Correctly predicting what location on the airfoil and at what temperature is challenging
- Glaze ice shapes tend to form large horns (like what we showed previously)
  - Dominated by water run back and the migration of unfrozen water
- Rime ice shapes tend to form more aerodynamic shapes
  - Dominated by droplets freezing immediately upon impact
- Sizing an ice protection system has to account for all situations
  - Not abating all of the accreted ice can produce ridge ice – unmelted or refrozen ice aft of the IPS which can result in accretion where it cannot be mitigated

Four Groups of Ice Shapes

If we classify based primarily on flowfield physics:

- Roughness (often earlier accretion times)
  - Ice Roughness is larger than the local boundary layer.
  - Effects determined by height, concentration, surface location.
- Horn ice (often from glaze conditions)
  - Characterized by large flow separation.
  - Horn size, location, and angle are key parameters with roughness and the detailed cross-section geometry having little effect.
- Streamwise ice (often from rime conditions)
  - Forms streamlined shape on the leading edge, with localized flow separation.
  - Surface roughness can have significant effect on aerodynamics.
- Spanwise-ridge ice (often from SLD, reimpingement, refrozen ice)
  - Obstacle in the flow since “airfoil” boundary layer has time to develop.
  - Location and height are key parameters, but ridge geometry and airfoil geometry are also important.

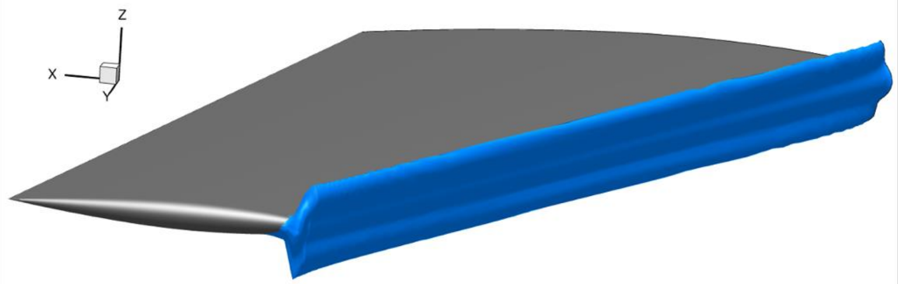


Temperature Change

- For this example, we will examine how to change the temperature and generate a new ice shape for a cold condition
- We currently make the assumption that changing the temperature is independent of impingement characteristics
  - The trajectories and impingements for a diam = 20µm particle at -3˚C is the same as one for -20˚C
  - This assumption has shown to be valid when doing a single-shot approach
- What we do is make changes in the mass and energy balance
  - How much freezes on impact
  - How much unfrozen water gets migrated further back
- This temperature change can be analyzed using previously computed trajectories
  - Efficiently conduct trade studies for an entire aircraft
  - Save time by not recomputing trajectories
  - Easy automation of the procedure
- Looking at a temperature sweep in this way is common in industry
  - This method helps to identify what the critical ice shape is for certification

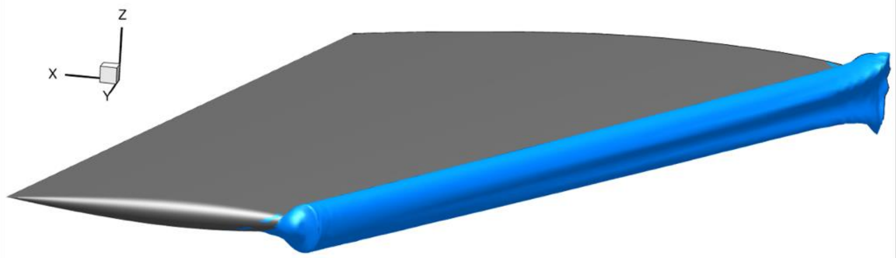
Temperature Changes

- We will demonstrate this by decreasing the temperature from T = 267.15K to T = 257.15K
- Will leverage the multi-bin analysis produced previously and perform a surface restart with GlennICE
- We can see that the new ice shape no longer produces horns
- The new ice shape is formed based upon an increased amount of water that is freezing upon impact
- Let us see how this changes some of the surface properties of interest



T = 267.15 K ice shape

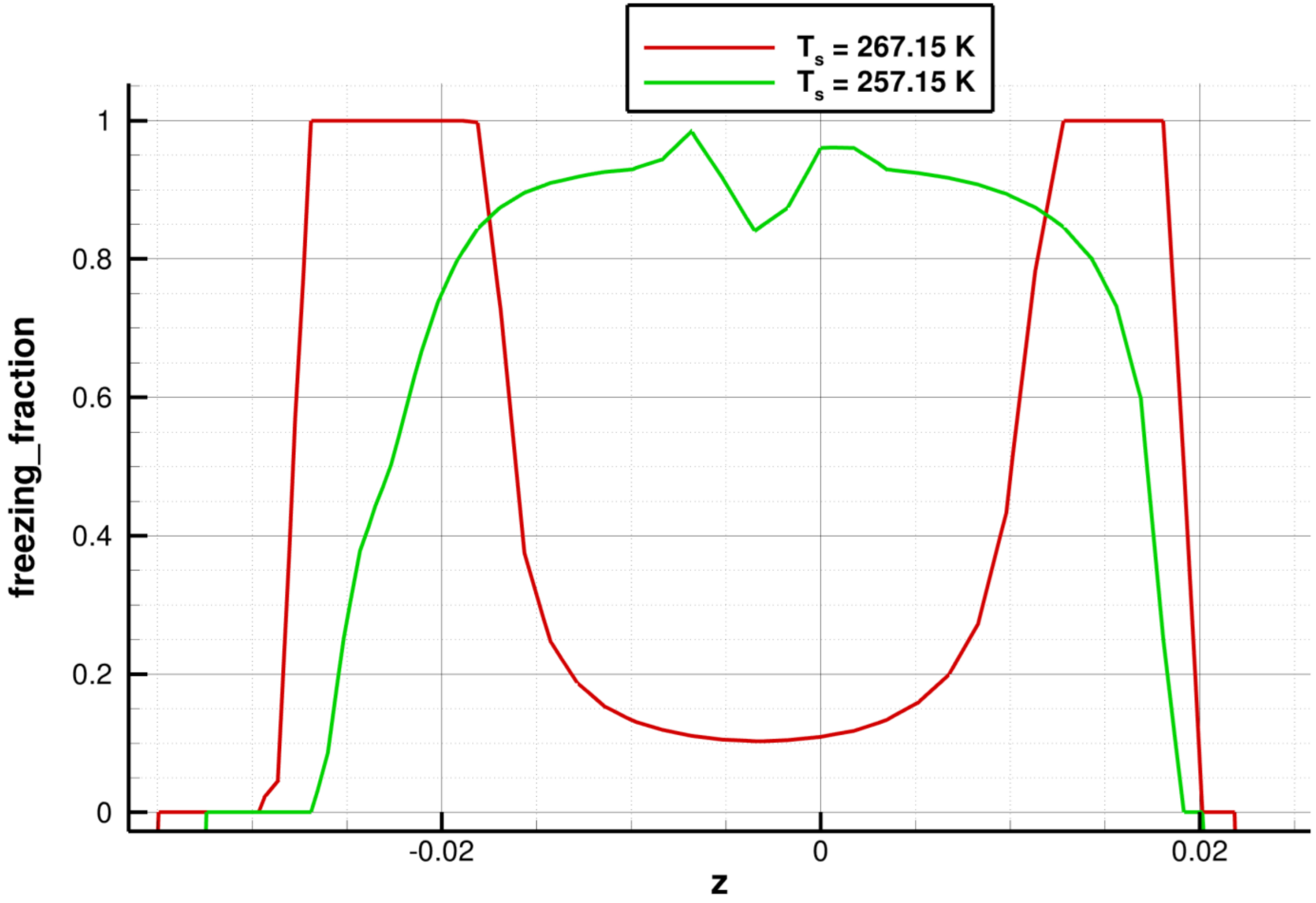
Decreasing Temperature



T = 257.15 K ice shape

Temperature Change Using a Surface Restart

- Utilizing the slice location of Y = 0.75m as before
- Freezing fraction is describing the amount of water that is freezing on each face
- It shows that we have significantly changed the location where ice is freezing
  - Horns are no longer being generated
  - Less incomplete freezing of particles
  - More ice is being formed on the leading edge

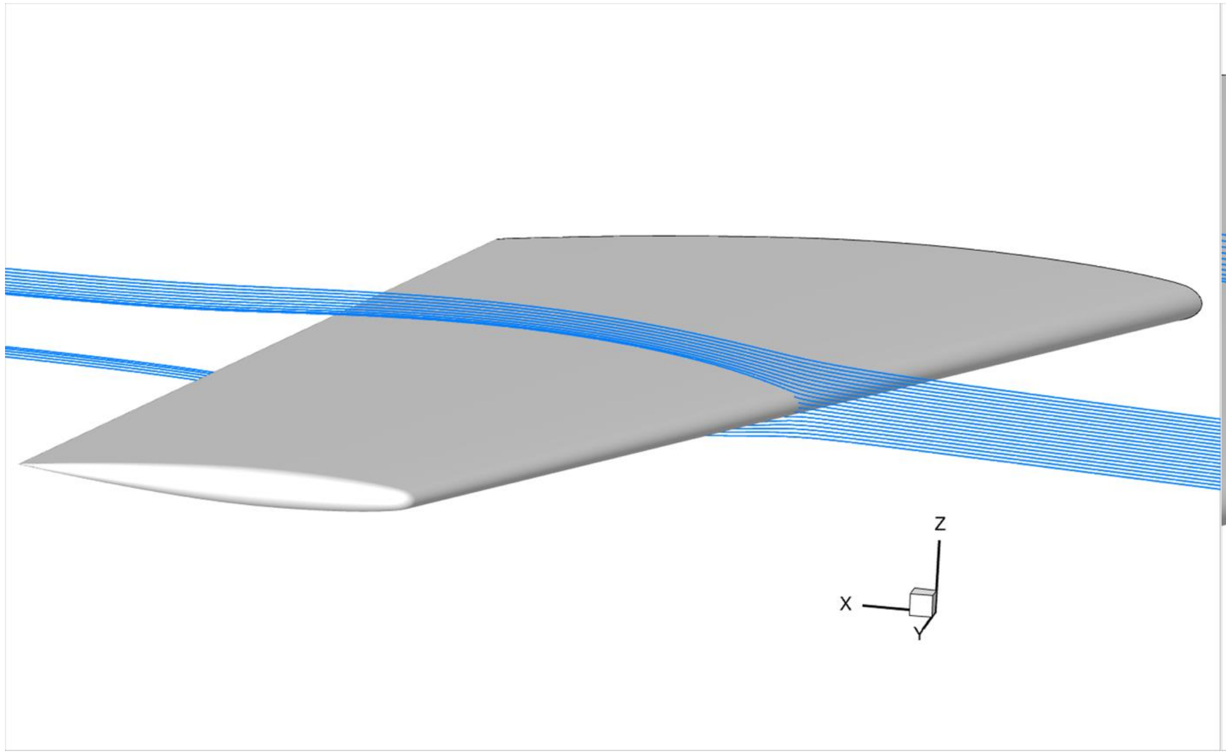


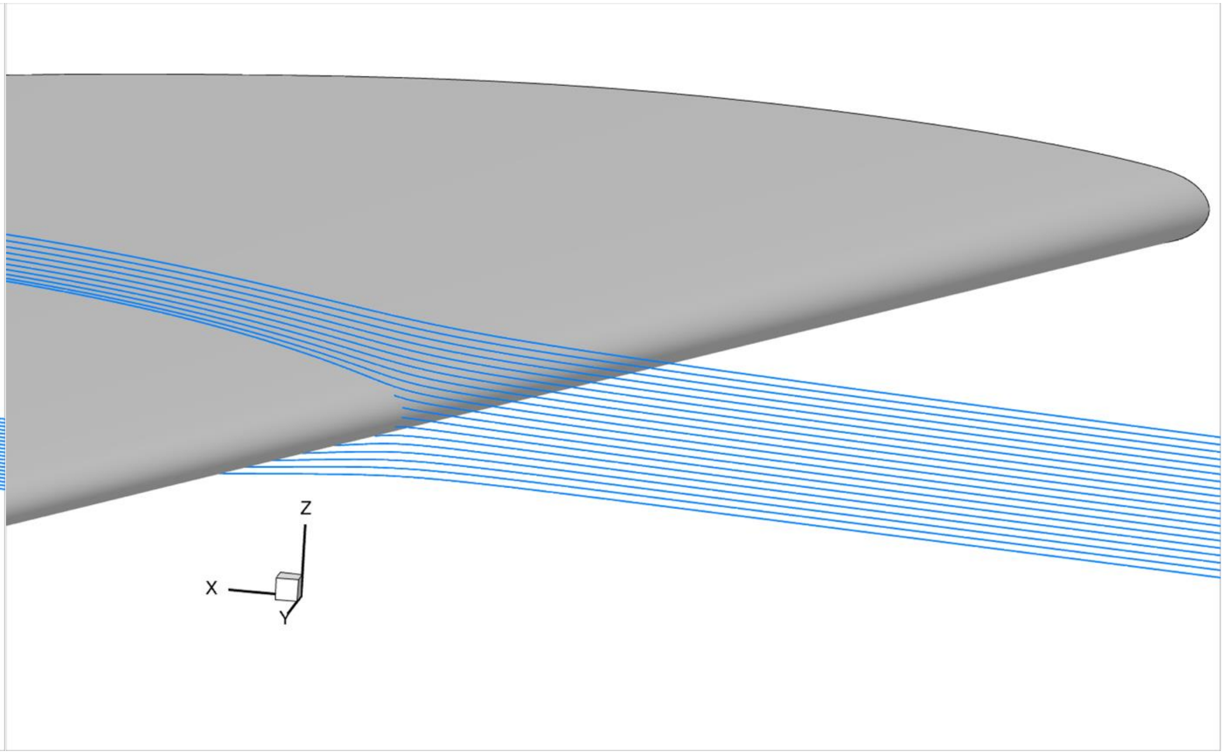
Trajectory Visualization

- Particle trajectories differ from simple streamline
  - Particles have a mass and size to them
  - Gravity and other forces can influence trajectories
  - These effects are related to the Stokes number or the modified inertia parameter
- For this example, we will release a rake of trajectories with a size of 20µm for the configuration we have been working on
- This capability is often used for visualization and debugging and is not looked at in a large-scale situation
- It is important to keep in mind that different particles will behave differently since mass with scale with r^3 and drag will scale with r^2

Trajectory Visualization

Particles that impinge on the geometry terminate, while those that do not continue traveling out of the domain

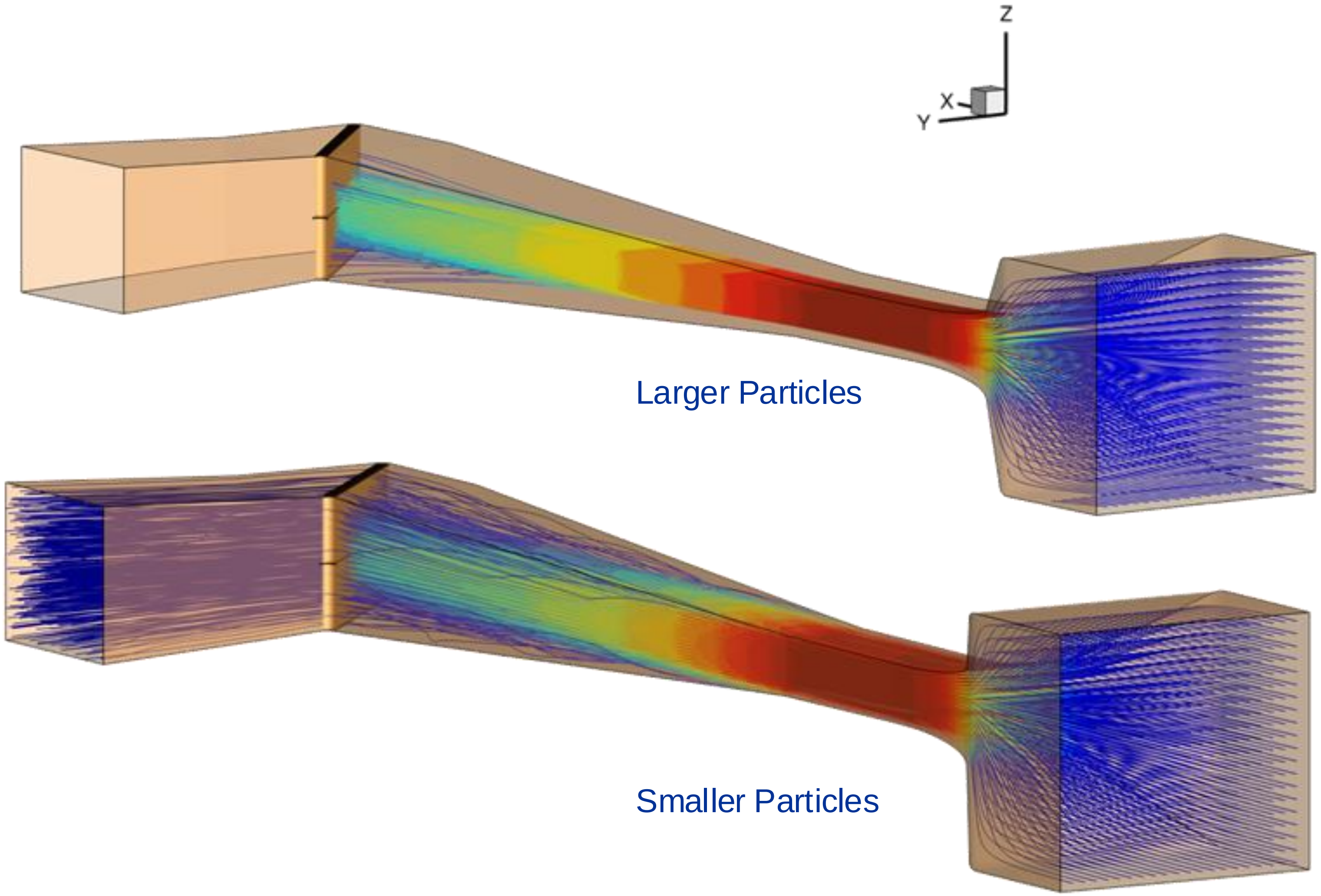




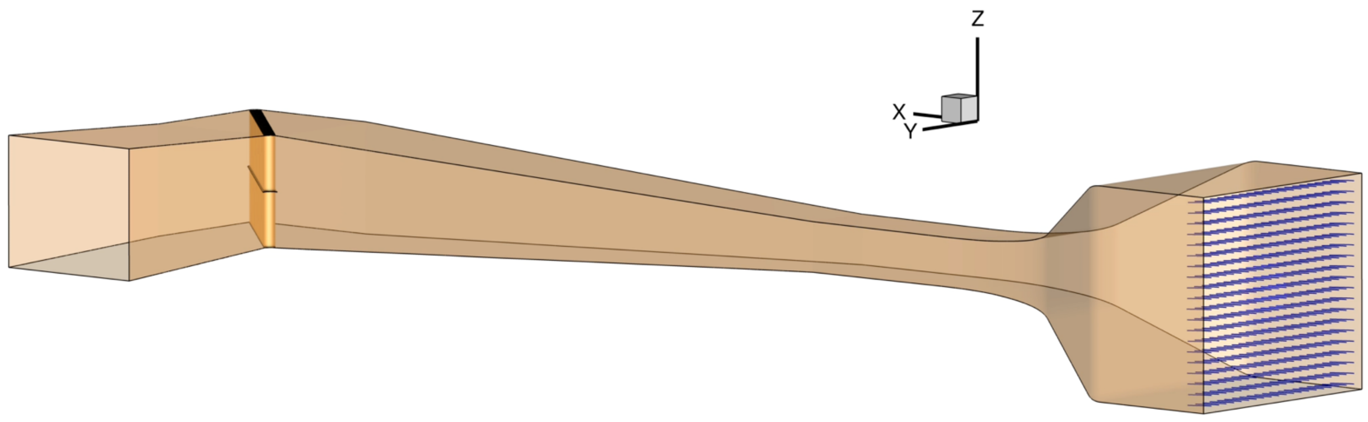
IRT TRAJECTORY VISUALIZATION

Trajectory Visualization – IRT

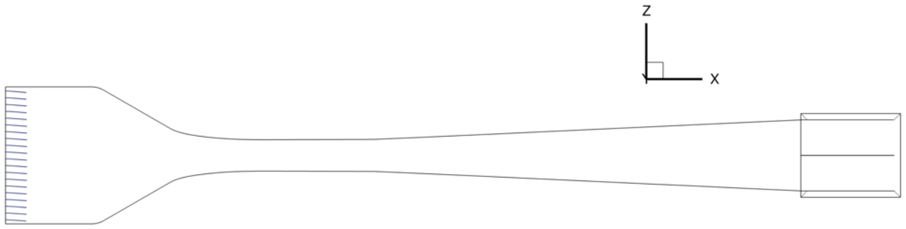
- In the IRT, we have spray nozzles that produce our icing cloud
  - Note: These droplets are not being released at the same location
  - Note: The droplets are not the same size or distribution
- Highlights the different behavior of differently sized particles
- Can easily see the influence gravity has



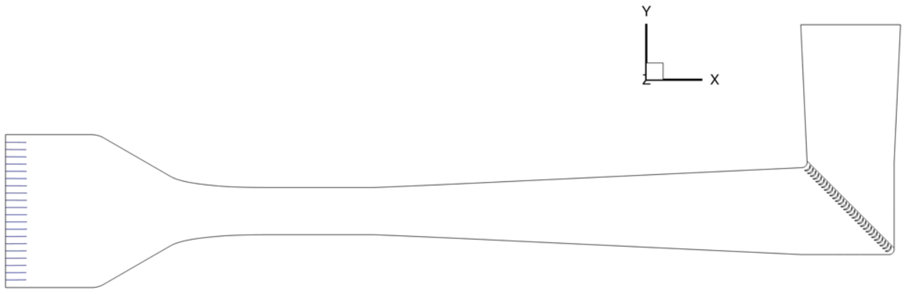
Trajectory Visualization – IRT



Iso View



Side View

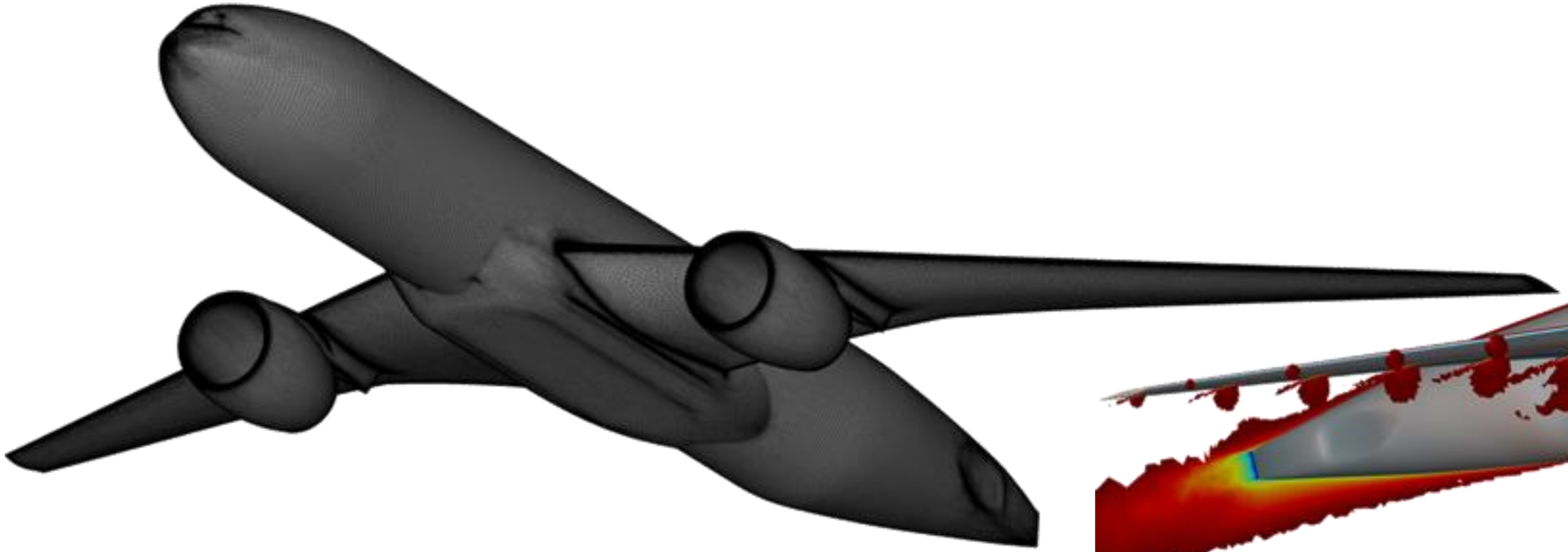


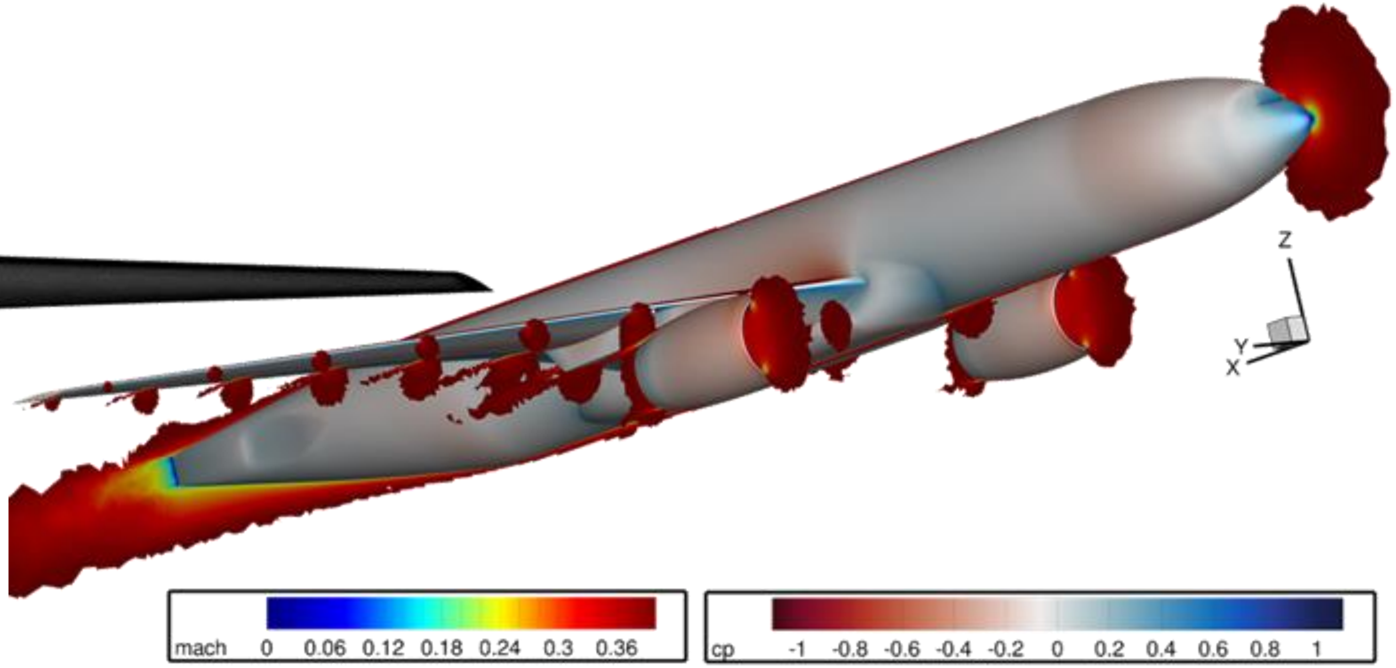
Top View

CRM-HL EXAMPLE CASE

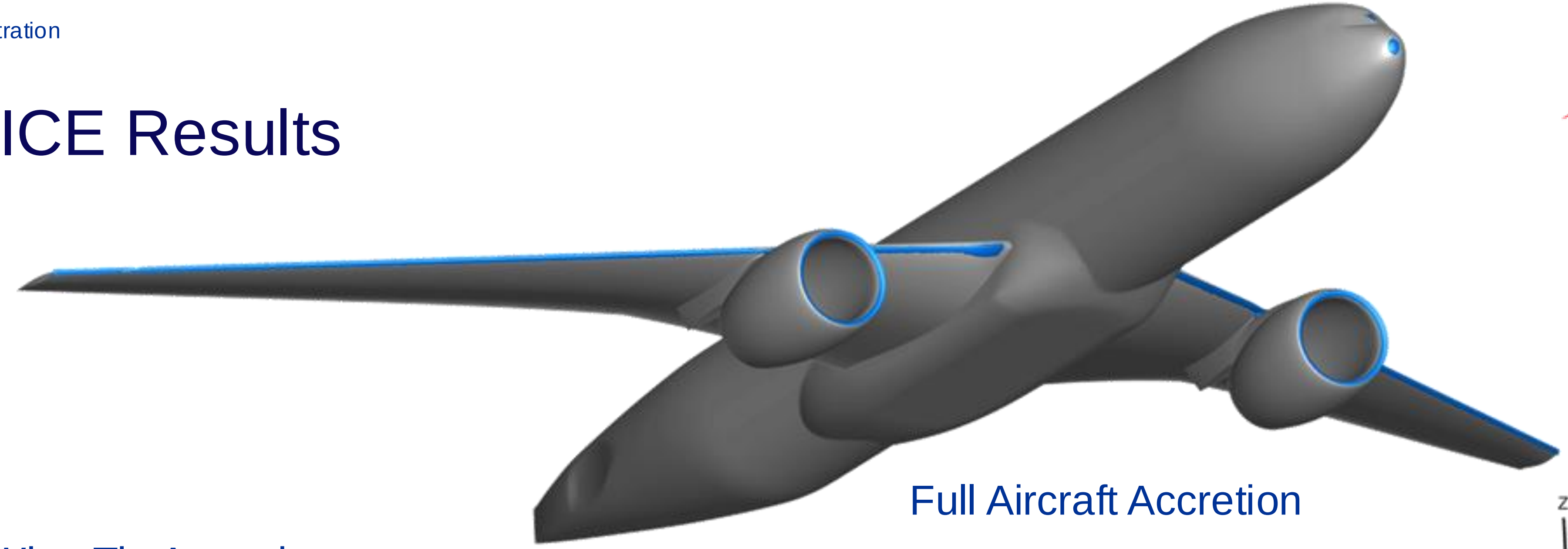
Case Background

- The High Lift Common Research Model (CRM-HL) has been a research platform that NASA and others have used for extensive experimental and computational analyses
- The goal is to be able to develop tools for predicting CLmax for relevant configurations
- Part of improved CLmax prediction is doing so with the inclusion of artificial ice shapes
- GlennICE has been used along with LEWICE3D to compute ice shapes on the CRM-HL for experimental analyses while showcasing refinement techniques



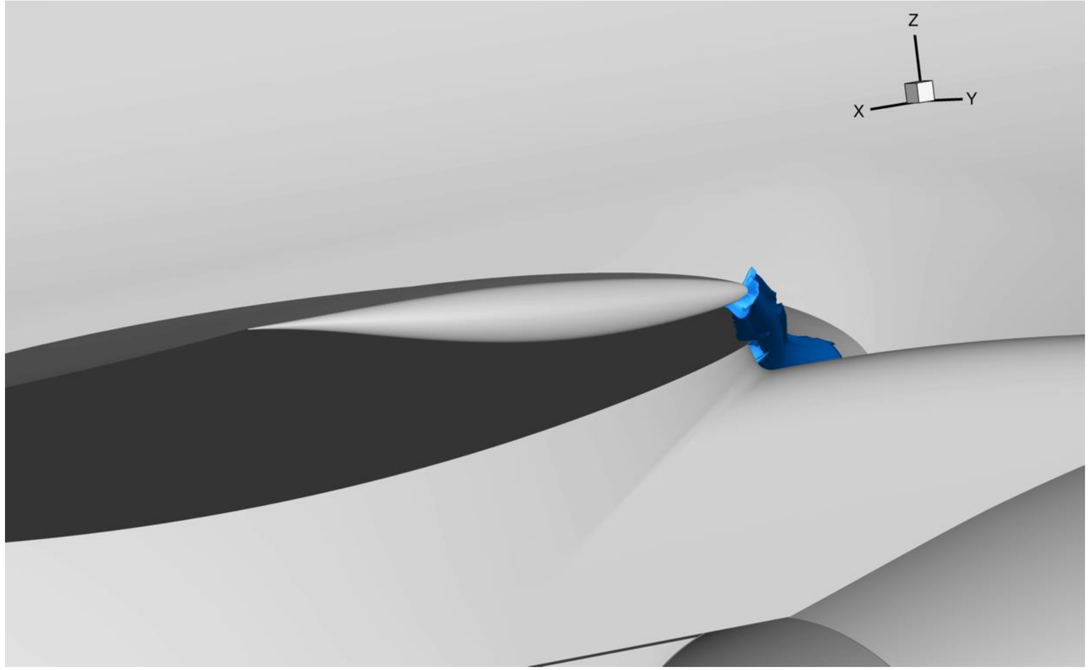


GlennICE Results

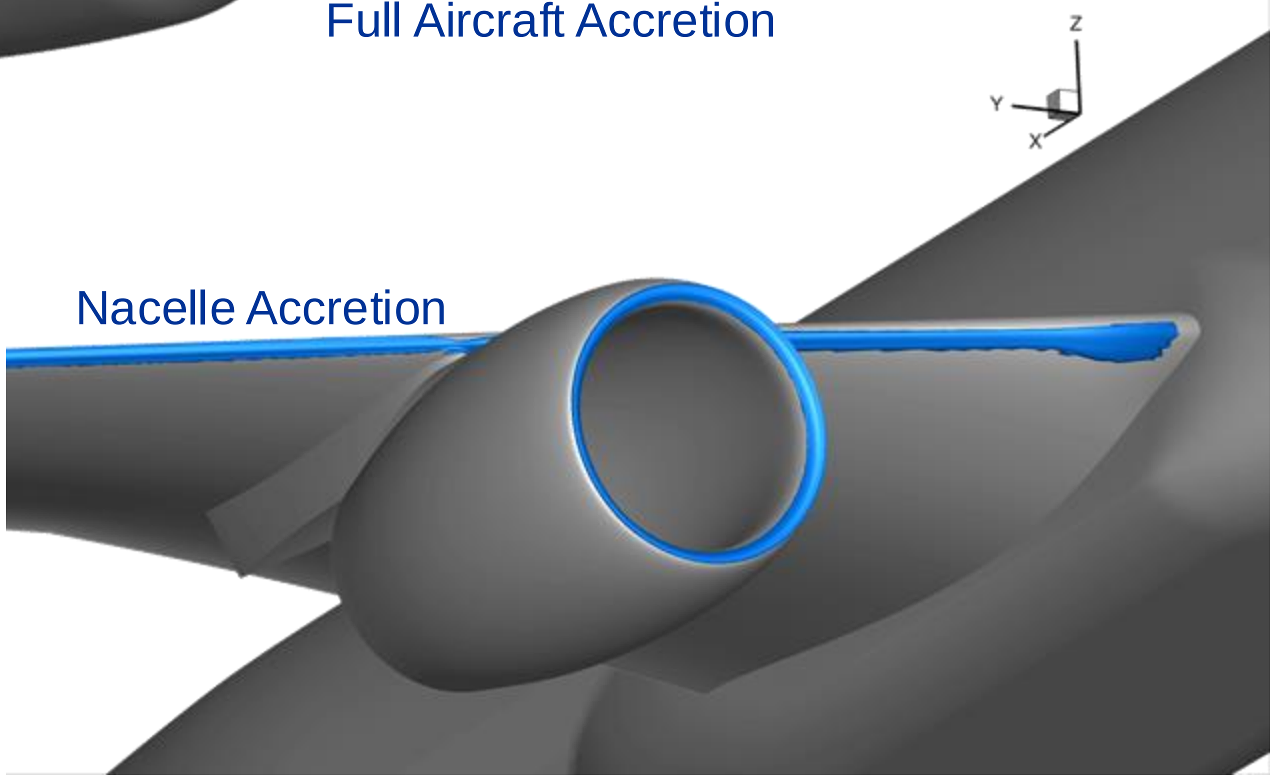


Full Aircraft Accretion

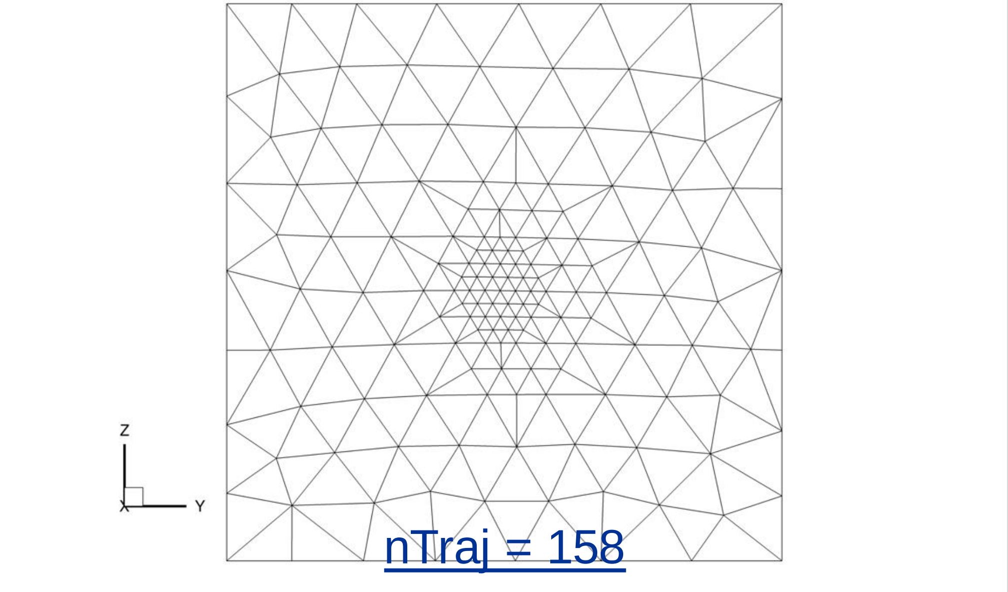
Wing and Wing Tip Accretion



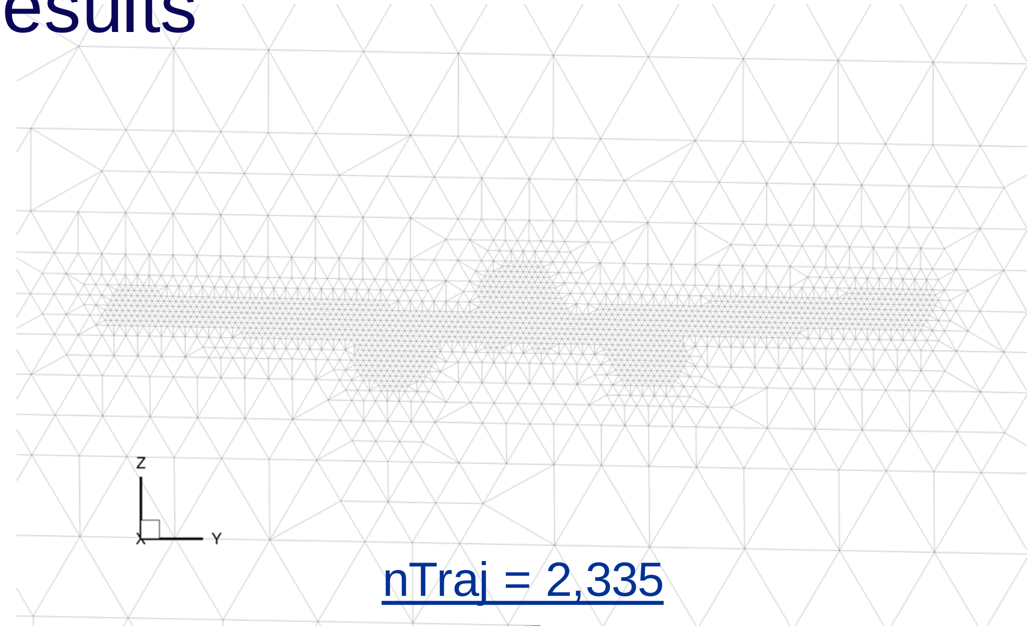
Nacelle Accretion



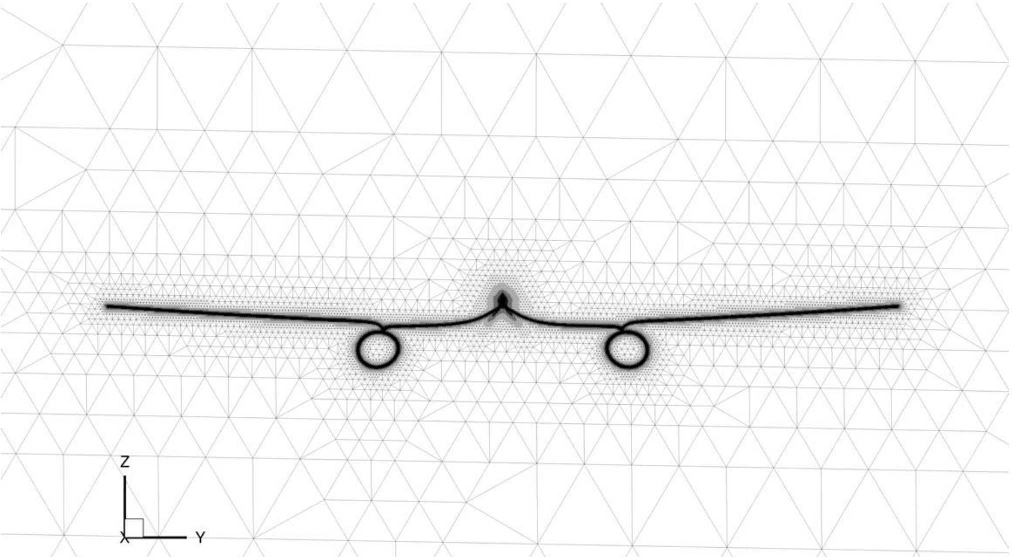
GlennICE Results



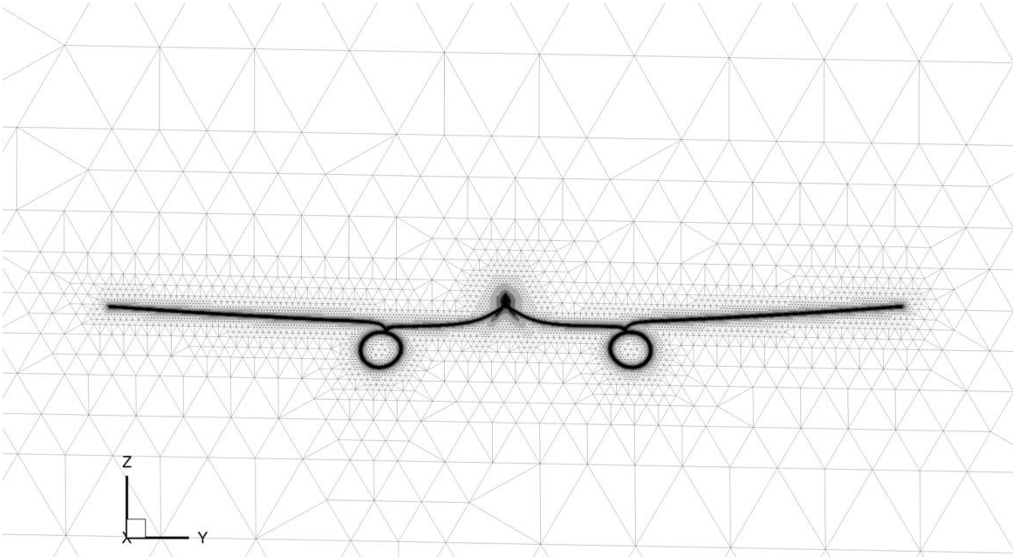
nTraj = 158



nTraj = 2,335

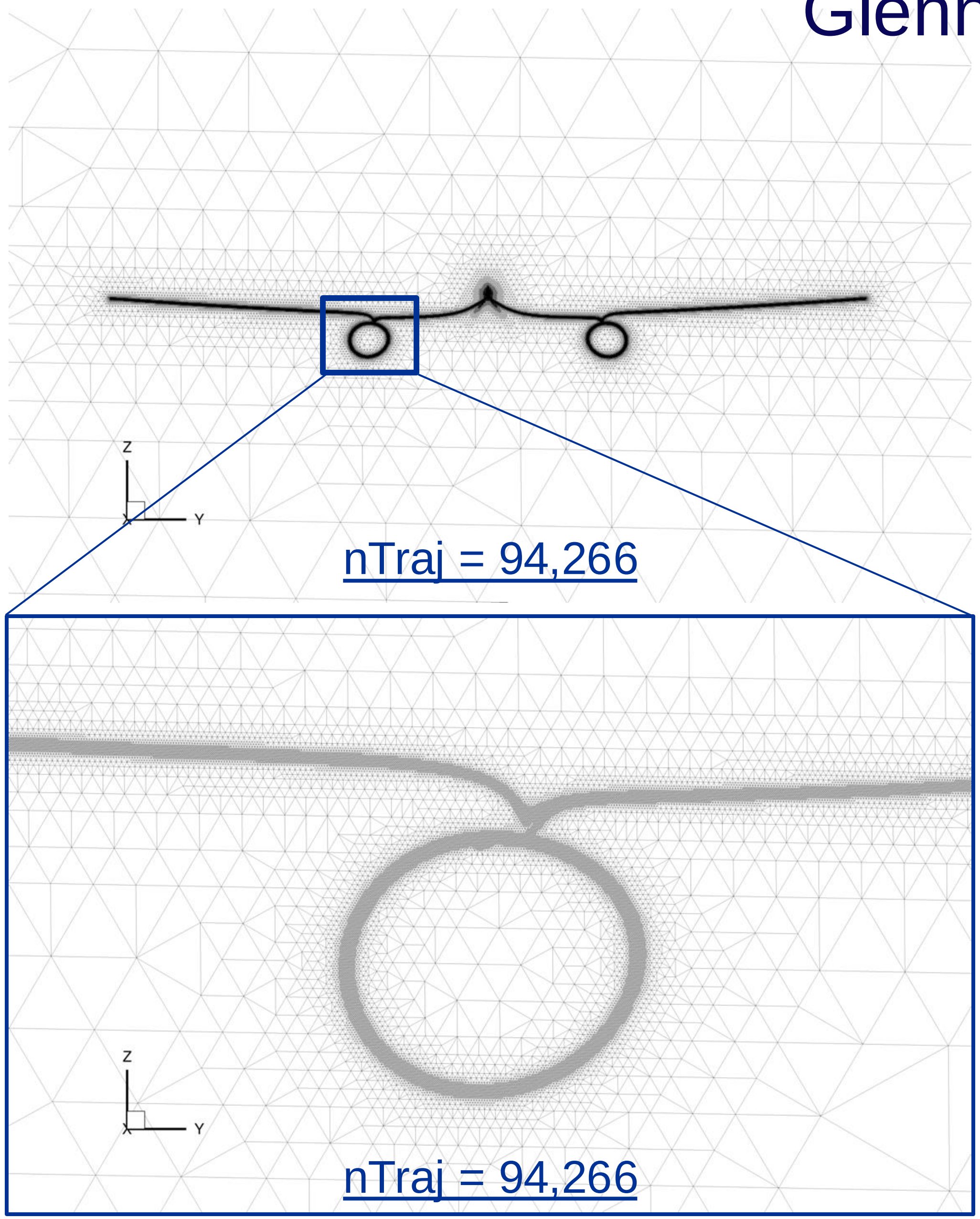


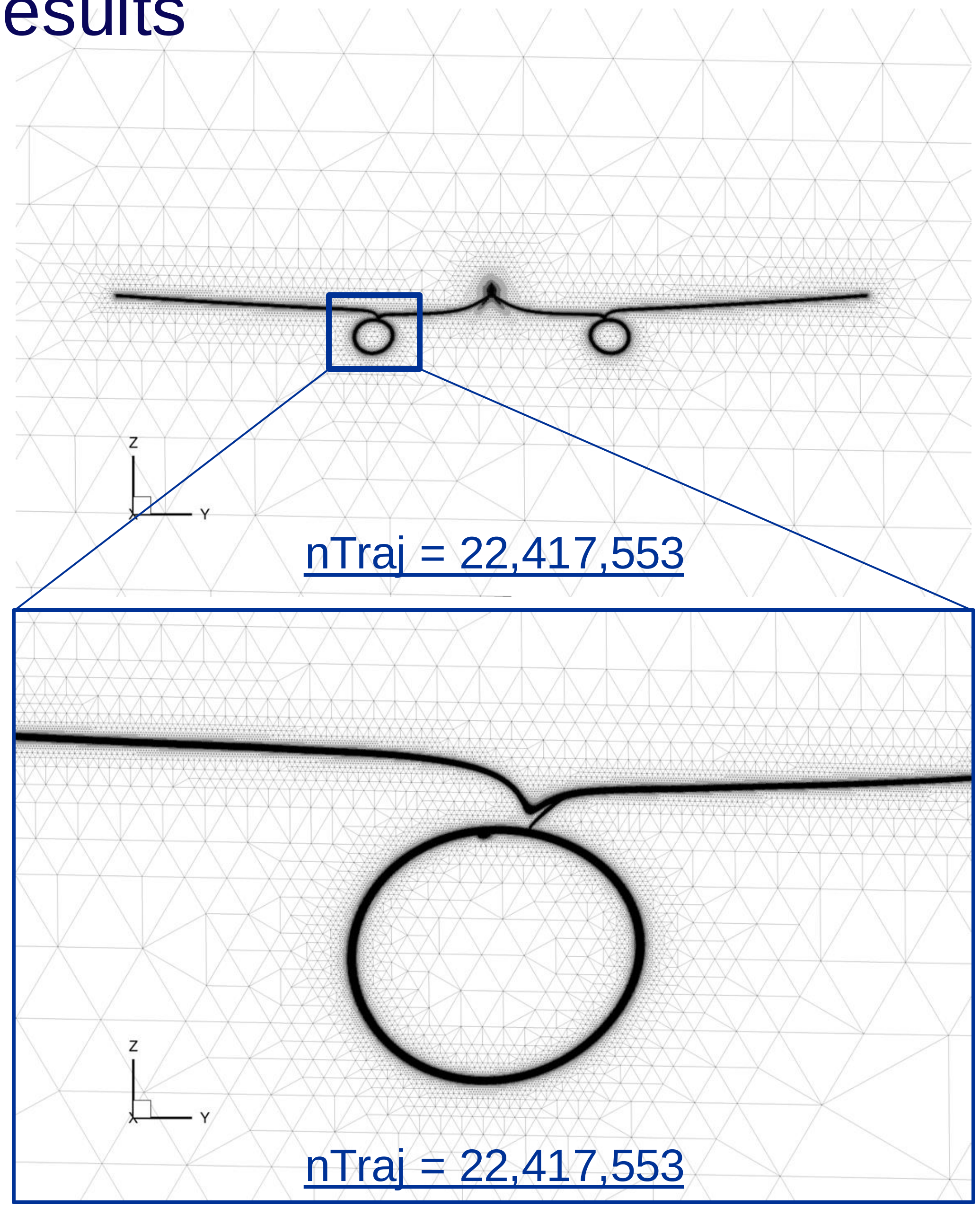
nTraj = 94,266



nTraj = 22,417,553

GlennICE Results



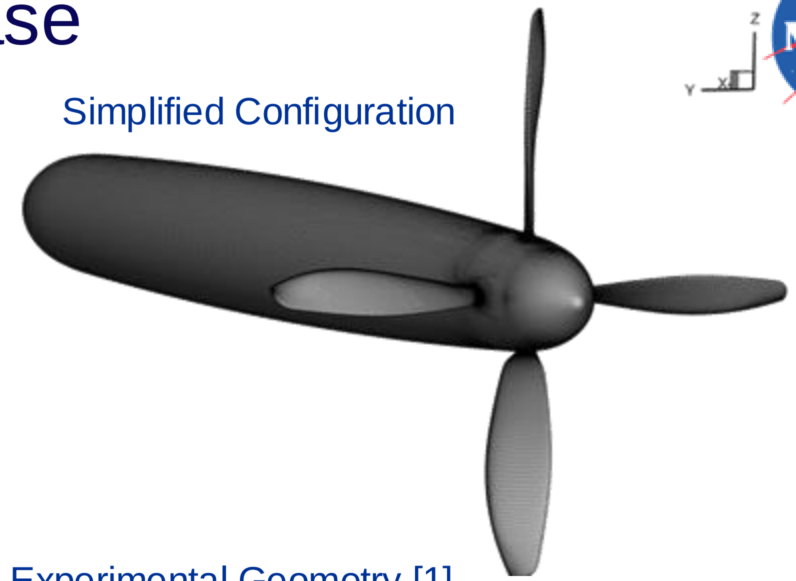


ARIES-I EXAMPLE CASE

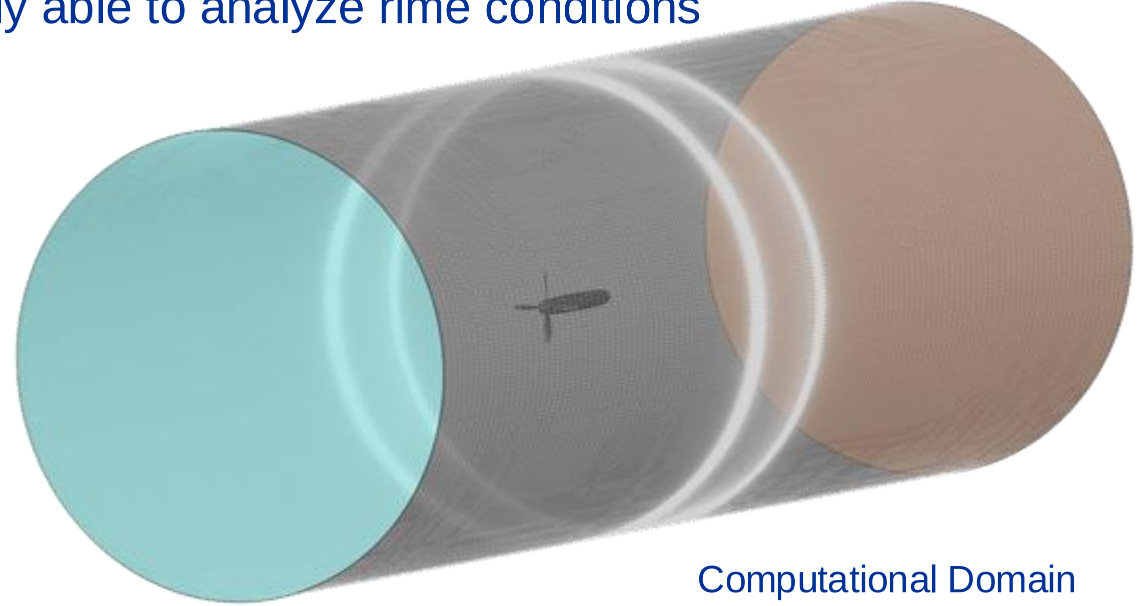
ARIES-I Example Case

- A simplified axisymmetric configuration of the NASA ARIES-I[1] geometry
- Demonstrate the rotational reference frame capabilities in GlennICE
- Allows for an analysis of propellor configurations typical of UAM/UAV configurations
- Currently utilizing an algebraic model to transfer unfrozen water
  - Only able to analyze rime conditions

Simplified Configuration



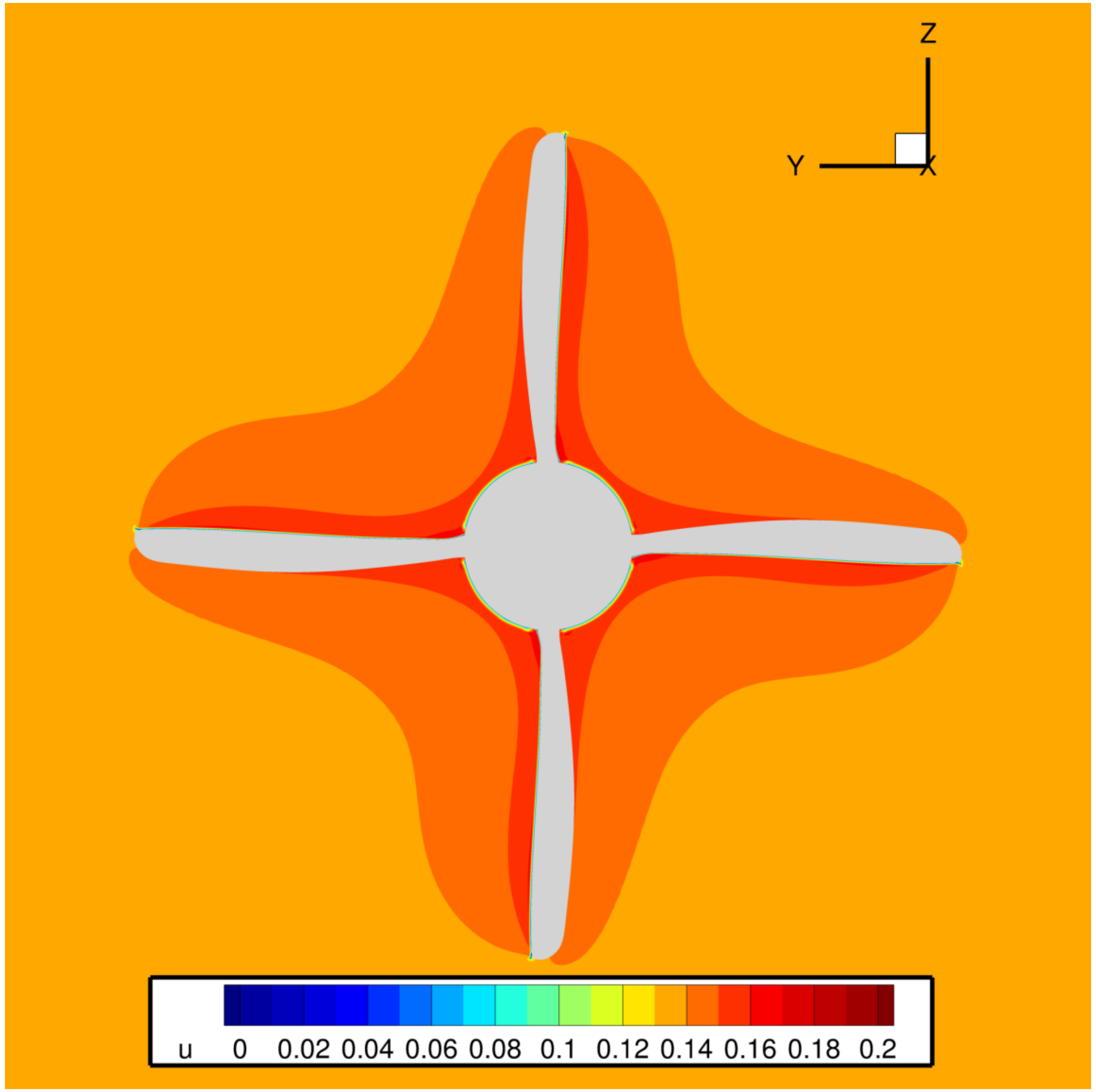
Experimental Geometry [1]



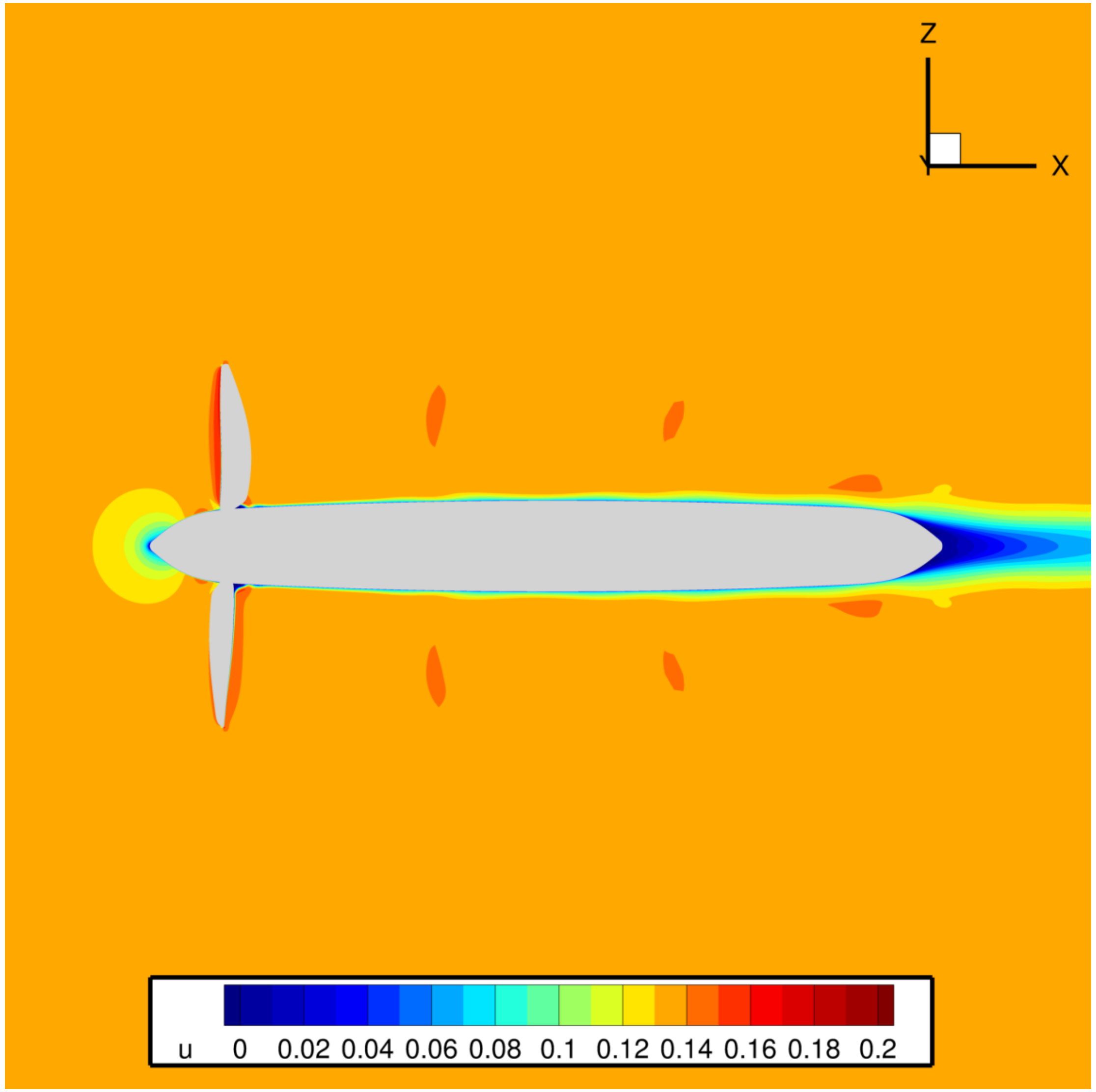
Computational Domain

[1] P.V. Hardenberg, et al., “Ice Shape Analysis of an eVTOL Propeller in Forward Flight at the NASA Glenn Icing Research Tunnel”, Aviation 2024

CFD Results



Front View of the Configuration

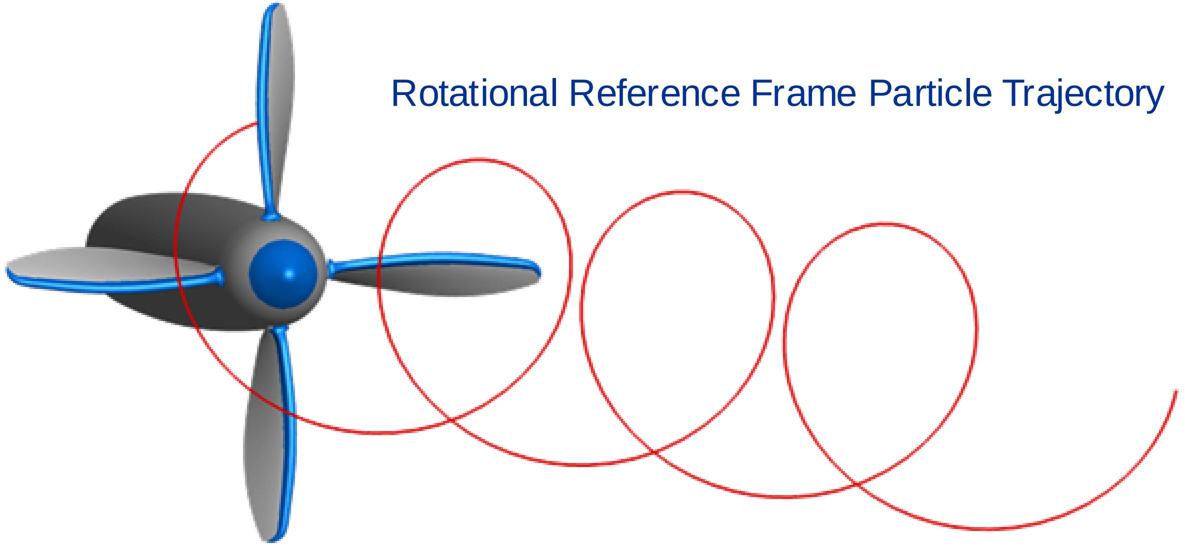


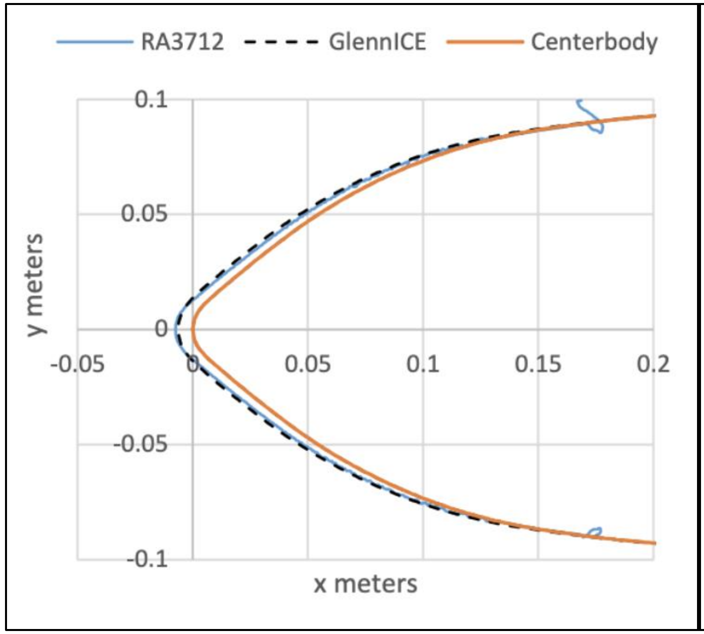
Side View of the Configuration

GlennICE Analysis

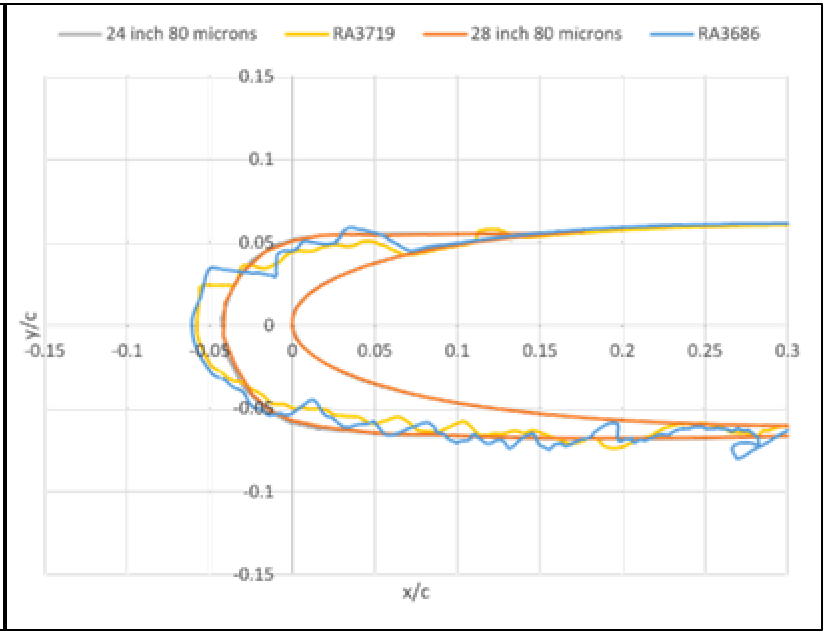
- Current regulations limit UAM configurations to minimal icing encounters
- Will help to enable UAM manufacturers to size IPS systems
- Will help to enable new technology, designs, and increase aviation safety
- Capable of translating this to next generation propulsion systems like open rotor

Rotational Reference Frame Particle Trajectory





Spinner Hub Comparison at MVD = 80μm [1]



Propellor Blade Comparison at MVD = 80μm [1]

[1] D. Rigby, P.V Hardenberg, “GlennICE Simulation of a 24, 28, and 36 inch Diameter eVOTL Propellers in Forward Flight”, Aviation 2024

FUTURE DEVELOPMENT

Future Technologies and Development

Active Developments

Mixing Planes
- Allows for radial distribution of particle release
- Enables internal analysis of engines
- Support design and certification of large bypass ratio engines
- Will compare to data obtained in PSL

Shallow Water Method (SWIM)
- Can account for Coriolis and centrifugal forces
- Enables rotational icing capabilities at glaze conditions
- Enabling technology for open-fan/open-rotor, UAM propellors, and engine configurations

Roughness Heat Transfer
- Heavily rely on empirical assumptions
- Most methods available for roughness modeling are not applicable to icing
- Strongly influences resultant ice shapes
- Will enable more accurate computational solutions

Future Developments

Multi-shot
- More tightly coupling GlennICE and a CFD solver
- Grow ice for a certain amount of time, recompute flowfield, recompute ice shape
- Generate more complex ice shapes

SLD – Splashing
- Secondary impingement of particles and particle breakup
- High interest by industry for enabling new technologies and certification of new platforms
- Crucial path forward for aircraft safety and sustainability

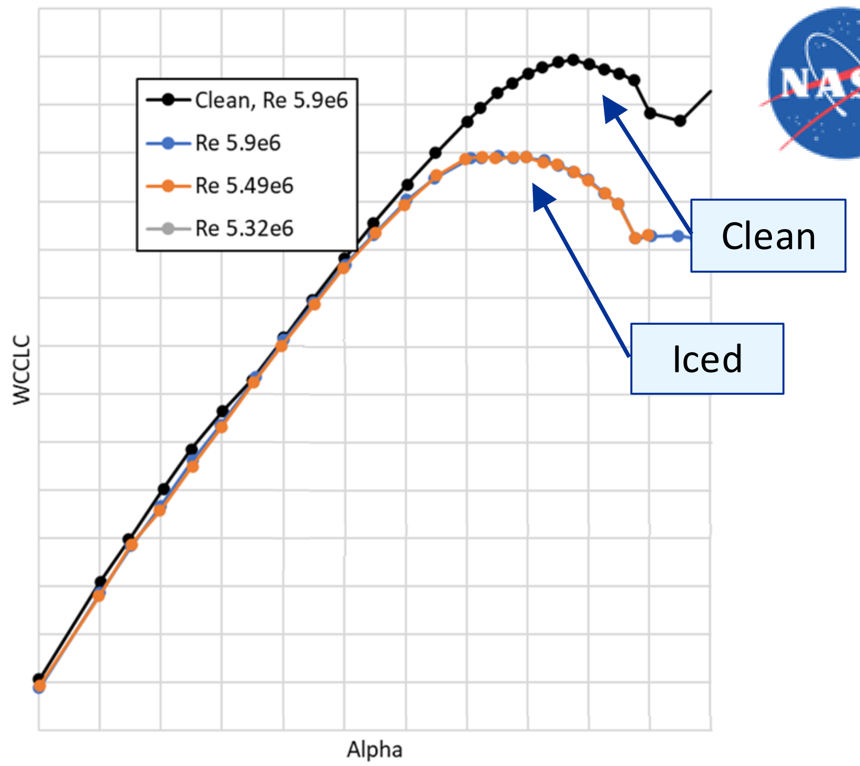
Ice Crystal Icing
- Enable melting and refreezing of accretion in an engine
- Cause of engine-out situations during flight
- Of interest to engine manufacturers
- Experimental data through SIDRAM available

ADDITIONAL WORK

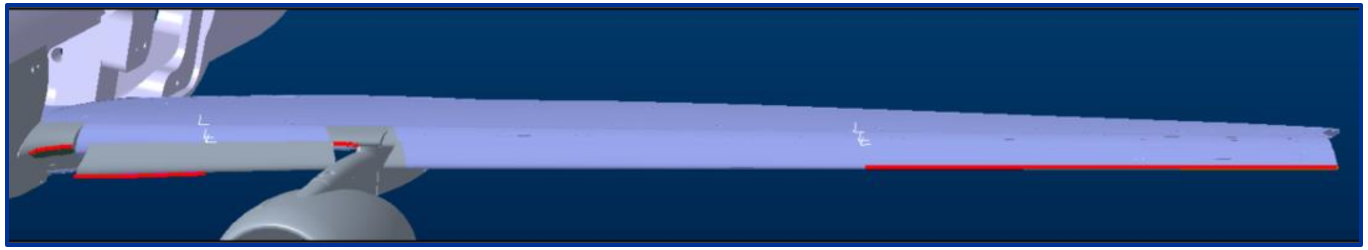
High Lift Common Research Model

Description of Technical Content

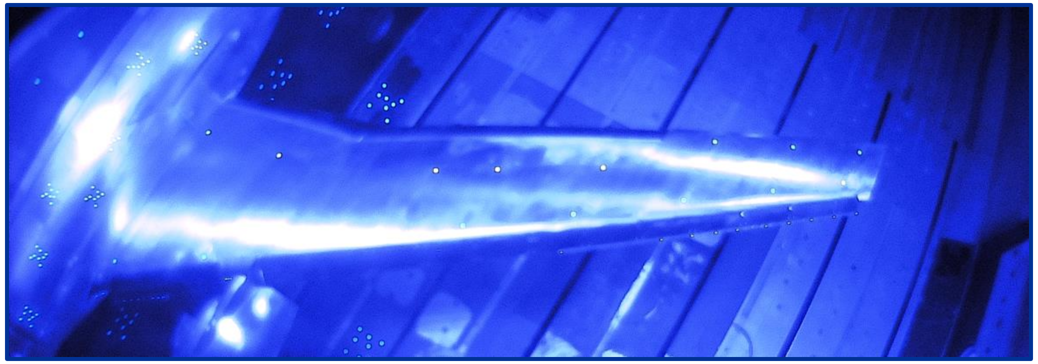
- Generation of an experimental database that fills two specific gaps in data, high lift and/or flight Reynolds number
- Subsequent usage of this data to benchmark computational tools to replace/reduce wind tunnel usage, lowering the cost to design and certify new vehicles
- In addition, there have been a variety of recent papers utilizing Wall Modeled Large Eddy Simulation on existing stowed straight and swept wing icing performance data



Performance data of the Clean and Iced CRM-HL configurations at the cross-facility Reynolds Numbers



Schematic of the artificial ice shapes (Red) on the High Lift Common Research Model



Usage of Ultraviolet (UV) light and UV sensitive paint for ice shape identification

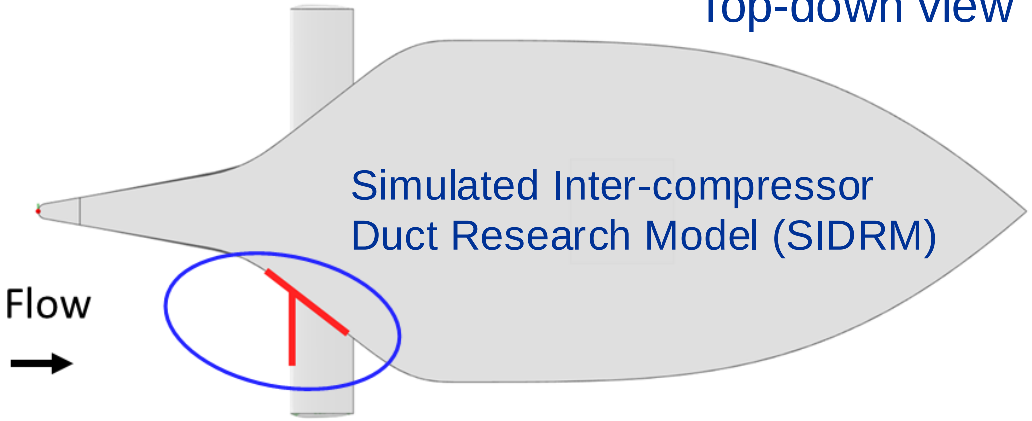
Transonic Truss Braced Wing Icing

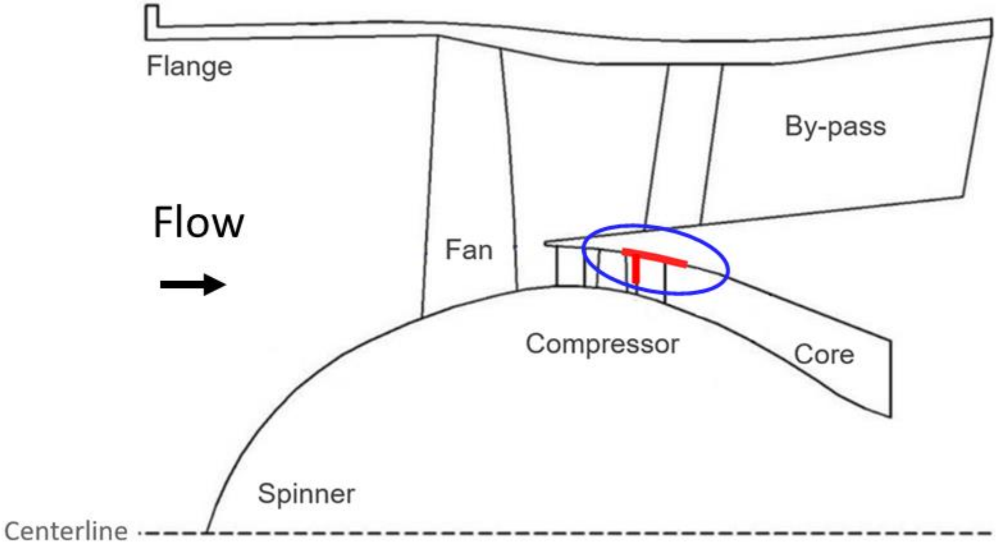
- Provide validated capability to assess impact of icing
- Assess potential impact of icing on fuel burn objective
- Develop potential materials that are both durable and icephobic
- Targeting IRT tests in FY24 and FY25

Engine Ice Accretion Testing – SIDRM

<u>Objectives:</u>
- Gather data to develop & validate computational icing tools for ice crystal (IC) icing
- Developed SIDRM Model
  - Simulates inner compressor duct & strut region
  - Ice crystal accretion using a heated surface

Top-down view





HOW DO I OBTAIN GlennICE?

Coming Soon to the NASA Software Store

We will be releasing on the Software Store in the next couple months

Will be available for US Citizens and Companies

https://www.software.nasa.gov

User Manual available at: https://ntrs.nasa.gov/citations/20240008077

Acknowlegdements

Thank you to the additional GlennICE development team members: Christopher Porter, Eric Galloway, David Rigby, William Wright, and Zaid Sabri.

NASA supported this research through the Transformative Tools and Technology project along with the The Advanced Air Transport Technology Project

These individuals and project are thanked for their contributions.

QUESTIONS?

Feel free to contact me at: thomas.ozoroski@nasa.gov