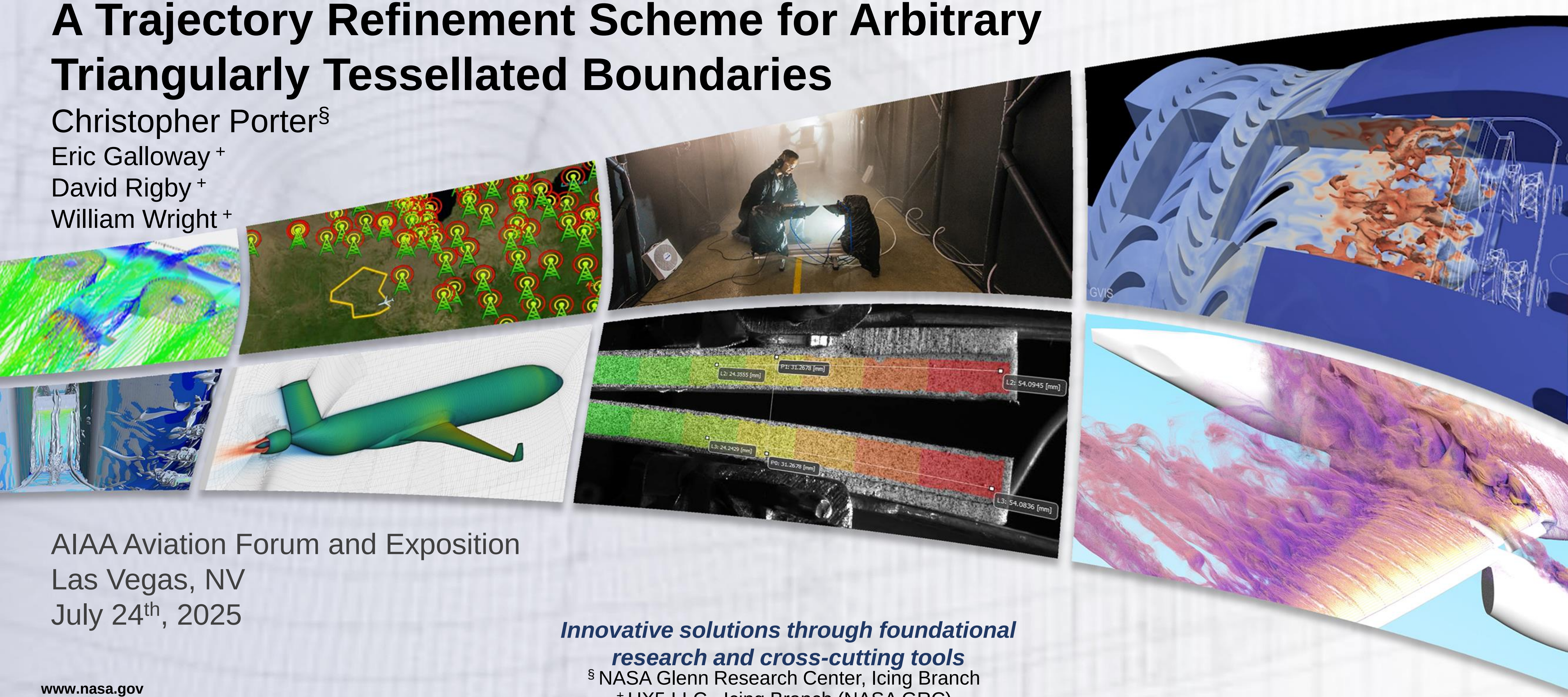
Transformational Tools and Technologies (T^3) Project

A Trajectory Refinement Scheme for Arbitrary Triangularly Tessellated Boundaries

Christopher Porter[§]
Eric Galloway [+]
David Rigby [+]
William Wright [+]



AIAA Aviation Forum and Exposition
Las Vegas, NV
July 24th, 2025

Innovative solutions through foundational research and cross-cutting tools

[§] NASA Glenn Research Center, Icing Branch
[+] HX5 LLC., Icing Branch (NASA GRC)

Acknowledgements

Thanks to NASA's Transformational Tools and Technologies (TTT) and Advanced Air Transport Technologies (AATT) Projects for the financial support of this effort.

Notice for Copyrighted Information

This is a joint work of employees of the National Aeronautics and Space Administration and employees of HX5, LLC. under Contract/Grant No. 80GRC020D0003 with the National Aeronautics and Space Administration. The United States Government may prepare derivative works, publish, or reproduce this manuscript and allow others to do so. Any publisher accepting this manuscript for publication acknowledges that the United States Government retains a non-exclusive, irrevocable, worldwide license to prepare derivative works, publish, or reproduce the published form of this manuscript, or allow others to do so, for United States government purposes.

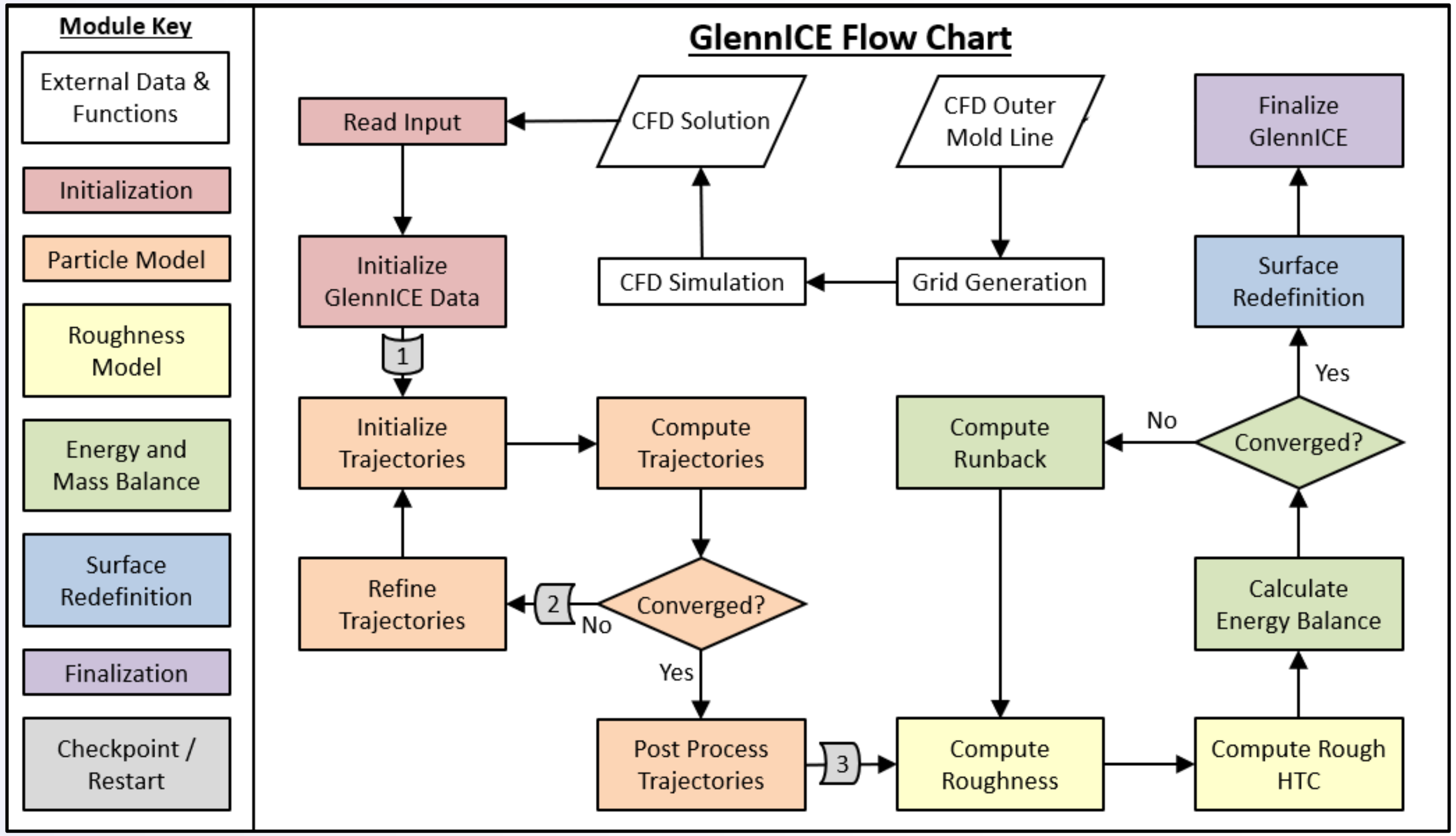
GlennICE

Flow Chart

GlennICE uses a discretized flow field generated from a computation fluid dynamics (CFD) solver.

Coupling between GlennICE and the CFD software is currently an entirely one-way coupling.

As such, GlennICE can be classified as a CFD post-processor.



Flow chart of the GlennICE software.

How to Add Efficiency?

Trajectory Parallelization

The ability to utilize multiple processors to reduce the wall clock time necessary to integrate many trajectories.

Adaptive Refinement

The ability to intelligently release and refine simulated trajectories to reduce the computational cost by avoiding the computation of unnecessary work to achieve a desired simulation accuracy.

Intelligent β Schemes

The ability to post process the results from the trajectory computation to determine local collection efficiency with fewer trajectories than required for a naïve construction.

Legacy Refinement Method

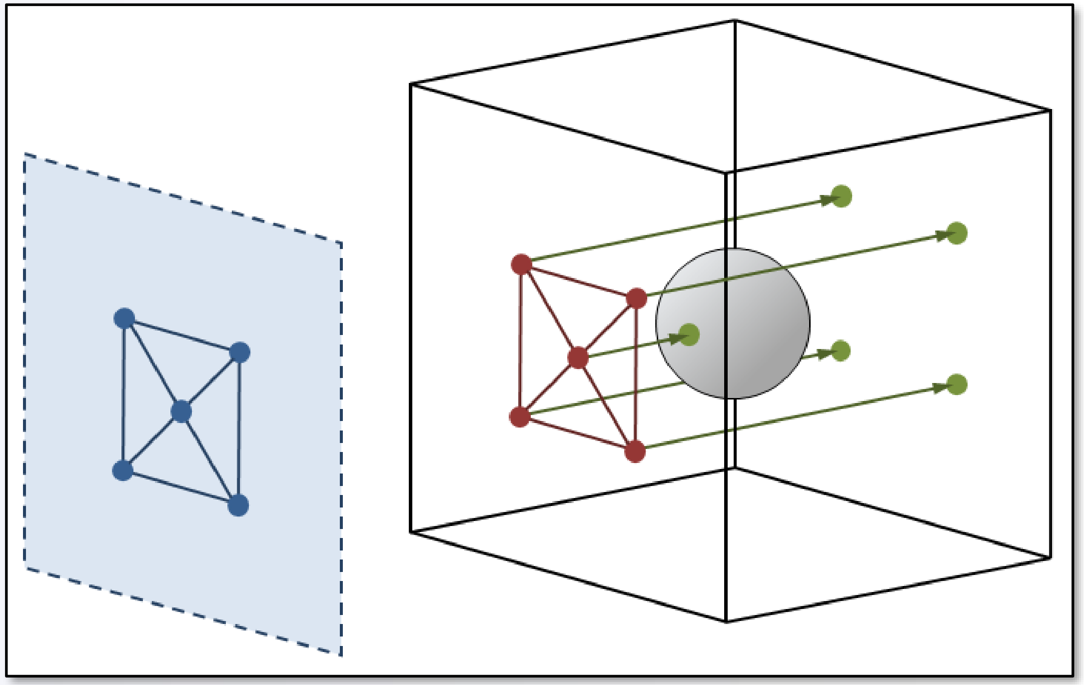
2.5D Delaunay

GlennICE's legacy refinement scheme did not compute the refinement directly on the boundary.

Instead refinement was performed on a fictious plane upstream (blue) and projected onto the boundary (red).

This is sufficient for simple freestream inlets, but isn't extensible to complex inlets such as annular inlets, or reinjection of bounced / broken / splashed droplets or crystals from an aerodynamic body.

To address these more complex configurations, a more general method is necessary.



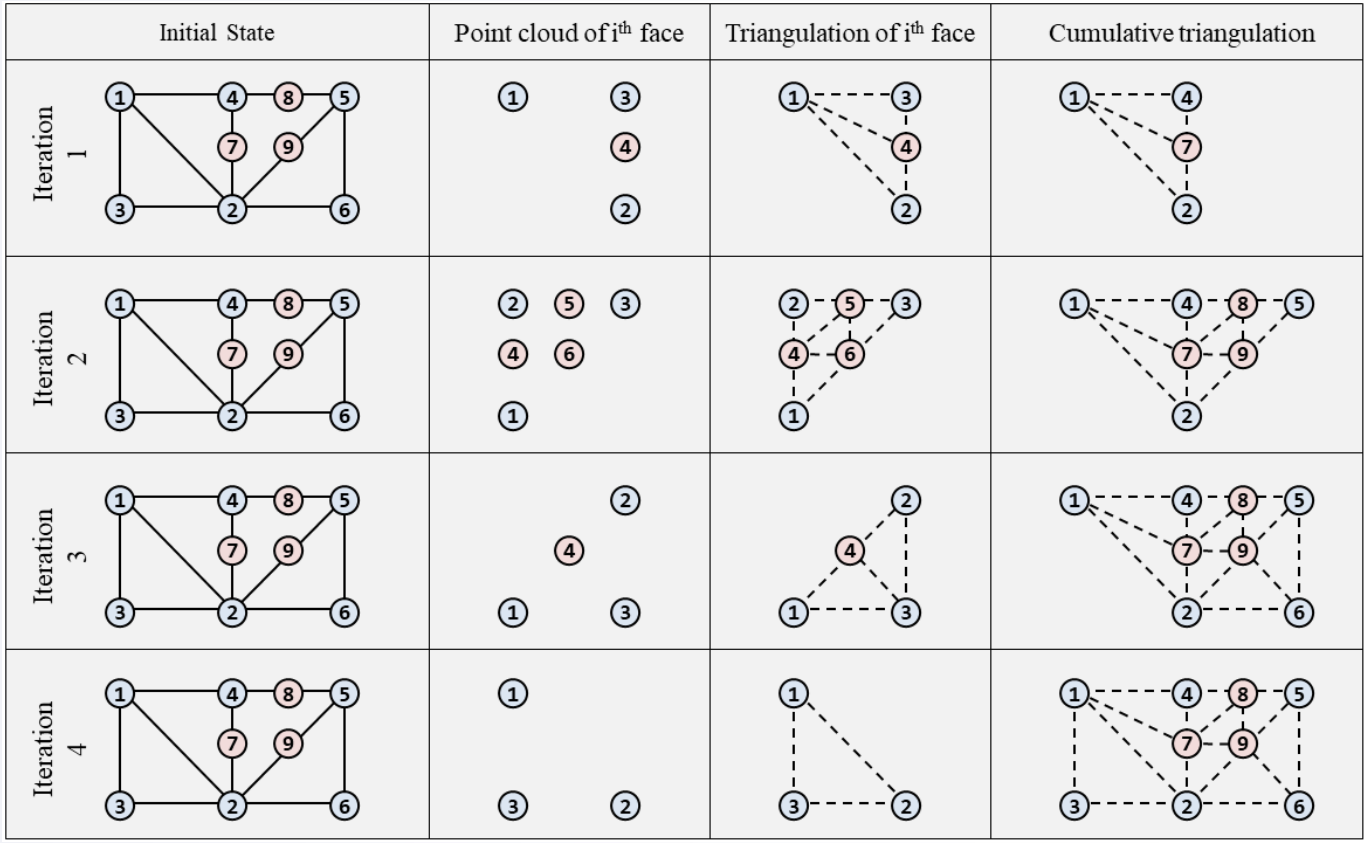
Schematic of GlennICE's legacy refinement methodology. The fictitious two-dimensional seed plane is depicted in blue, and the resulting refinement on the inlet boundary depicted in red.

Point Injection and Mesh Generation

Arbitrary Tessellated Surface

For each face of the inlet boundary a two-dimensional Delaunay triangulation is performed. Thus, the number of iterations of this meshing process is equivalent to the number of faces of the underlying inlet boundary. On each iteration the following set of steps are performed:

1. The release points coincident with face ‘i‘ of the inlet boundary are extracted and the indexing of these release points is transformed from a global indexing to a local indexing for these subset of points.
2. These points in three-dimensional space are rotated to the yz plane.
3. A two-dimensional Delaunay triangulation is performed.
4. The local indexing of the subset of release points are transformed back to the global indexing of the total set.
5. The resulting triangulated release points are added to a cumulative list of triangulated release points.



Tabular schematic illustrating the meshing algorithm. Nodal indexes in the first and fourth columns are globally indexed, and the second and third columns are locally indexed. Blue colored nodes denote the initial release points, while red colored nodes denote new release points added by GlennICE’s refinement algorithms.

Delaunay Robustness

Barycentric Distortion

Robustness issues can arise when performing the Delaunay triangulation due to:

1. Colinear points on the convex hull
2. Highly anisotropic point clouds

A pre-processing distortion is applied to the point cloud prior to the triangulation to increase robustness.

Eq. 1 - Cartesian to Barycentric

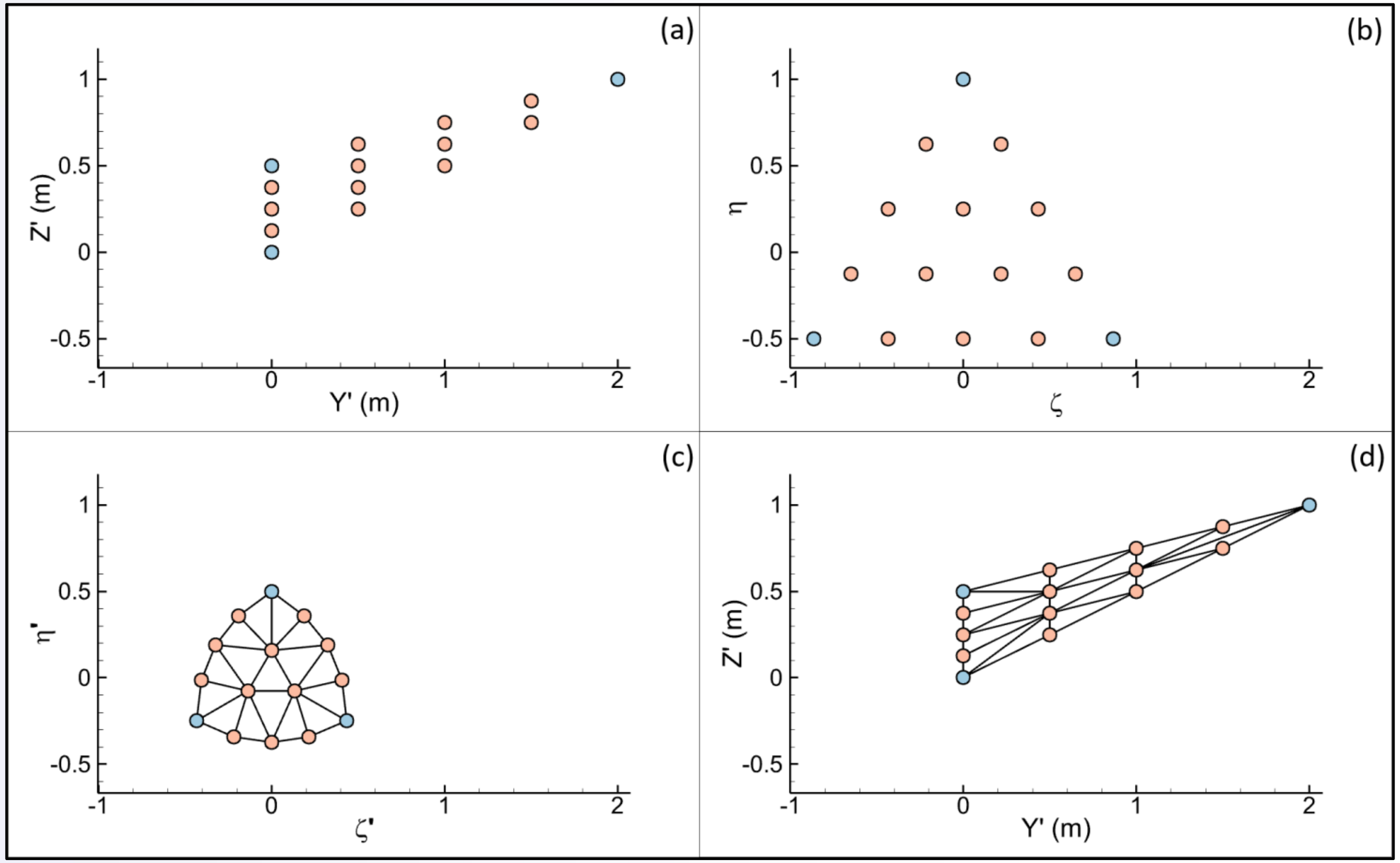
$$\begin{bmatrix} u_i \\ v_i \\ w_i \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ p_{1Y'} & p_{2Y'} & p_{3Y'} \\ p_{1Z'} & p_{2Z'} & p_{3Z'} \end{bmatrix}^{-1} \cdot \begin{bmatrix} 1 \\ p_{iY'} \\ p_{iZ'} \end{bmatrix}$$

Eq. 2 - Barycentric Distortion 1

$$\begin{bmatrix} p_{i\eta} \\ p_{i\zeta} \end{bmatrix} = \begin{bmatrix} 0 & -\frac{\sqrt{3}}{2} & \frac{\sqrt{3}}{2} \\ 1 & -\frac{1}{2} & -\frac{1}{2} \end{bmatrix} \cdot \begin{bmatrix} u_i \\ v_i \\ w_i \end{bmatrix}$$

Eq. 3 - Barycentric Distortion 1 & 2

$$\begin{bmatrix} p_{i\eta'} \\ p_{i\zeta'} \end{bmatrix} = \begin{bmatrix} 0 & -\frac{\sqrt{3}}{2} & \frac{\sqrt{3}}{2} \\ 1 & -\frac{1}{2} & -\frac{1}{2} \end{bmatrix} \cdot \begin{bmatrix} u_i - \gamma u_i^2 \\ v_i - \gamma v_i^2 \\ w_i - \gamma w_i^2 \end{bmatrix}$$



Depiction of the triangulation workflow in GlennICE's refinement algorithm. Blue colored nodes denote the initial release points, while red colored nodes denote new release points added by GlennICE's refinement algorithms.

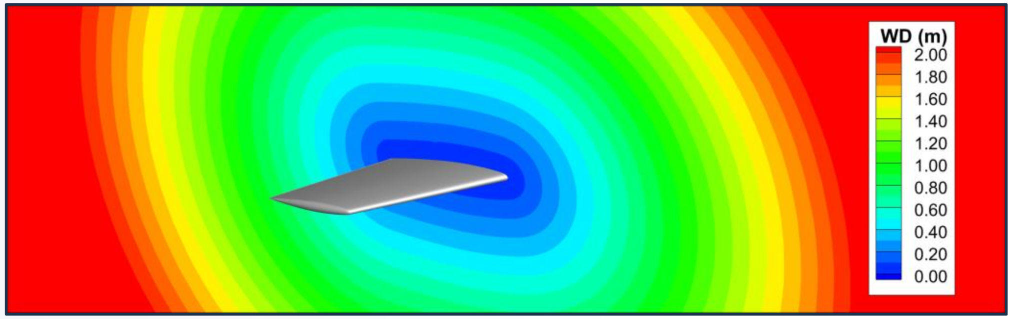
Adaptive Refinement

Feature Finding

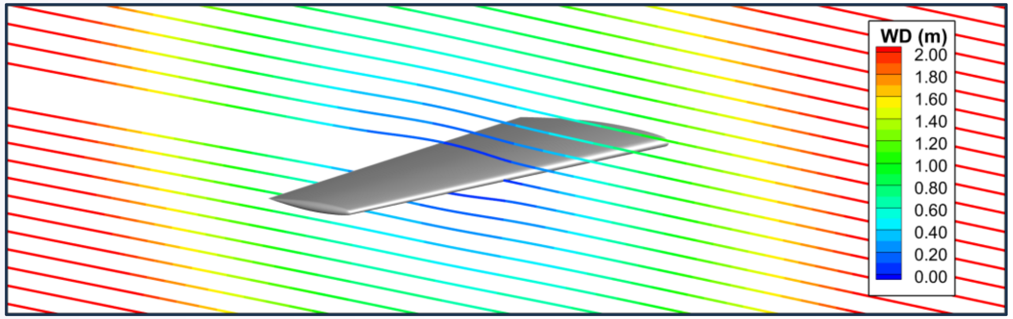
GlennICE’s feature finding algorithm relies on the determination of the “Minimum Wall Distance” for each trajectory. The Minimum Wall Distance is a measure of how close a trajectory came to a surface of interest.

This value is determined by extracting the Wall Distance at each timestep along a trajectory and extracting the minimum value of wall distance across the entire trajectory.

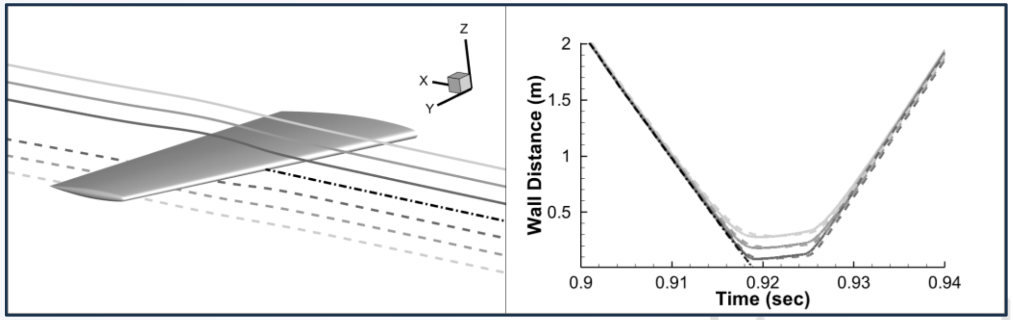
GlennICE computes the Wall Distance scalar field internally using Wigton’s Fast-Distance Calculation.[1] This allows a GlennICE user to choose which computational surfaces they want to impinge.



The wall distance scalar field at the 0.75m spanwise location on the ONERA M6 wing.



Values of wall distance along a rake of trajectories.



The wall distance as a function of time (right) for a subset of trajectories denoted graphically (left).

---

[1] Wigton, L., “Research in Computational Aeroscience Applications Implemented on Advanced Parallel Computing Systems,” NASA/CR-96-206062, 1996.

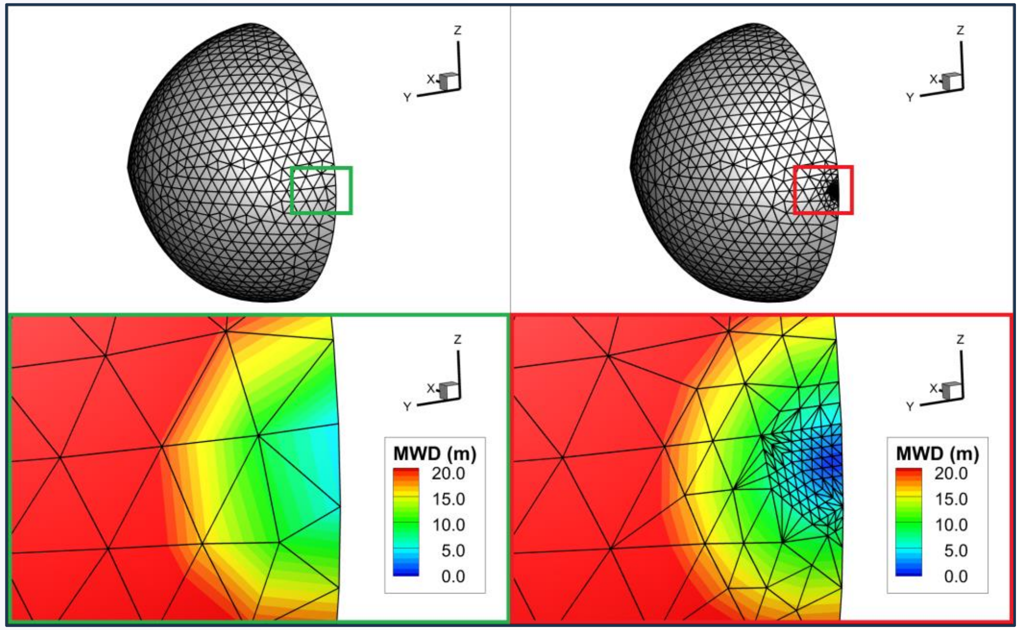
Feature Finding

Freestream Inlet Example

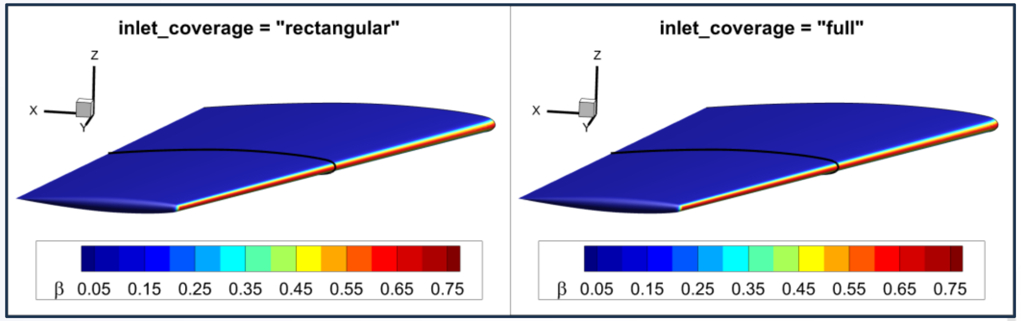
The feature finding algorithm is quite simple, effective, and robust. If the spacing between neighboring trajectories is greater than the Minimum Wall Distance of either trajectory, a new trajectory is computed between the two neighbors.

To say it another way, the local grid refinement is proportional to the local Minimum Wall Distance.

This feature finding methodology is not self terminating. It will continuously refine the impingement limit region. An additional metric, `hit_percent_limit`, was created to terminate the feature finding algorithm. This metric looks at the ratio of trajectories that hit the surface of interest compared with the total number of trajectories simulated. Once GlennICE achieves a certain ratio, the feature finding algorithm terminates.



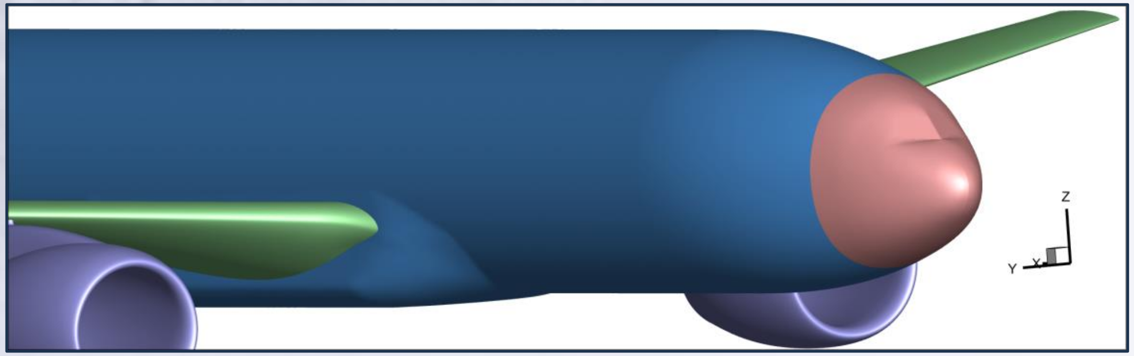
Refinement for Iteration 1 (left) and Iteration 4 (right) of the ONERA M6 example case.



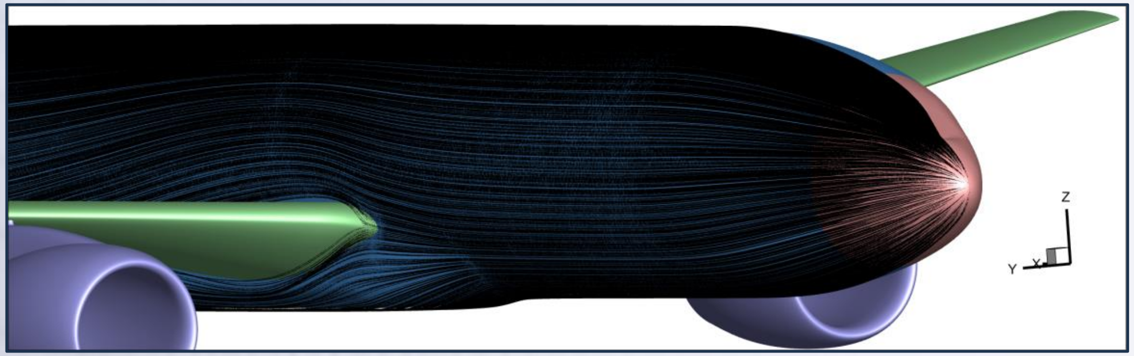
Comparisons of collection efficiency on the ONERA M6 wing using GlennICE's legacy refinement method (left) and new refinement method (right).

Point Injection and Mesh Generation

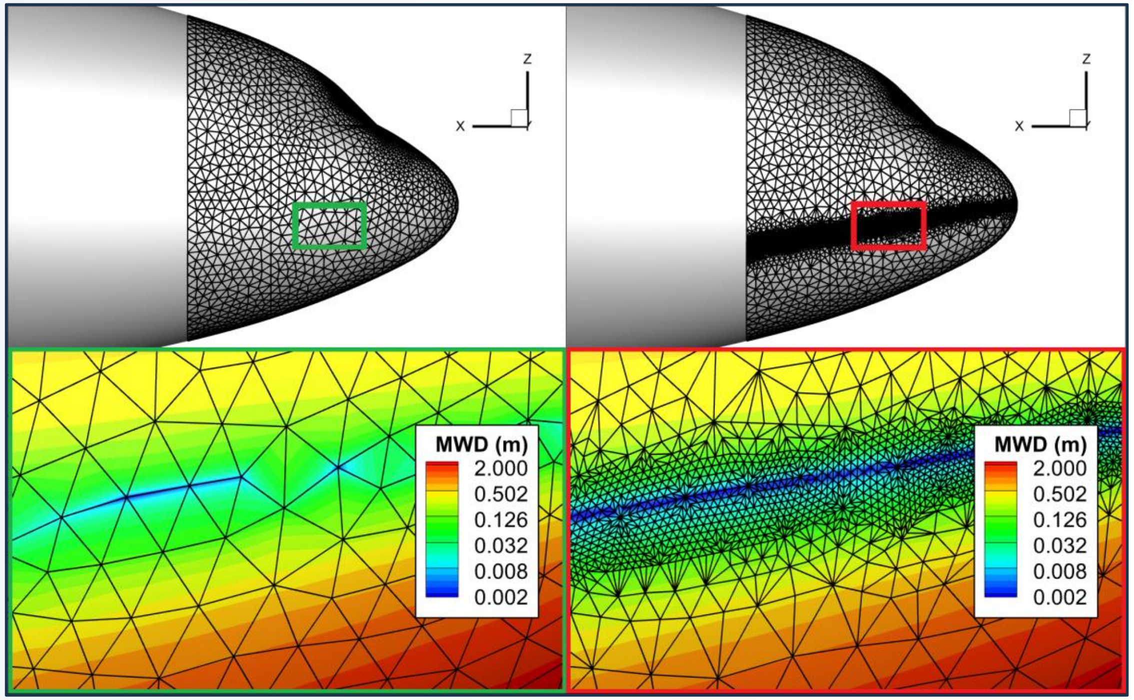
Generality



Computational surfaces of the CRM-HL simulation. (Pink - Nose, Blue - Fuselage, Green - Wing, Purple - Pylon/Nacelle)



The first iteration of trajectories released from the nose surface of the CRM-HL.



Release mesh for the nose of the CRM-HL after one (left) and four (right) refinement iterations.

Summary

NASA has prioritized the development of Lagrangian based computational ice accretion tools to more accurately model the physical interaction between a dispersed cloud and an aerodynamic vehicle.

Advanced algorithms are required to enable Lagrangian based methods to be used as an engineering design and analysis tool.

NASA has developed, or is in the process of developing, algorithms that significantly reduce the wall clock time and/or computational cost required to predict the impingement of water using Lagrangian methods.