

**AERODYNAMIC LOADS OVER ARBITRARY BODIES BY METHOD OF  
INTEGRATED CIRCULATION**

by

Vivek Ahuja

A dissertation submitted to the Graduate Faculty of  
Auburn University  
in partial fulfillment of the  
requirements for the Degree of  
Doctor of Philosophy

Auburn, Alabama  
August 3, 2013

Keywords: Potential-flow, Unstructured mesh, aerodynamic loads, Vorticity based drag prediction, circulation based loads, aerodynamic optimization

Copyright 2013 by Vivek Ahuja

Approved by

Roy Hartfield, Chair, Woltosz Professor, Department of Aerospace Engineering  
John Burkhalter, Co-chair, Professor emeritus, Department of Aerospace Engineering  
Andrew Shelton, Assistant Professor, Department of Aerospace Engineering

## ABSTRACT

A method for the application of vorticity based potential-flow solvers to unstructured surface meshes has been created. This method is designed to maintain the advantages of the vorticity based solvers while removing the limitations of geometrical inputs associated with the philosophical approach. A discussion on the necessity and advantages of this approach has been presented. A modification to the evaluation of skin-friction coefficients using surface vorticity has also been developed. An unstructured wake-strand model has been developed to allow handling of wakes emanating from unstructured meshes. An attempt has been made to extend potential-flow solvers to the current industry surface meshing and numerical solver standards.

This automated pipeline is considered to be robust enough to allow integration with optimizers unrestricted in their geometrical design spaces. A test-suite of basic shapes and bodies was tested to evaluate the fidelity of the new approach. An advanced test-suite was developed to stress-test the solver as well as to test out the geometry handling pipeline required to handle these test cases. It was determined that the solver is able to provide high fidelity results for a wide variety of test cases and is able to: work with conformal and non-conformal geometry interfaces, resolve surface intersections, work with both structured and unstructured meshes, work with both thick and thin bodies and work with manifold and non-manifold surface interfaces.

## ACKNOWLEDGMENTS

The author would like to thank Dr. Roy Hartfield for providing the encouragement, support and a free-thinking environment for conducting this research. Thanks are also due to Dr. John Burkhalter and Dr. Andrew Shelton for helping develop the fundamental theories used in this work and to Dr. Mark Carpenter for providing an objective overview of the work as an external reader of this document. Thanks are also due to my mother, Dr. Sulekha Ahuja, for providing that metaphorical wall of support to rest on during these past few wonderful years of work.

Style manual or journal used:

American Institute of Aeronautics and Astronautics Journal

Computer software used:

TECPLOT 10/360, FORTRAN

## TABLE OF CONTENTS

ABSTRACT.....	ii
ACKNOWLEDGMENTS .....	iii
STYLE MANUAL.....	iv
LIST OF ILLUSTRATIONS.....	vii
LIST OF TABLES.....	xiv
NOMENCLATURE .....	xv
1. INTRODUCTION .....	1
2. MOTIVATION .....	13
3. THE NEED FOR UNSTRUCTURED MESHES .....	16
4. SOLVER METHODOLOGY.....	26
4.1 Spatial velocity inductions.....	26
4.2 Solver philosophy .....	30
4.3 Vorticity wakes .....	33
4.4 Aerodynamic loads .....	41
4.4.1 Method of Integrated Circulation.....	41
4.4.2 Skin friction drag evaluations .....	50
5. MESHER METHODOLOGY .....	53
5.1 Size based meshing .....	53
5.2 Mesh and facet quality .....	59
5.3 Mesh auto-refinement .....	61
5.4 Patch interface and conformity .....	69

6. SOLVER EFFICIENCY ENHANCEMENT .....	73
6.1 Fluid-Dynamic-Mesh-Repartitioning .....	73
6.2 Persistent Geometrical Induction matrix .....	77
7. BOUNDARY PHYSICS .....	79
7.1 Slip walls.....	79
7.2 Velocity inlets .....	80
7.3 Trailing edges.....	83
7.4 Symmetry planes.....	87
8. RESULTS AND ANALYSIS.....	90
8.1 Basic geometry test suite .....	91
8.1.1 Rectangle wings.....	91
8.1.2 Delta wings .....	93
8.1.3 Double-delta 75° / 65° .....	95
8.1.4 Arrow wing 70° .....	97
8.1.5 Diamond wing 70° .....	99
8.1.6 Sweep wing 45° .....	101
8.1.7 Unit sphere and axisymmetric bodies .....	103
8.2 Advanced geometry test suite .....	106
8.2.1 F-16A .....	107
8.2.2 F-18A .....	114
8.2.3 DLR-F6.....	121
8.2.4 NASA Canard Fighter.....	128
8.2.5 Rutan Varieze.....	136
9. CONCLUSIONS .....	143
REFERENCES .....	148
APPENDIX.....	155

## LIST OF ILLUSTRATIONS

Figure 3.1: A typical structured mesh over a high curvature surface .....	18
Figure 3.2: Solver run time data versus mesh size for various geometries.....	19
Figure 3.3: Definition of manifold and non-manifold mesh surfaces.....	21
Figure 3.4: Case study sphere in structured and unstructured environments .....	24
Figure 4.1.1: The mapped vortex ring for a facet on the surface mesh .....	27
Figure 4.1.2: Velocity induced at a point by a segment of a facet-bound vortex ring.....	28
Figure 4.3.1: The wake-strand model at the trailing edge .....	35
Figure 4.3.2: Vorticity decay rates and velocity desingularisation.....	37
Figure 4.3.3: Adaptive pseudo-time stepping in the relaxed strand theory .....	39
Figure 4.3.4: Local physics driven wake strands .....	40
Figure 4.3.5: Relaxed wake strands on bodies of revolution .....	41
Figure 4.4.1: Concept of integrated circulation .....	42
Figure 4.4.2: Integrated circulation loops around a sample geometry.....	43

Figure 4.4.3: The elliptical wing vorticity and wake solution .....	45
Figure 4.4.4: Integrated circulation loops around the elliptical wing .....	45
Figure 4.4.5: Composite span-wise sectional lift coefficient for the elliptical wing .....	46
Figure 4.4.6: Integrated circulation loops around the F-18A geometry .....	47
Figure 4.4.7: Composite span-wise sectional lift coefficient for the F-18A.....	48
Figure 4.4.8: Span-wise integrated circulation for rectangle wing (AR=4, 12° AOA) .....	49
Figure 4.4.9: Lift coefficient as function of spacing between ICLs for rectangle wing .....	49
Figure 4.4.10: Topological “walking” used to determine B.L. growth length .....	51
Figure 4.4.11(a): Vorticity distribution for the Bell-Boeing tilt-rotor (5° AOA).....	52
Figure 4.4.11(b): Skin-friction coefficient for the Bell-Boeing tilt-rotor (5° AOA) .....	52
Figure 5.1.1: Size based meshing case study for a truncated cone surface .....	56
Figure 5.1.2(a): Initial pressure-solver mesh for the Varieze (108,040 facets) .....	58
Figure 5.1.2(b): Re-tessellated vorticity-Solver mesh for the Varieze (10,036 facets) .....	58
Figure 5.1.3: Mesh sizing distribution and gradients.....	59
Figure 5.2.1: Unstructured mesh facet quality distributions.....	61
Figure 5.3.1: Edge collapsing on below-threshold facets. Before (left), after (right).....	62
Figure 5.3.2: Edge swapping on facet pairs, before (left) and after (right) .....	63



Figure 5.3.3: Individual facet-splitting, before (left) and after (right).....	64
Figure 5.3.4: Patch-splitting, before (left) and after (right) .....	65
Figure 5.3.5: Piercing correction, before and after (cutting surface hidden).....	66
Figure 5.3.6: F-16A initial geometry pre-solver topological probing results.....	68
Figure 5.3.7: F-16A case study results for automated mesh refinements.....	68
Figure 5.4.1: Mig-21 mesh conformity case study results.....	71
Figure 5.4.2: Mig-21 case study convergence history results.....	72
Figure 6.1.1: Rationale behind FDMR .....	74
Figure 6.1.2: The F-35 geometry before and after FDMR .....	76
Figure 6.1.3: Convergence history for the F-35 solution (AOA = 5°, 1 m/sec flow).....	77
Figure 7.2.1: The F-18 geometry with velocity inlets and symmetry plane .....	82
Figure 7.2.2: The F-18 geometry with no Neumann relaxation .....	82
Figure 7.2.3: The F-18 geometry with Neumann relaxation .....	82
Figure 7.3.1: Condition-1 for auto-detecting “thick” trailing edges marked by the solver .....	84
Figure 7.3.2: Centrifugal normal vectors for a general facet.....	85
Figure 7.3.3: Typical results of trailing edge auto-detection .....	85
Figure 7.3.4: The F-18 geometry manifold and non-manifold surfaces.....	86

Figure 7.4.1: The DLR-F6 geometry with and without symmetry BC.....	88
Figure 7.4.2: The DLR-F6 geometry with symmetry BC (AOA = 0°, 1 m/sec flow).....	88
Figure 8.1.1: Lift coefficient versus angle of attack for rectangle wings .....	92
Figure 8.1.2: Lift versus Drag for rectangle wings.....	93
Figure 8.1.3: Lift coefficient versus angle of attack for delta wings .....	95
Figure 8.1.4: Unstructured mesh on the 75°/65° double-delta .....	96
Figure 8.1.5: Lift coefficient versus angle-of-attack for 75°/65° double-delta .....	97
Figure 8.1.6: Unstructured mesh on the 70° arrow wing.....	98
Figure 8.1.7: Lift coefficient versus angle-of-attack for the 70° arrow wing.....	99
Figure 8.1.8: Unstructured mesh on the 70° diamond wing .....	98
Figure 8.1.9: Lift coefficient versus angle-of-attack for the 70° diamond wing .....	100
Figure 8.1.10: Lift coefficient versus angle-of-attack for the 45° sweep wing .....	102
Figure 8.1.11: Normalized sectional lift coefficient for the 45° sweep wing.....	103
Figure 8.1.12: The unit sphere .....	104
Figure 8.1.13: Lift coefficient versus angle of attack for the sphere .....	104
Figure 8.1.14: The ogive axisymmetric body .....	105
Figure 8.1.15: Lift coefficient versus angle of attack for the ogive body.....	106

Figure 8.2.1: The Lockheed-Martin F-16 .....	107
Figure 8.2.2: Original geometry for the F-16A post CAD design .....	108
Figure 8.2.3: Auto-mesher results on the F-16A .....	110
Figure 8.2.4: Vorticity distribution on the F-16A.....	111
Figure 8.2.5: ICL distribution about the symmetry plane on the F-16A .....	112
Figure 8.2.6: Lift coefficient span-wise distribution about the symmetry plane .....	112
Figure 8.2.7: Lift coefficient versus angle of attack for the F-16A .....	113
Figure 8.2.8: Lift versus drag for the F-16A.....	114
Figure 8.2.9: The McDonnell Douglas F-18.....	115
Figure 8.2.10: Auto-mesher results for the F-18A.....	115
Figure 8.2.11: Vorticity distribution on the F-18A.....	116
Figure 8.2.12: ICL distribution about the symmetry plane on the F-18A .....	117
Figure 8.2.13: Lift coefficient span-wise distribution about the symmetry plane .....	118
Figure 8.2.14(a): Panel model of the McDonnell Douglas F-18A with 1,336 panels .....	119
Figure 8.2.14(b): Unstructured model of the McDonnell Douglas F-18A with 7,853 facets...	119
Figure 8.2.15: Lift coefficient versus angle of attack for the F-18A .....	120
Figure 8.2.16: Lift versus drag for the F-18A.....	121

Figure 8.2.17: The DLR-F6 geometry on a mount in the DLR wind-tunnel.....	122
Figure 8.2.18: Initial AIAA CAD tessellations for the DLR-F6 .....	123
Figure 8.2.19: Auto-mesher results for the DLR-F6.....	123
Figure 8.2.20: Vorticity distribution on the DLR-F6.....	124
Figure 8.2.21: ICL distribution about the symmetry plane on the DLR-F6 .....	125
Figure 8.2.22: Lift coefficient span-wise distribution about the symmetry plane .....	126
Figure 8.2.23: Lift coefficient versus angle of attack for the DLR-F6 .....	127
Figure 8.2.24: Lift versus drag for the DLR-F6.....	128
Figure 8.2.25: The Canard Fighter model on a mount at NASA Ames tunnel.....	130
Figure 8.2.26: Auto-mesher results for the Canard Fighter post CAD design .....	130
Figure 8.2.27: Vorticity distribution on the Canard Fighter .....	132
Figure 8.2.28: ICL distribution about the symmetry plane.....	133
Figure 8.2.29: Lift coefficient span-wise distribution about the symmetry plane .....	133
Figure 8.2.30: Lift coefficient versus angle of attack for the Canard Fighter .....	135
Figure 8.2.31: Lift versus angle of attack for the Canard Fighter.....	136
Figure 8.2.32: The Rutan Varienze.....	137
Figure 8.2.33: Auto-mesher results for the Varienze post CAD design .....	138

Figure 8.2.34: Vorticity distribution on the Varieze.....	139
Figure 8.2.35: ICL distribution about the symmetry plane.....	140
Figure 8.2.36: Lift coefficient span-wise distribution about the symmetry plane .....	141
Figure 8.2.37: Lift coefficient versus angle of attack for the Varieze .....	142

## LIST OF TABLES

Table 1.1: Overview of Aerodynamic Evaluation Schemes .....	2
Table 3.1: Case study results .....	25
Table 5.1.1: Size based meshing results for the truncated cone surface.....	56
Table 5.3.1: F-16A automated mesh refinement results .....	68
Table 7.4.1: DLR-F6 solver results comparison (AOA = 0°, 1 m/sec flow) .....	89
Table 8.2.1: Auto-mesher results for the F-16A .....	110

## NOMENCLATURE

$x_P$	X-coordinate of spatial point P
$y_P$	Y-coordinate of spatial point P
$z_P$	Z-coordinate of spatial point P
$V_i$	Velocity induced by the $i$ edge of a vortex ring mapped to a facet
$\Gamma$	Circulation around a marked spatial loop
$dl$	Incremental length along an edge of a mapped vortex ring
$r$	Position vector of a spatial point P from a vortex filament
$N_{edges}$	Total edges on a polygonal facet of arbitrary valency
$A_{j,P}$	Geometrical influence coefficient for a spatial point P by mesh facet $j$
$\gamma_{k,wake}$	Vorticity shed into the wake node $k$
$N_{sources}$	Number of vorticity sources flowing into a wake node
$N_{sinks}$	Number of vorticity sinks flowing away from a wake node
$\theta_{2P}$	Angle between the vortex filament length vector and the position vector from the terminal vertex of the filament to the spatial point P

$\theta_{1P}$	Angle between the vortex filament length vector and the position vector from the initial vertex of the filament to the spatial point P
$r_c(t)$	Radius of a vortex core along a given wake strand at pseudo-time $t$
$\varpi$	Angular velocity of a two-dimension vortex
$dt$	Increments in pseudo-time
$X_{Trefftz}$	Distance to the downstream location of the Trefftz plane
$X_{Seed}$	Distance to the seed point of the wake node
$n_{\infty, x}$	Component of the free-stream velocity vector in a direction normal to the surface of the Trefftz plane
$\rho_{\infty}$	Free-stream density
$dy$	Incremental length along the integrated circulation loop span-wise direction
$b$	Span of the integrated circulation loop surface around a geometry
$V_{\infty}$	Free-stream velocity
$L_{induced}$	Induced lift around a geometry
$D_{induced}$	Induced drag around a geometry
$w_y$	Induced downwash between the two consecutive integrated circulation loops at the span-wise location $y$
$y_0$	Downwash evaluation position between any two consecutive integrated circulation loops



$\mu$	Free-stream fluid viscosity
$Re_x$	Local Reynolds number for a given mesh facet
$C_F$	Skin friction coefficient for a given mesh facet
$L_{12}$	Length of edge between vertices 1 and 2 for a given mesh facet
$L_{23}$	Length of edge between vertices 1 and 2 for a given mesh facet
$L_{31}$	Length of edge between vertices 1 and 2 for a given mesh facet
$B$	General vector potential field

## 1. INTRODUCTION

Current industry trends point toward an increasing use of optimization based design pipelines for aerospace and marine applications<sup>1-4</sup>. Numerous optimization schemes have been developed based on varying underlying philosophies<sup>2-3</sup>. However, for complex systems, one aspect is common to all of these optimizers: the need for hundreds or thousands of design evaluations to arrive at a truly global optimum design<sup>1-2</sup>. Since each of these designs must be tested for aerodynamic or fluid-dynamic performance, a balance between minimizing the computation time versus maintaining fidelity of solution and validity of the physics must be made. This balancing represents a major limitation in the comprehensive use of Multi-Disciplinary-Optimization (MDO) design pipelines<sup>2</sup>.

There is a certain subset of MDO applications in aerospace and marine designs where the determination of aerodynamic loads is the precursor for the quantified performance index in the optimization process<sup>2</sup>. For these applications, traditionally four major methodologies have been used. These include empirical formulations, potential-flow methods, solution of Navier-Stokes or Euler equations in volumetric meshes (hereby generally referred to in this document as Computational-Fluid-Dynamics or CFD) and experiments in the field (wind- and water-tunnels as well as flight testing). Each of these four approaches brings with it a unique requirement for user inputs, computation times,

fidelity of solution, possibility for integration into MDO, range of applicable physics and overall cost of the process. Table-1.1 summarizes these parameters in a general sense for each of the four approaches. Clearly each approach balances out advantages in certain niche areas with disadvantages in other sectors.

**Table 1.1: Overview of Aerodynamic Evaluation Schemes**

	<b>Empirical equations</b>	<b>Potential methods</b>	<b>Navier-Stokes solvers</b>	<b>Experimental methods</b>
Complexity of input data/geometry	Low	Intermediate	Heavy	Heavy
Run time for single design	Low	Intermediate	Intermediate/ Heavy	Heavy
Solution fidelity	Intermediate	Intermediate	Good	Good
Automation for optimization	Very high	high	Low	Very low
Range of physics	Limited	Limited	High	High
Cost of process	Very low	Low	High	Very high

Experimental methods usually offer the closest approximation to actual flight test data but at exceptional cost per design case. In addition, they are highly susceptible to environmental errors and anomalies (i.e. noise) in the results if any attempt is made to reduce the costs or the computational time for the test, all of which are the highest compared with the other three approaches. As such, experimental schemes are restricted

to studies of designs at advanced levels within the design cycle and typically follow CFD testing and validations of the design expectations.

CFD approaches are rapidly making inroads into the MDO environment and have had visible success in demonstrating the use of wide range of physics during optimizations <sup>2</sup>. However, the approach is currently restricted to simpler optimization cases for numerous reasons <sup>2</sup>. One reason is the hardware associated with CFD approaches. For aerospace applications in the external-aerodynamics categories as well as certain marine applications, the necessary hardware to do thousands of CFD runs is very prohibitive to most users during the early design phases. However, this limitation is expected to be temporary. Current trends suggest that this limitation is rapidly being eroded as hardware expansion costs reduce industry-wide and shared industry pooling of large computing facilities becomes common practice <sup>2</sup>. Even so, for the present time, it is expensive to use CFD in MDO pipelines.

The primary limitation, however, is more philosophical in nature and relates to how CFD approaches are applied over a user specified geometrical surface. Since CFD approaches are volumetric in nature, i.e. they require volume meshes around the surface of interest to solve for the flow-field, the weak link in the solution fidelity chain becomes the quality of mesh and the automation of mesh generation. Current volumetric meshers are more robust today than at any other time as far as single run cases are concerned. However, they still require significant user inputs and visual verifications before being passed to the solver. For a large number of external aerodynamics problems, solver stability is directly related to the overall volume mesh quality while the solution fidelity

is dependent on the refinement of volume meshes in areas of interest around the input geometry. Slight variations in the mesh cell density or quality can cause solver divergence during iterations (non-physical “trip-ups”) or cause fluctuations in the evaluated loads (from pressure field summations) that can render useless an automated MDO process. Furthermore, the requirement to visually identify regions of interest for further refinement prior to solver runs makes it virtually impossible to automate the process under the direction of an optimizer.

Generally, the meshing is split into two phases in CFD approaches: surface and volume. The input geometry must be cleaned up in order to create a sufficiently refined surface mesh before the volume meshing can begin. This creates an additional layer of possible errors and user inputs and the final result is highly sensitive to the user settings during the surface meshing phase.

Finally, there is the issue of interfacing with the optimizer. Because of the dozens of independent meshing parameters that must be varied for changing input geometries, automation of geometry variation is a major problem in terms of optimizer control <sup>2-3</sup>. So far, only simple cases with modest design spaces have been shown to be optimized using CFD. This is expected to change in the coming years as more focus is applied on making the meshing independent of user inputs.

Potential flow methods have their own unique set of advantages and limitations with respect to an MDO environment <sup>5-7</sup>. Depending on the type of solver chosen for these approaches, the choice of user inputs, flexibility, stability and output data varies.

However, the biggest limitation of potential flow solvers is the restricted range of physics in which they operate <sup>2, 5-7</sup>. This is typically the purely subsonic flow regimes (Mach < 0.8) or the truly supersonic flow regimes (Mach > 1.5) and even then only for inviscid flow-fields. By comparison, both CFD and experimental methods can smoothly transition between subsonic, supersonic and hypersonic flows with both viscous and non-viscous environments.

However, for external aerodynamics MDO requirements for cruise conditions, this range of physics is usually very usable given the other major advantages offered by potential flow solvers. Typical applications in this genre include subsonic aircraft and missiles designed for long-range cruise flights, slow-speed wind-energy turbines, sail-boat designs, steady and unsteady flow behind propellers, aerostats and marine propulsion. Once the initial assumption is made to use potential-flow solvers inside their range of applicability, the other inherent advantages make them highly competitive and popular for MDO.

One of the major advantages of potential-flow solvers over CFD solvers is the requirement of only surface meshes in order to capture the entire volumetric effect caused by the input geometry to the exposed flow-field. The elimination of volume meshing removes many of the aforementioned limitations regarding volume meshers. Furthermore, surface meshes result in substantial savings in computation time during the design evaluations (volume meshing for external aero cases can consume 20-40% of the overall solver time in CFD <sup>3</sup>; this estimate applicable to most tetrahedral/polyhedral meshers, i.e. Delaunay meshers) as well as reducing the overall hardware requirements

(the majority of the memory requirements are associated with volume mesh generation and saving of the generated mesh on the local machines).

Another advantage inherent to potential-flow solvers is the solver stability<sup>2</sup>. Only geometrical surfaces with self-intersections, non-manifold interfaces and non-conformal patches on the surface mesh are known to be problem areas. Other areas of instability can include the inappropriate use of automated wake relaxations but this is generally minor.

Finally, the solution convergence times associated with potential-flow solvers is known to be very efficient for most applications even when using surface meshes with large variations in mesh densities<sup>2, 5</sup>. This reduces overall computation times for an MDO application substantially compared with a similar CFD approach.

Potential-flow solvers are not without their inherent limitations beyond the restricted range of physics, however<sup>5-7</sup>. As these solvers have become more advanced over the years, they have started to emulate the same limitations that afflict CFD solvers. The primary cause behind this is the use of pressure field summations to evaluate loads over the geometry. The need for this is driven by the attempts to push potential-flow solvers as alternates to CFD solvers for aerospace applications by various authors and organizations<sup>5-11</sup>. For example, NASA's PMARC, VSAERO<sup>8-10</sup> and PANAIR<sup>11-12</sup> potential flow solvers attempt to include schemes to evaluate viscous boundary layers<sup>13</sup> and flow separation<sup>8-10</sup> using the pressure fields evaluated by the underlying potential-flow solvers. This has been a semi-successful attempt to expand the physics of potential-flow solvers and move the methodology closer to CFD. However, doing so has meant

that these solvers have lost several of the aforementioned advantages associated with potential-flow solvers whilst also not attaining the true fidelity of the CFD solvers with regard to viscous flow regimes and flow separation effects.

This argument can be elaborated on using a philosophical overview of the approaches taken by the three NASA potential-flow solvers. All three of these codes attempt to add a pseudo-coupled boundary layer model to the potential-flow results by adjusting the geometrical surface to account for the thickness of the local boundary layer<sup>8-13</sup>. This boundary layer thickness is evaluated locally using the knowledge of the local pressure and velocity fields as well as the distance from the nearest leading edge marked by the pressure field results. And while they do gather some success<sup>8-11</sup> in establishing the boundary layer flow, the limitations incurred on the solver outweigh any substantive gains from such an approach.

The primary limitation incurred is the requirement for these solvers to use a pressure-solver<sup>8-11</sup> rather than a vorticity-solver (elaborated later in this document). A pressure-solver necessitates the use of quadrilateral source distributions coupled with trailing edge doublet distributions<sup>8-10</sup>. As such, the solver becomes highly sensitive to the fidelity of the surface mesh on which it is applied (similar to CFD solvers and unlike vorticity-solvers). The pressure-solver cannot tolerate local surface “bumps” and “dents” on the mesh without lowering fidelity of the evaluated pressure field and hence the aerodynamic loads<sup>9-10</sup>. Furthermore the solver results become highly dependent on the accurate capture of the pressure field. This means that for simple airfoil cases, for example, highly refined surface meshes must be created in the high curvature regions in



order to capture the pressure gradients <sup>11</sup>. This is again similar to the CFD solvers and causes similar cascading effects downstream into the optimization pipeline including higher computation times, solver sensitivity to surface meshes and reduced flexibility for the optimizer to dramatically vary the geometry within the design space <sup>2</sup>.

If further flow modifications are needed such as viscous boundary-layers <sup>13</sup>, wherein the potential solution must be iterated until stability with the boundary layer is established, the computation times go up even higher <sup>9-11, 13</sup>. Another attempt in recent times has been to incorporate separation flow effects within the potential-flow solutions <sup>11</sup>. The general idea behind this is to use the pressure-field evaluated from the potential solver to determine flow separation points on the geometry using empirical formulations and use them to move the trailing edge locations up or down on the geometry. This is computationally very intensive and has been shown to be effective only under very controlled conditions <sup>10-11</sup>. As such, any solution results for known flow separation cases are purely the domain of CFD solvers that can evaluate these regimes more efficiently and with relatively higher solution fidelity.

Pressure-solvers and Vorticity-solvers use different panel types to suit their underlying theme. Pressure-solvers, as the name alludes, use the determination of the pressure field around given geometry to determine loads. As such, they use combination of source and sink panels coupled with doublet panels for the trailing edge. Source and sink distribution panels are mathematically accurate in their evaluation of the local pressures and velocities and hence suit pressure-solvers.

Vorticity-solvers take an entirely different approach altogether. They depend on the evaluation of the circulation around a given geometry and use the Kutta-Joukowski formulation applied to each panel for evaluation of the induced loads <sup>14</sup>. As such, these solvers typically use vortex rings or doublet distributions over the surface facets <sup>15</sup>. These types of potential flow panels are not suited for evaluation of the pressure field accurately except in far-field regions relative to the panel and on panel control points where the external physics conditions are satisfied <sup>14</sup>.

With these underlying philosophical differences, the requirement for surface meshes and their applicability to an MDO environment vary substantially between the two solvers <sup>2, 14</sup>. Pressure-solvers are able to work with unstructured meshes as well as structured ones, mainly because each surface facet is evaluated for a certain source or sink strength regardless of the orientation of its edges relative to the flow direction. However, vorticity-solvers have thus far been used only with structured meshes since the local application of the Kutta-Joukowski formulation for loads at the facet level requires segregation of facet edges into two categories: bound or trailing <sup>2, 14</sup>. This forces the mesh to rely on pre-established U-V mapping and restricts the use of advanced geometries with vorticity solvers <sup>2</sup>.

To balance this limitation, vorticity-solvers are known to be substantially more robust and stable compared to pressure-solvers <sup>14</sup>. They allow for the user of non-manifold and non-conformal mesh surfaces (which the pressure-solvers cannot handle) and are less sensitive to surface perturbations. They also allow use of coarser meshes on

the geometry <sup>14</sup>. This reduces the computation time substantially and affords greater stability to the solver, making it usable in an optimization environment <sup>2</sup>.

It is therefore apparent that both solvers have their inherent advantages and disadvantages. However, it is also apparent that in order to be truly applicable for a wide range of MDO applications, niche advantages from both solvers is needed and must be combined to harness the true potential of potential-flow solvers. This forms the basis of the current effort.

## 2. MOTIVATION

A summary of the main objectives of the new approach can be written thus:

A) Retain advantages of the potential-flow vorticity-solvers:

- Fidelity of solution:

Aerodynamic loads (lift, induced drag and viscous drag) are used as a measure of the fidelity of the solution. The range of free-stream conditions is for Mach Numbers below 0.8 using the compressibility correction factor. It will be shown later that for certain geometries the range of free-stream Mach number can be extended to 0.9.

- Speed of solution:

High solver speed is paramount for use of any aerodynamic prediction methodology within an MDO environment. It is possible to use the overall CPU time till solver convergence as a measure of the solver speed. This document will refer to solver speed in the context of this parameter throughout.

- Flexibility:

Ability to evaluate aerodynamic loads over surfaces with high curvature, proximity, intersections, non-conformity of patch interfaces

and abrupt sizing gradients is paramount to creating a high fidelity MDO environment. Although difficult to quantify in and of itself, the measure of the speed of the solver and the fidelity of predicted loads will be used as a measure of the robustness against such “high stress” geometrical surfaces

- Stability of solver:

The solver must always attempt to prevent divergence during iterations even when applied over difficult-to-solve geometrical surfaces. Quantification of solver stability must be developed and used to improve automated prognosis methods to prevent solver divergence.

B) Establish ability to work with unstructured surface meshes

- Allow use of triangular facets of arbitrary orientation relative to flow
- Bring potential-flow surface meshing to current industry standards:

This can be split into several sub-goals:

- a) Introduce size based meshing to potential flow solvers
- b) Quantify surface mesh quality and automate its control
- c) Maintaining mesh intra-patch and inter-patch conformity or show lack of sensitivity of solver to it
- d) Introduce surface meshing data-packet structures similar to CAD/CAE industry standards

- Remove requirement for forced U-V mappings on underlying geometry, thereby improving
- Open wide-spectrum geometries to potential-flow solvers. This remains a qualitative goal which can be demonstrated via a diverse testing suite for the current solver.

Each of these goals will be addressed in the current document and a summary of the results will address the degree to which they have been achieved.

### 3. THE NEED FOR UNSTRUCTURED MESHES

A structured surface mesh is typically defined as one where a given surface is mapped along two numerical axes (“i-j” or “u-v”) and each facet is bound by four vertices extracted from the rectangular structure of the underlying map (known as the U-V map). In essence, this means that regardless of the physical convolution or warping of the physical surface, the underlying U-V mapping is rectangular with uniformly spaced increments. Each vertex of this mesh has an associated pair of numerical coordinates so that a vertex P is stored in the mesh as:

$$\begin{aligned}x_P &= X(i, j) \\y_P &= Y(i, j) \\z_P &= Z(i, j)\end{aligned}\tag{3.1}$$

And a surface facet is created out of a set of four corner vertices as:

$$\begin{aligned}Fx_{i,j} &= [X(i, j), X(i + 1, j), X(i + 1, j + 1), X(i, j + 1)] \\Fy_{i,j} &= [Y(i, j), Y(i + 1, j), Y(i + 1, j + 1), Y(i, j + 1)] \\Fz_{i,j} &= [Z(i, j), Z(i + 1, j), Z(i + 1, j + 1), Z(i, j + 1)]\end{aligned}\tag{3.2}$$

As such, any structured surface mesh facet is essentially quadrilateral in nature. As a result, facets have to be “forced” to assume triangular shapes to meet geometrical demands. This is accomplished by “edge-collapsing” wherein two of the facets on a given

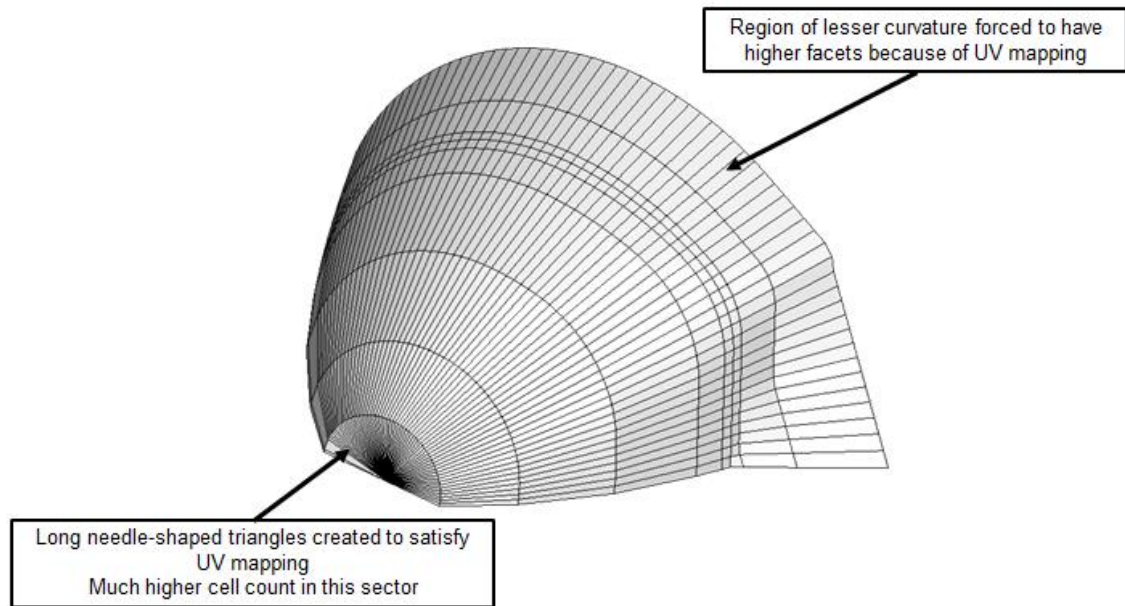
edge of the facet are merged on top of each other to reduce the quadrilateral facet to a triangle in physical space. In numerical space such triangles still retain the underlying quadrilateral U-V map.

The advantages and disadvantages of such approaches to mesh generation are obvious. The forced quadrilateral U-V mappings of the underlying facets ensure that the memory requirement for a given forced structured triangle is always more than an equivalent unstructured triangle facet. In addition, forced U-V mappings ensure that regions of high curvature or concave warping are always populated with more facets than actually required from a geometrical standpoint. As a result, a structured mesh patch forced on such geometry always results in higher mesh facets. Further, the quality of such facets is typically very poor (with downstream in the solver where oscillations are induced).

Additionally, the typical quadrilateral facet has to be sanitized to establish its “effective” surface normal direction given its higher level nature with regard to the baseline triangle. With triangular facets, this is not required given the deterministic nature of the triangle geometry in three-dimensional space. However, since the quadrilateral facet is a combination of a pair of triangles (which can be further varied depending on choice of diagonal used), there is no fixed surface normal. This problem is typically eliminated by evaluating the surface normal as a cross-product unit vector result of the two diagonal vectors of the facet. Understandably, this reduces the geometry capture fidelity of the facet. If the quadrilateral facet were to be split into two facets to improve this fidelity, the number of resultant mesh facets increases.



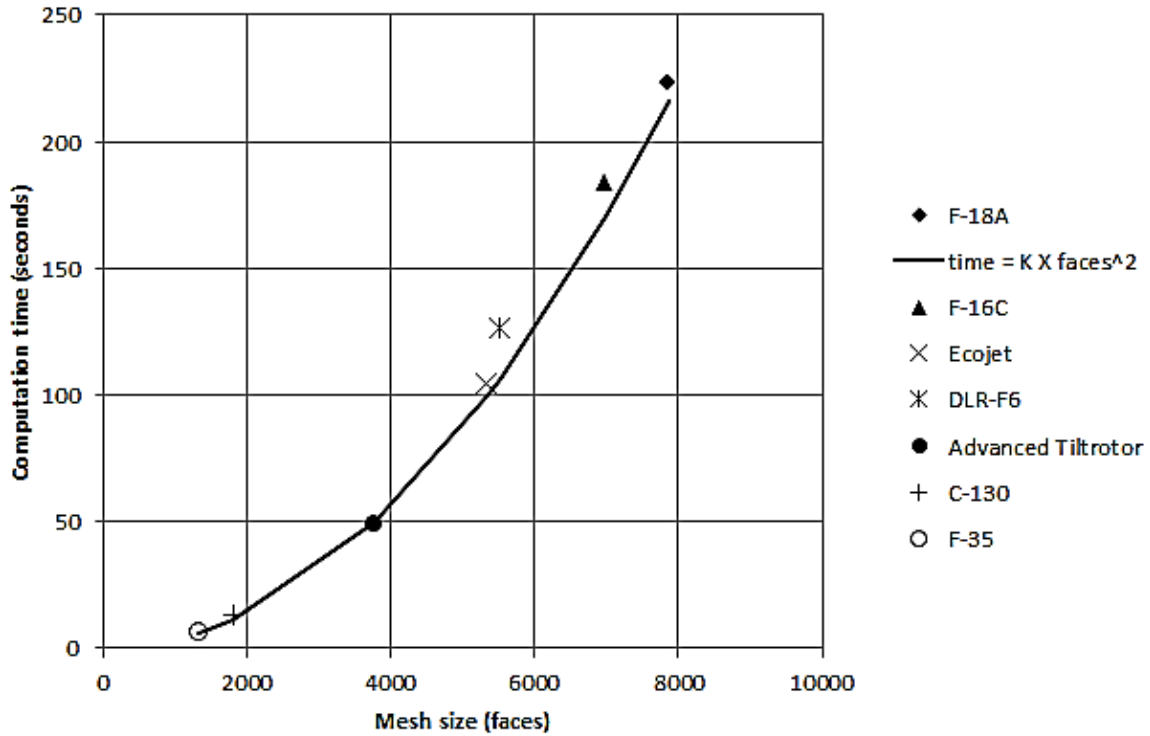
Figure-3.1 illustrates the application of a structured mesh patch on a high curvature surface. The long needle-shaped facets near the nose of the geometry are visible. Such facets are typically of very poor quality and can significantly increase the mesh facet count.



**Figure 3.1: A typical structured mesh over a high curvature surface**

A general rule of thumb obtained from the current effort shows that the computation time for the solution convergence is directly proportional to the square of the mesh facet count (Figure-3.2).

$$\text{Computational time} \propto \text{facet}^2 \quad (3.3)$$



**Figure 3.2: Solver run time data versus mesh size for various geometries. All geometries run for 10 iterations only**

Further, the regions of lower curvature are populated with much higher number of facets than needed because of the need to satisfy the underlying U-V mapping with appropriate physical locations. Such adherence to the U-V mapping means that the mesh is unable to recognize and react to high curvature regions effectively in physical space. This restricts the effective application of these meshes to simplified regions with long gradual curvature regions.

Despite the aforementioned shortcomings of the structured surface meshes, with potential-flow solvers (and especially the vorticity-solvers) they offer certain unique advantages. Arguably, vorticity-solvers in their prior form have been made possible mainly because of the inherent advantages of the structured mesh. Further, mesh

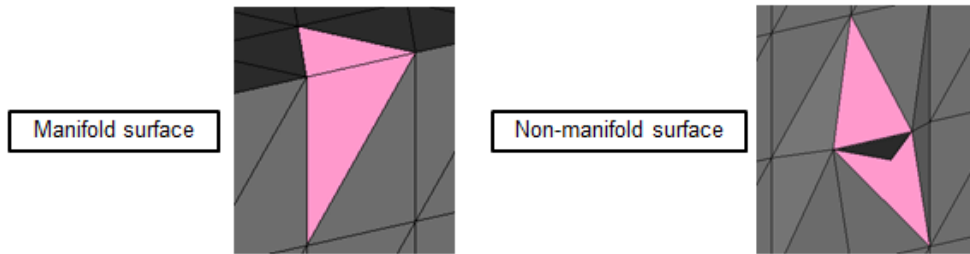
generation with structured surface mesher is much simpler from a development standpoint than a comparative unstructured surface mesher.

The vorticity-solvers require a facet edge pool to be split into bound and trailing vortices during the application of the Kutta-Joukowski formulation for induced loads at the facet level. This means that facets of arbitrary orientation (as obtained from unstructured meshes) cannot be applied. Traditionally, the U-V mapping of the structured meshes has been used in all legacy vorticity-solvers to establish the bound and trailing vortices. The bound vortices are simply marked as those facet edges aligned along a certain numerical axis (U or V) and the other edges are marked as trailing. Note that such marking usually only corresponds to the correct flow field if the physical space warping and curvature is not significantly different from the numerical axes alignment. Therefore, such solvers are only applicable on geometrical surfaces that meet this criterion which is a major limitation of the vorticity-solvers.

Further, the induced load formulation requires the knowledge of the vorticity in the next bound vortex downstream of a given facet. This information is also evaluated from the U-V mapping information. This reduces the computational burden on the solver and allows the evaluation of loads in a simple and straightforward manner.

Having established the merits and demerits of the structured surface meshes, a similar analysis can be done for the unstructured surface meshes. The primary philosophical difference between the two meshes is the lack of an underlying U-V map on the unstructured meshes and that the unstructured meshes are first and foremost

triangulated (instead of quadrilateral in structured meshes). Meshes of this type are not bound by any numerical forces in the distribution of surface vertices. Each surface mesh generated using this approach is essentially a numerical pool of vertices, edges and faces that are marked using connectivity mappings evaluated at the time of mesh generation. A given vertex may be shared by numerous edges depending on the curvature of the region. A given edge may be shared by two faces (for a manifold surface) or more than two faces (for a non-manifold surface) as shown in Figure 3.3.



**Figure 3.3: Definition of manifold and non-manifold mesh surfaces (selected facets are shown in pink; note the number of faces sharing an edge)**

Information storage for the mesh can vary. Industry practitioners use differing mesh storage concepts. For example, a facet can be fully defined by knowing its three vertices in three-dimensional space. Such a mesh representation can be written as  $F \rightarrow V$  where  $F$ =Facets and  $V$ =Vertices. Similarly, a facet can also be defined entirely from the knowledge of its three edges (which in turn are marked from the vertex mapping) and such a mesh representation can be written as  $F \rightarrow E \rightarrow V$ . For the current solver the mesh information storage is  $F \rightarrow V$ .

Data packets store each facet's information including vertices, edges, topological connectivity and external physics as primary data while derivative data such as facet area, quality metrics, surface normal vector, centroid etc. are stored or evaluated as necessary.

Unstructured surface meshes bypass some of the aforementioned limitations of the structured surface meshes. Geometry capture with unstructured meshes is more efficient and can be designed to significantly reduce overall mesh count and improve the quality of the facets (i.e. reduce the presence of large number of needle shaped facets). They can capture regions of concave and convex warping with much more even distribution of facets and facet edge-sizes. They can also be used to enact local surface refinements more effectively in physical space rather than having to do so in the underlying U-V space.

Because each facet on such a mesh is a self-contained entity, bad quality facets can be deleted, improved or refined to meet mesh requirements. Refinement can reduce the overall solver convergence oscillations (caused by the poor-quality cells; discussed later) and thereby reduce the computation time to convergence. Unstructured meshes also allow superior control on the mesh facet sizing as well as the sizing gradients. Further, each facet can be marked with its local external physics and boundary information. This allows much simplified creation of boundary zones (such as velocity inlets and trailing edges), but also assists in the automated detection of trailing edges on a surface.

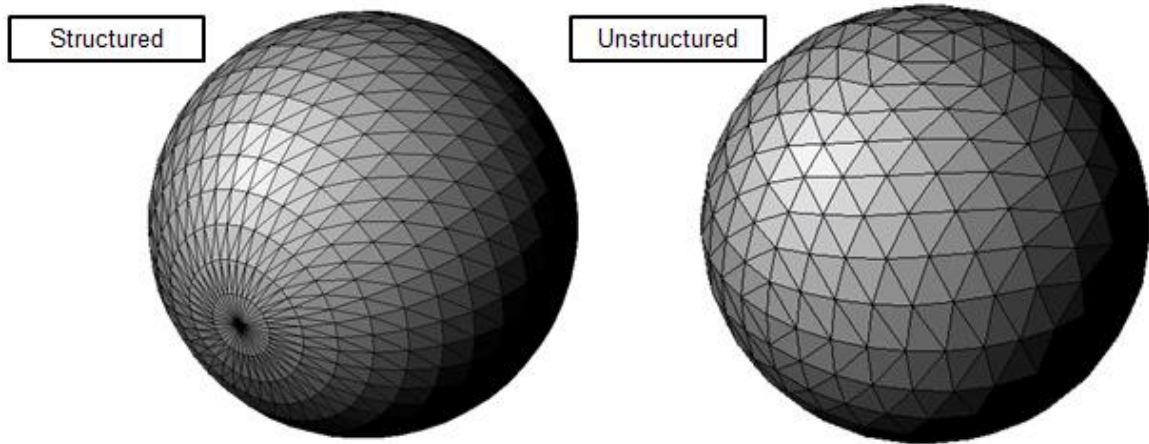
The limitations of the unstructured surface meshes pertain mainly to their application with the potential-flow vorticity-solvers. And while the local evaluation of the

vorticity can be made similar to the pressure-solvers using the external and boundary physics on the surface mesh, the induced loads cannot be evaluated without marking appropriate bound and trailing vortices as discussed for structured meshes. This has been the main limitation in the use of unstructured meshes with vorticity solvers thus far and is the primary motivation for the development of Method of Integrated circulation as will be discussed in the following Sections of this document. However, before moving to the discussion of the solver and evaluation of integrated loads, a case study has been presented to further illustrate the need for unstructured meshes.

A simple geometry has been taken (a unit radius sphere with its center at the coordinate axes origin) and meshed using a structured and unstructured surface mesher developed for this effort. The two spheres have been refined to similar mesh requirements, i.e. the number of mesh vertices along any quadrant arc of the sphere must be equal to ten (this can be visually inspected from Figure 3.4). Note that the structured mesh is forced to convert its quadrilateral facets into triangular ones so that they can both be run through the current solver. Additionally, note that the forced U-V mapping on the structured mesh causes the existence of a large number of bad quality facets near the polar regions of the sphere (Figure-3.4) whereas the unstructured mesh easily bypasses this potential pitfall. As a result, the final mesh size between the two meshes is substantially different (Table 3.1).

The two meshes were run through the solver under identical conditions and the results are shown in Table-3.1. Note that the aim here was to obtain a potential flow solution around the two spheres. As a result, no wake definitions were necessary and the

expected coefficient of lift for any angle-of-attack was expected to be zero. The flow velocity for the current case study was kept at 1.0 m/sec.



**Figure 3.4: Case study sphere in structured and unstructured environments**

The unstructured surface mesh is seen to take about ~88% less time to reach much superior convergence limits in fewer iterations and with higher solution fidelity. The structured mesh reaches the limit set on the solver iterations and still does not meet the required convergence criteria. This is caused by the solver oscillations resulting from large number of poor quality facets in its polar regions. The resulting mesh size created out of a similar refinement criterion is also favorable for the unstructured mesh (648 facets) compared with the structured mesh (1,404 facets).

The resulting trends of this case study are applicable to a wide range of test cases and geometrical variations and help establish the necessity and advantage of the unstructured meshes for the vorticity-solvers. The next Sections of this document discuss the development of the Method of Integrated Circulation to allow evaluation of circulation based loads on such meshes.

**Table 3.1 Case study results**

	<b>Structured</b>	<b>Unstructured</b>	<b>Reduction</b>
Number of facets	1404	648	53.85 %
Solver run time (Seconds)	148.88	17.59	88.19 %
CL (expected = 0.0)	0.000025	0.000009	64.00 %
Convergence (expected = 1E-3)	9E-3	7E-4	92.22 %
Solver iterations (max = 20)	20	7	65.00 %



## **4. SOLVER METHODOLOGY**

This Section details the vorticity solver used for the current effort along with the modified approaches for the evaluation of the integrated aerodynamic loads. The Sections are split to include discussions on the following:

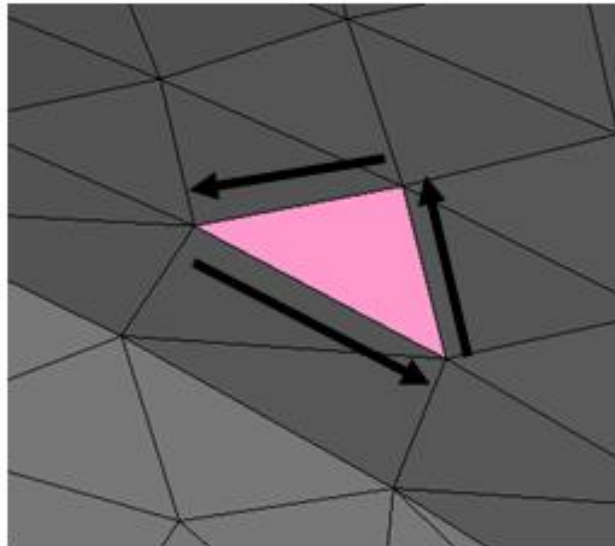
- a) The generalized spatial velocity inductions by segmented elements of the vortex ring mapped to each facet
- b) The philosophy of the solver from a numerical application standpoint
- c) Development of the relaxed wake-strand model for unstructured meshes
- d) The method of integrated circulation for evaluation of the induced loads on the unstructured mesh as well as the modifications to skin-friction drag models to allow their application into a vorticity-solver

### **4.1 SPATIAL VELOCITY INDUCTIONS**

At its most fundamental level, the solver developed for the current effort reduces to the application of a vortex ring on a triangulated facet. Each edge of the facet therefore becomes a segmented part of the continuous vortex ring on that facet. Since each of these edges remains a finite, line segment in three-dimensional space, the induced velocity of

each segment at any other point in space is easily determined from the geometry of the plane described by the two end vertices of the line and the point of induction in space. Once this induction has been evaluated for each edge of the facet, they are summed up and treated as the induced velocity of the facet at the point of interest in space.

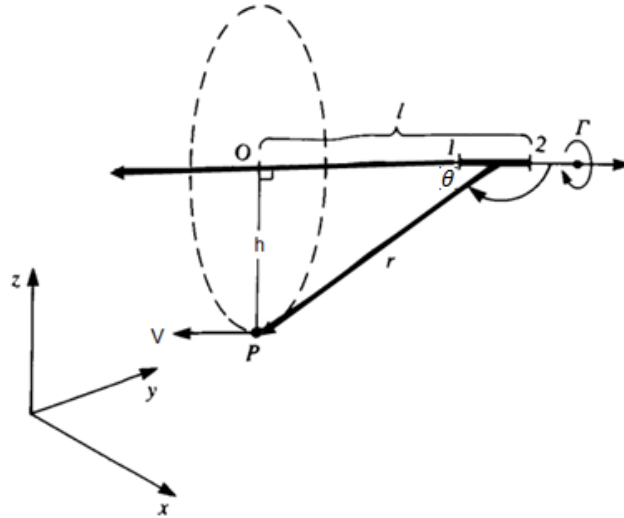
Additionally, the direction of vortex ring is aligned to match the winding of the facet as shown in Figure-4.1.1. Because each valid (non-degenerate) facet on the surface mesh is a closed loop with a known winding, the vortex rings are easily mapped on their surface. This is also in accordance with Helmholtz laws which require every vortex ring to be closed unless stretched to infinity on both ends of a segment<sup>14-15</sup>.



**Figure 4.1.1: The mapped vortex ring for a facet on the surface mesh**

The derivation for the velocity induced by a linear segment of a facet can be mathematically derived in terms of the coordinates of the two end vertices and the point

of interest. Consider a point P in space where a segment of a vortex ring is inducing a velocity as shown in Figure-4.1.2.



**Figure 4.1.2: Velocity induced at a point by a segment of a facet-bound vortex ring**

The velocity induced at P by an infinitely small segment of the ring edge is known from the application of the Biot-Savart law <sup>14</sup> as:

$$dV_i = \frac{\Gamma}{4\pi} \frac{dl \times r}{r^3} \quad (4.1.1)$$

Considering the geometry shown in Figure-4.2, Equation-4.1.1 can be modified:

$$dV_i = \frac{\Gamma}{4\pi} \frac{dl \cdot \sin(\theta)}{r^2} \quad (4.1.2)$$

Now, given that,

$$h = r \cos(\theta)$$

$$l = h \tan(\theta)$$

$$dl = h \cdot \frac{d\theta}{(\cos(\theta))^2}$$
(4.1.3)

Substituting Equation-4.1.3 into Equation 4.1.2, we get:

$$dV_i = \frac{\Gamma}{4\pi} (\cos(\theta))^2 \cdot \frac{d\theta}{(\cos(\theta))^2} \frac{\sin(\theta)}{h^2} = \frac{\Gamma}{4\pi} \frac{\sin(\theta) d\theta}{h}$$
(4.1.6)

Equation-4.1.6 can be integrated along the entire length of the segment so that the formulation for the net induced velocity by one segment of the vortex ring is:

$$V_i = \int_{\theta_1}^{\theta_2} dV_i = \frac{\Gamma}{4\pi h} \int_{\theta_1}^{\theta_2} \sin(\theta) d\theta$$

$$V_i = \frac{\Gamma}{4\pi h} (\cos(\theta_2) - \cos(\theta_1))$$
(4.1.7)

In Equation-4.1.7, all variables on the RHS are easily evaluated from vector formulations of the planar geometry defined by the three points in space (two vertices and one point of interest). Further, if the point of interest were to be so located that the radius becomes zero (all three points are collinear) then the solver inputs a zero velocity induction even though from Equation-4.1.7 it is obvious that we would have a mathematical singularity. Note that Equation-4.1.7 also reduces to the two-dimensional vortex Equation for the limiting case of infinitely long vortex (as a result of this, the two angles in Equation-4.1.7 becomes  $0^\circ$  and  $180^\circ$  respectively). Note also that the velocity magnitude evaluated by equation-4.1.7 is in a plane perpendicular to that defined by the

three points in space. However, knowing the surface normal vector for this plane from the three points, the velocity magnitude is broken down into three vector components along the global mesh coordinate axes.

Equation-4.1.7 is the induced velocity by one segment of a facet. Therefore, to evaluate the velocity induced by the entire ring, we evaluate Equation-4.1.8 over all the relevant edges of the ring to get the net velocity induction as:

$$V = \sum_{i=1}^{N_{edges}} V_i \tag{4.1.8}$$

A discussion of the derivation for the Biot-Savart law in the form used in Equation-4.1.1 is given in appendix-A.

## 4.2 SOLVER PHILOSOPHY

Considering Equation-4.1.7, it is clear that the only unknown variable in the Equation is the vorticity strength. The LHS of the Equation is a variable that can be controlled at certain points on the geometry using the boundary physics conditions. The remaining terms in the RHS minus the vorticity strength are geometrically dependent and therefore determined from processing the surface mesh. If the vorticity strength could be determined using known values of the velocity at the control points, then the flow-field can be determined at all points in space around the geometry, allowing for evaluation of

the loads. This can be further quantified by considering Equation-4.1.7 and Equation-4.1.8 as:

$$V_j = \Gamma_j \sum_{i=1}^{N_{edges}} \left[ \frac{1}{4\pi h_i} (\cos(\theta_{2i}) - \cos(\theta_{1i})) \right] = \Gamma_j A_{j,P} \quad (4.2.1)$$

The coefficient  $A_{j,P}$  is the geometrical influence coefficient for the velocity induction at point P by the facet “j” of the surface mesh. If the point P were a control point, then the LHS is a boundary defined velocity term. Conceptually expanding Equation-4.2.1 for a mesh with N facets, the overview of the fundamental solver philosophy is:

$$\begin{bmatrix} A_{1,1} & \cdots & A_{1,N} \\ \vdots & \ddots & \vdots \\ A_{N,1} & \cdots & A_{N,N} \end{bmatrix} \begin{bmatrix} \Gamma_1 \\ \dots \\ \Gamma_N \end{bmatrix} = \begin{bmatrix} B_1 \\ \dots \\ B_N \end{bmatrix} \quad (4.2.2)$$

Where the matrix B is the velocity requirements on the control points (and hence known from the external physics requirements) and the matrix A is the N×N matrix of geometrical influence coefficients for each of the control points. The matrix  $\gamma$  is therefore the N unknown variables that must be solved for the flow-field to be fully defined around the geometry. The system defined in Equation-4.2.2 is essentially a system of N Equations and N unknowns that can be solved explicitly.

The control points selected for triangular facets lie on the facet area-centroid and are unique in space (they have no edges from any facet on the mesh that crossed through it to introduce singularity) for a manifold, conformal surface mesh.

The  $N \times N$  matrix formulation in Equation-4.2.2 introduces certain problems as the size of the mesh increases. Since these matrices must be inverted during the solver run, the issue of numerical errors cascading up the inversion loop is severe and can lead to completely non-physical and erroneous results for the vorticity <sup>2</sup>. Consequently, the matrix must either be evaluated in patches numerically during the inversion phase or the solver must be handed small patches of the mesh individually <sup>16</sup>. The current effort handles this at the physical level and splits the mesh into partitions that are fed to the solver individually, freezing the remaining Sections of the mesh (their vorticity values) at the previous iteration.

This approach creates an iterative solver for the overall surface vorticity. Given the nature of the geometry, an exact solution can never be arrived at practically and the solver iterations must be terminated once the surface vorticity values have “converged” to a reasonable level of numerical accuracy. The convergence criterion used in the current effort is based on the stability of the surface vorticity solution and the stability of the relaxed wake strands in the Trefftz-plane downstream of the geometry.

The vorticity convergence is evaluated as an area-averaged value of the facet vorticity strengths on the entire surface mesh. The wake-instability is quantified as the

RMS change of the two-dimensional strand positions within the Trefftz plane (Equation-4.2.3).

$$convergence_{wake} = \sum_{i=1}^{N_{strands}} \frac{[(y_t^2 + z_t^2) - (y_{t+dt}^2 + z_{t+dt}^2)]}{N_{strands}} \quad (4.2.3)$$

The wake-instability criterion is secondary in nature and is under user toggle control. This is because terminating wake relaxation at a lower convergence value can lead to a wake that is not force-free and therefore can adversely affect the solution. Only in the case of simple geometries can such relaxation be frozen without affecting the overall result. Wake instability as a convergence criterion comes into play only as a means to reduce overall computation time and is entirely case dependent.

A further note regarding compressibility can be made at this point. The application of the Biot-Savart law for velocity induction, although formulated for incompressible flows, can be modified with the addition of the compressibility correction factor<sup>24</sup> and is used in the current approach to simulate higher Mach number flows. The importance of such corrections is made clear in Section-8 of this document.

#### **4.3 VORTICITY WAKES: THE RELAXED STRAND MODEL**

Unstructured meshes drive the wake geometry to be unstructured since the trailing edges are not mapped numerically. As a result, the wake sheet concept<sup>2, 14</sup> used in the



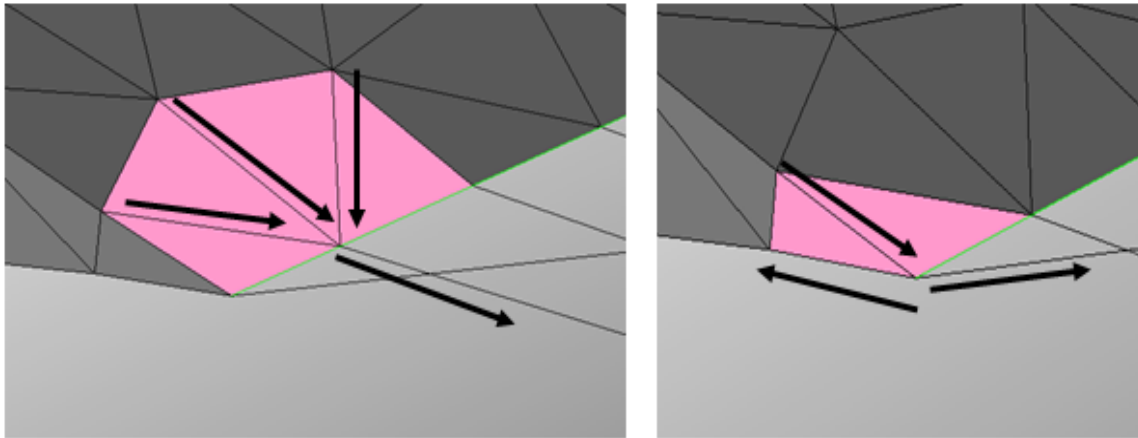
standard relaxed wake models is not applicable with the current approach. This problem has been resolved by the evolutionary extension of the relaxed wake sheet into the relaxed wake strand approach developed for this effort.

The general principle for this approach is based around the conservation of vorticity<sup>14</sup> in the wake (in accordance with Helmholtz Laws of vorticity) for a strand emanating from “nodes” rather than edges on the geometry. This is illustrated in Figure-4.3.1. The starting point for this model is based on user-marked trailing edges on the geometry (either automatically or manually; see Section-7 on boundary conditions and external physics). Once the trailing edges are marked, the solver determines the vertex pool corresponding to these edges from the global vertex map. These vertices become the starting point or nodes, for the individual wake strands. Each node is then numerically propagated downstream using the induction effects of the overall facet-bound vortex rings, the free-stream velocity and the wake-strands from the previous iteration (excluding the strand being updated).

The evaluation of the strength of the vorticity strand being propagated into the wake requires the knowledge of contributing “source” and “sink” edges for each node. These are obtained from the global vertex and edge connectivity maps and are illustrated visually in Figure-4.3.1. Edges of any facet that share the wake node are automatically marked as contributors to the node (unless the edge is marked as a trailing edge, in which case no vorticity is allocated to it and hence cannot be a contributor; green-highlighted edges in Figure-4.3.1). A vorticity contributor can be a source or a sink depending on

whether the direction of “flow” of vorticity along the edge (determined from the facet winding) is towards or away from the wake node.

Consider the two cases shown in Figure-4.3.1. Here the case on the left has three edges that share the wake node in addition to the two trailing edges also sharing it (marked green). The three edges sharing the wake are in turn shared by two facets each as part of the manifold surface. As a result, each edge is contributing the difference between the vorticity of its shared facets to the wake node. So this wake node has three contributing edges, two trailing edges and four contributing facets on the top surface. A similar situation may exist on the bottom surface as well.



**Figure 4.3.1: The wake-strand model at the trailing edge**

Similarly, consider the second case on the right in Figure-4.3.1. Here the wake node is being shared by two facets on the top surface, has two contributing edges and one trailing edge. The net vorticity of this wake then is the difference between the facet vorticity of the two facets sharing it.

Mathematically, the vorticity of the strand can be written as:

$$\gamma_{k, \text{ wake}} = \sum_{i=1}^{N_{\text{sources}}} \gamma_i + \sum_{j=1}^{N_{\text{sinks}}} \gamma_j \quad (4.3.1)$$

Because no new sources or sinks are added by the wake nodes, the net vorticity of the geometry remains conserved and the wake propagates only the required vorticity in the wake to satisfy the Kutta condition at the trailing edges.

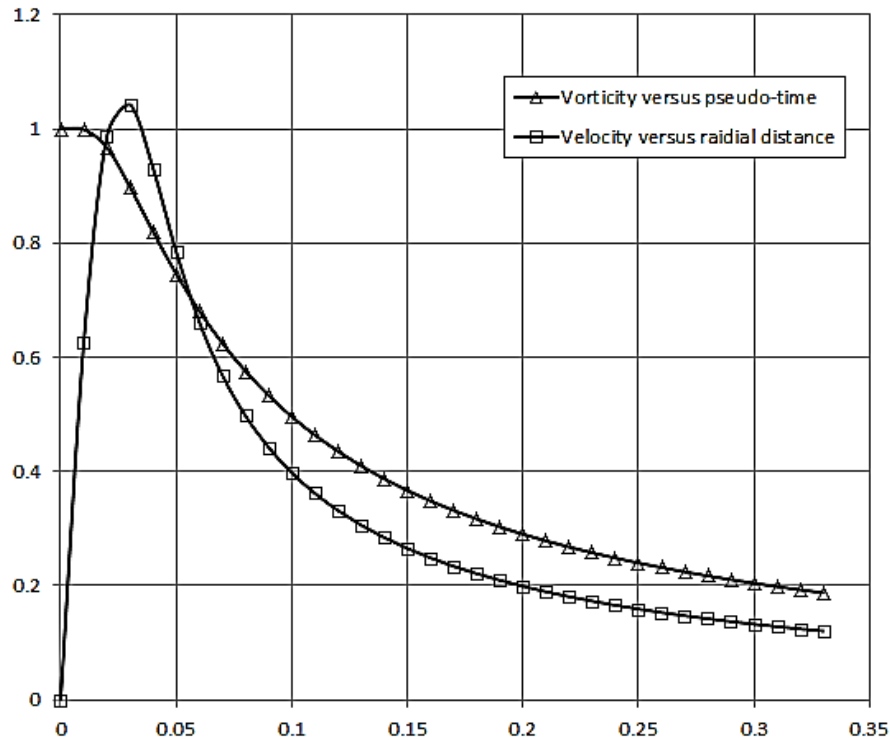
Vorticity strength reductions downstream of the trailing edge can only occur due to viscous effects. This is modeled using the application of the Lamb-Oseen<sup>17-19</sup> vortex model obtained from the exact solution of the Navier-Stokes equations for a laminar two-dimensional vortex. This model assumes a two-dimensional vortex with circular symmetry in which the streamlines are circles around the axis and the vorticity is a function of the radial distance away from the filament axis and the pseudo-time<sup>17</sup>. The exact solution of the Navier-stokes equations<sup>17</sup> then takes the form:

$$\omega = \frac{\gamma \rho_{\infty}}{4\pi\mu t} e^{\left(\frac{-\rho_{\infty} r^2}{4\mu t}\right)} \quad (4.3.2)$$

Assuming the knowledge of the initial vorticity shed into the strand at the node on the trailing edge, Equation-4.3.2 is modified to give us the vorticity decay equation<sup>17</sup>:

$$\gamma = \gamma_0 \left(1 - e^{\left(\frac{-\rho_{\infty} r^2}{4\mu t}\right)}\right) \quad (4.3.3)$$

In Equation-4.3.3, the time used corresponds to the pseudo-time and starts at zero at the trailing edge. It is marched alongside the strand until the termination of the wake downstream at the Trefftz location. The behavior of Equation-4.3.3 is shown in Figure-4.3.2 for the vorticity decay rate versus wake pseudo-time for a fixed radial position and for the velocity induction by a single filament in the radial direction for a fixed pseudo-time. The advantage posed by the implementation of the Lamb-Oseen model is visible from the results in Figure-4.3.2. The vorticity decay rate results in the “desingularisation” of the rectilinear line vortex, in which the vorticity has a delta-function singularity. The velocity function also indicates the growth and an effective solid-rotation core along the length of the propagating wake-strand filament which simulates viscous behavior.



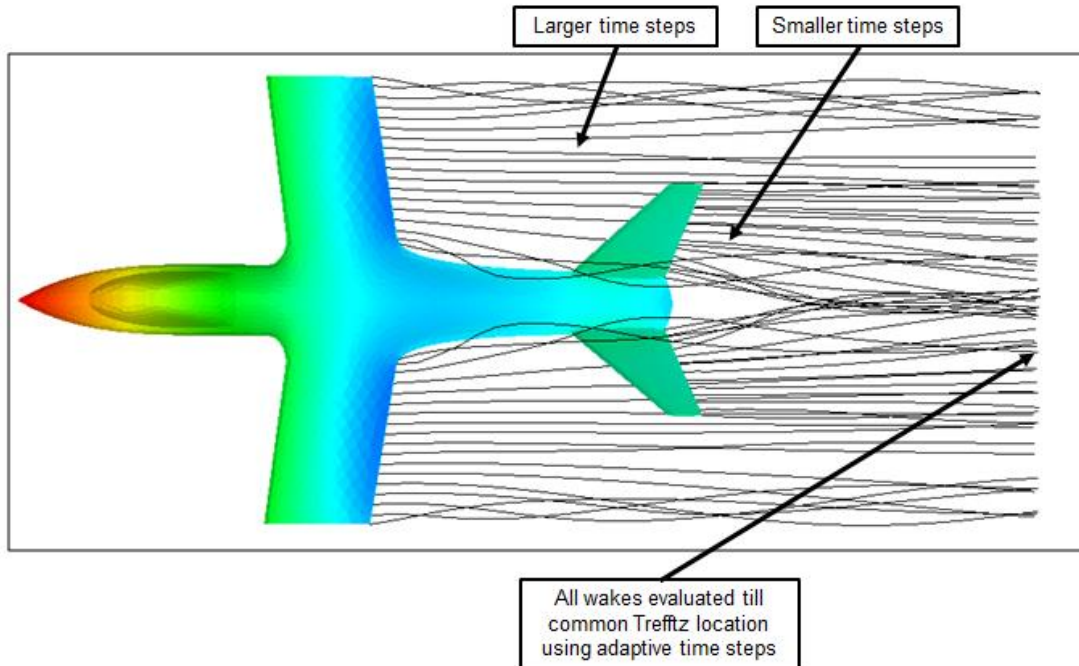
**Figure 4.3.2: Vorticity decay rates and velocity desingularisation using the Lamb-Oseen vortex core model**

The numerical propagation of the wake nodes from the trailing edges downstream is evaluated using this pseudo-time increment evaluated for each step of each strand. The unstructured nature of the strand-based wake allows for the evaluation of strand-specific pseudo-time marching. This allows for the local “compression” and “stretching” of individual strands so that all of them are propagated to the same Trefftz plane location downstream. The Trefftz plane location remains the only external user specification for the generation of the relaxed wake strands. Once the time Trefftz location is known and the starting node on the trailing edge is evaluated, the local steps from any node location at a given time is evaluated using Equation-4.3.4:

$$dt = \left[ \frac{X_{Trefftz} - X_{Seed}}{L_{avg}} \right] \times n_{\infty, x} \quad (4.3.4)$$

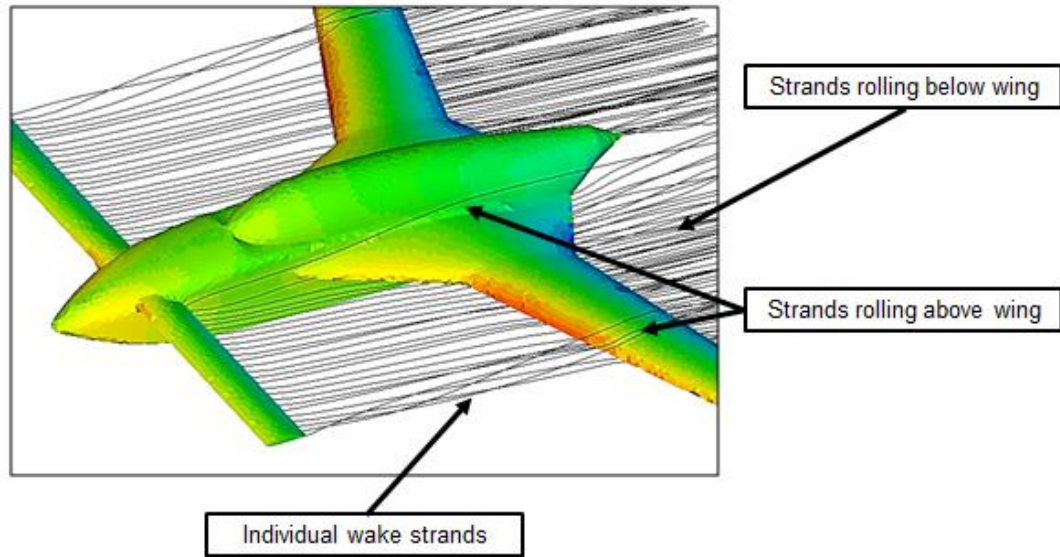
This allows for the time step for each propagation step to change depending on where the previous propagation has taken the node relative to the fixed Trefftz plane location and the local velocity conditions. The final propagation step is either truncated or projected to intersect with the Trefftz plane.

The time increment is thus adaptive and responds to each strand node individually. This is an advantage over wake-sheet methods since strands starting from differing positions on the geometry can be made to propagate to the same location downstream. This is illustrated in Figure-4.3.3.



**Figure 4.3.3: Adaptive pseudo-time stepping in the relaxed strand theory**

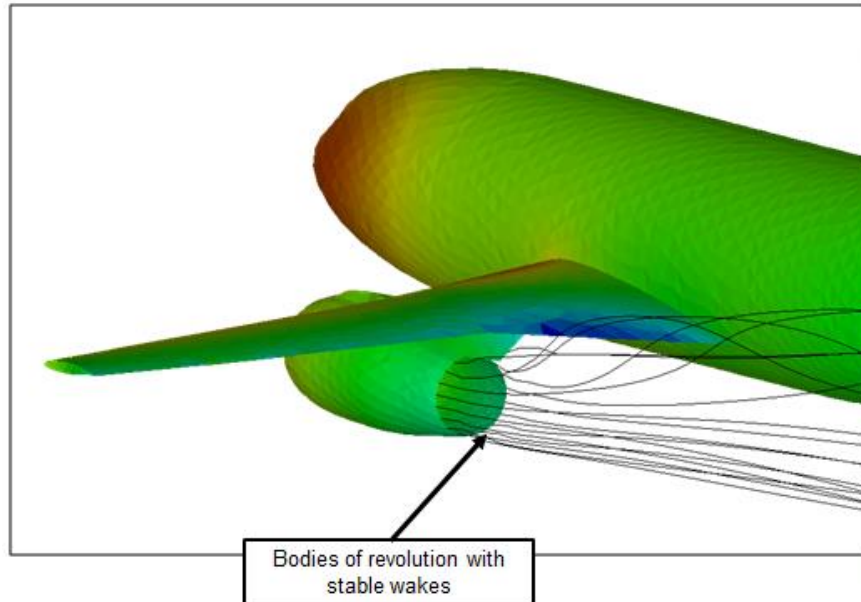
The wake strand model has several inherent advantages. The presence of adaptive time stepping allows for the elimination of all user inputs other than the downstream location of the Trefftz plane. User-specified time steps are no longer required. Further, the model allows the wake to conform to the geometry based on the local conditions. This is illustrated in Figure-4.3.4 for the case of the Rutan Varietze. The presence of the canard for this geometry creates relatively complex wake physics where certain strands follow their local physics and go above the wing whereas in other cases the go under the wing. Because each strand is a self-contained vortical entity, the wake-strand model allows such behavior to take place without problems.



**Figure 4.3.4: Local physics driven wake strands**

Another advantage allowed for by the strand model is the ability of the wake to be applied to bodies of revolution. This ability derives from the nodes based approach to vorticity shedding instead of the edge based models of prior wake-sheet models. A body of revolution has a trailing edge that rolls up on itself with the final node touching coincident in space with the first node. Since the wake-strand model makes no distinction as to the identity of vorticity contributing elements, the geometric relevance is unimportant. As a result, wake stability remains unaffected.

This is particularly useful for geometries with podded engines as shown in Figure-4.3.5 for the DLR-F6 geometry. Note the “pull” generated by the wing-bound vorticity on the engine wake as it propagates downstream. It is possible that future enhancements can be made in conjunction with volumetric sources to allow study of integrated aeropropulsive effects on wing design.



**Figure 4.3.5: Relaxed wake strands on bodies of revolution**

## **4.4 AERODYNAMIC LOADS**

Aerodynamic loads are evaluated as three components in the current approach:

- a) The induced lift from the Method of Integrated Circulation
- b) The induced drag from the Method of Integrated Circulation
- c) The skin-friction drag from the local vorticity distribution

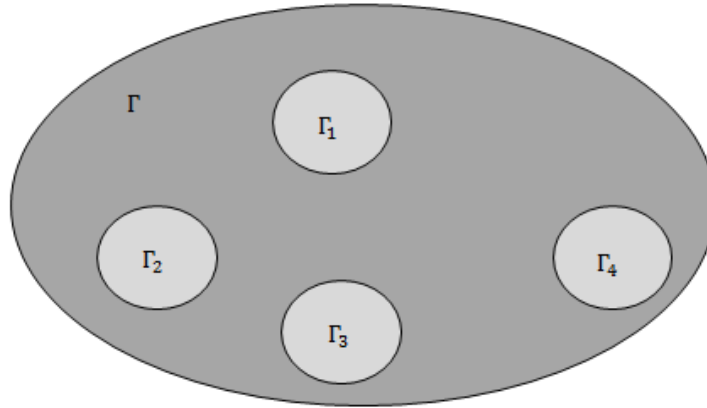
### **4.4.1 METHOD OF INTEGRATED CIRCULATION**

At the core of this approach is the extension of the basic laws on which potential-flow methods are based: linearization of aerodynamic effects <sup>14</sup>. The extension is easily



illustrated in two dimensions as shown in Figure-4.4.1. The net circulation in the large loop encompassing the series of smaller loops with their own circulation is evaluated as:

$$\Gamma = \Gamma_1 + \Gamma_2 + \Gamma_3 \dots + \Gamma_N \quad (4.4.1.1)$$



**Figure 4.4.1: Concept of integrated circulation**

This idea can be extended to three dimensions for a body of arbitrary surface tessellation. Assume that such a body has been solved for the local facet-bound vorticity using the vorticity solver described previously. Given that the tessellations are arbitrary, no application of the Kutta-Joukowski lift and induced drag equation can be made at the facets on this body, even though the vorticity values are evaluated correctly.

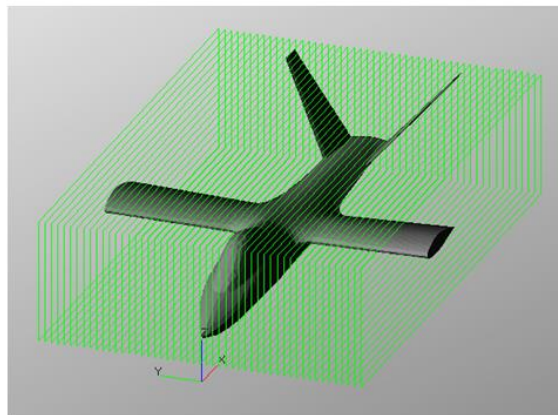
However, given the potential-flow environment around this body, the net spatial effect of all the facets on the body at any point in space is equal to the integrated effect of the individual facets in isolation. If a two-dimensional closed loop were created around this body in a plane perpendicular to the direction of flow, we could evaluate the net

circulation around this loop by determining the velocity at all of the loop's edges and then use the generalized equation for circulation <sup>14</sup> as:

$$\Gamma_k = \int_0^L V_{induced, l} \cdot dl \quad (4.4.1.2)$$

If this were a two-dimensional body, the net induced lift from the body could be evaluated using the Kutta-Joukowski theorem <sup>14</sup>. However, the body is three dimensional. As a result, several more sections must be created until the body is enclosed inside an effective bounding box of sectional planes, all of which are perpendicular to the direction of flow. An example of this approach is shown in Figure-4.4.2 for the Bell-Boeing advanced tilt-rotor geometry.

Each of these loops is referred to in this document as an Integrated Circulation Loop (ICL). Once these loops have been created around the geometry of interest, the Kutta-Joukowski equation for induced lift is applied for each loop and the lift is evaluated for each slice of the bounding box in the span-wise direction.



**Figure 4.4.2: Integrated circulation loops around a sample geometry**

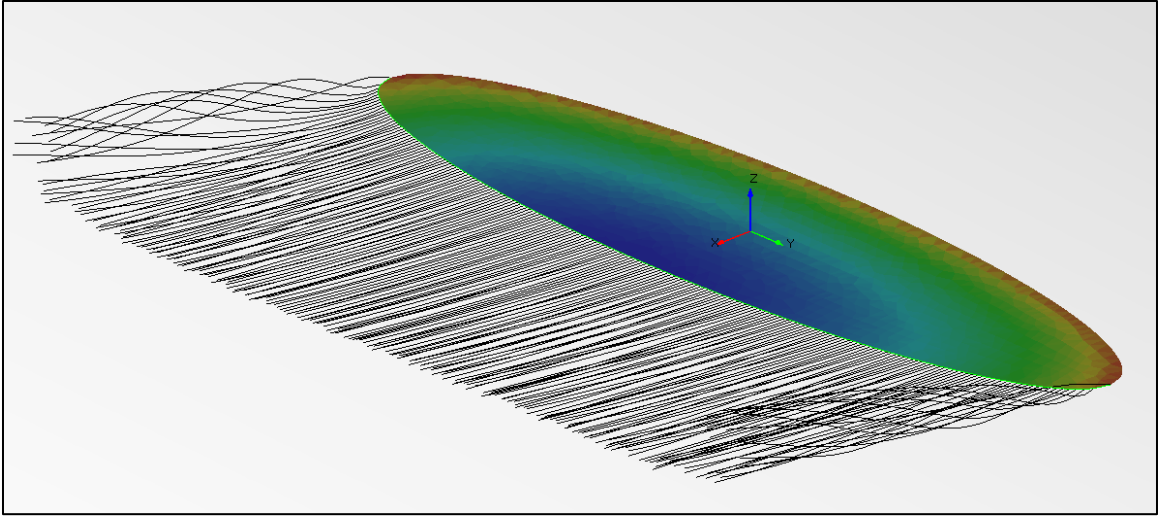
The presence of these closely spaced ICLs also allows for the evaluation of the composite span-wise circulation distribution and is then used to determine the induced downwash between the ICLs using standard lifting-line equations <sup>14</sup> and using the net circulation value of each loop as the “bound” circulation values. The induced lift and drag are thus evaluated as:

$$L_{induced} = \int_{-\frac{b}{2}}^{\frac{b}{2}} \rho_{\infty} V_{\infty} \Gamma_y dy \quad (4.4.1.3)$$

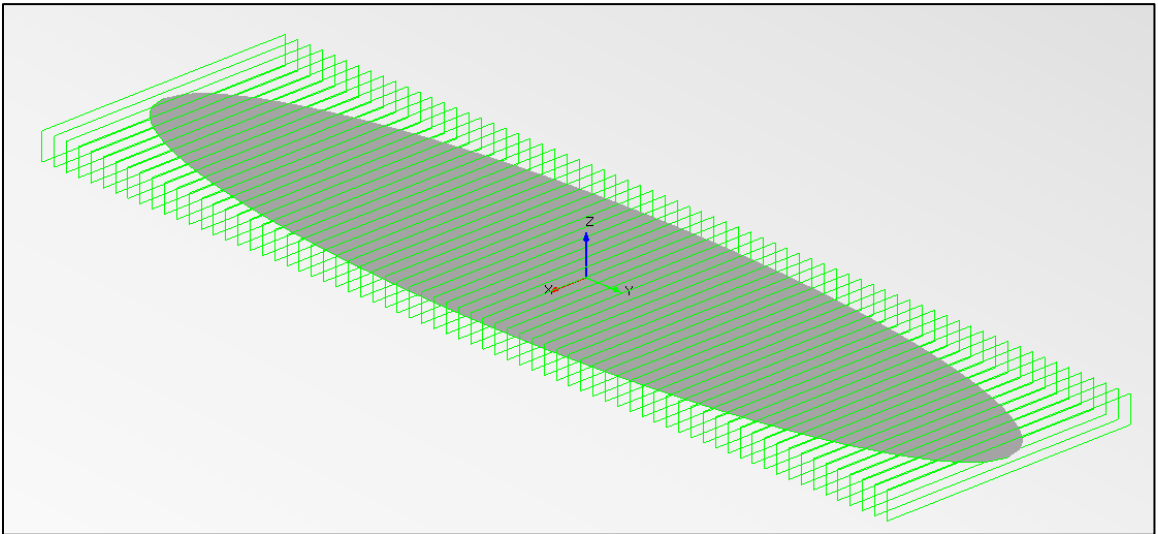
$$D_{induced} = \int_{-\frac{b}{2}}^{\frac{b}{2}} \rho_{\infty} w_y \Gamma_y dy \quad (4.4.1.4)$$

$$w_y = \frac{1}{4\pi} \int_{-\frac{b}{2}}^{\frac{b}{2}} \left( \frac{1}{y - y_0} \right) \frac{-d\Gamma_{y_0}}{dy} dy_0 \quad (4.4.1.5)$$

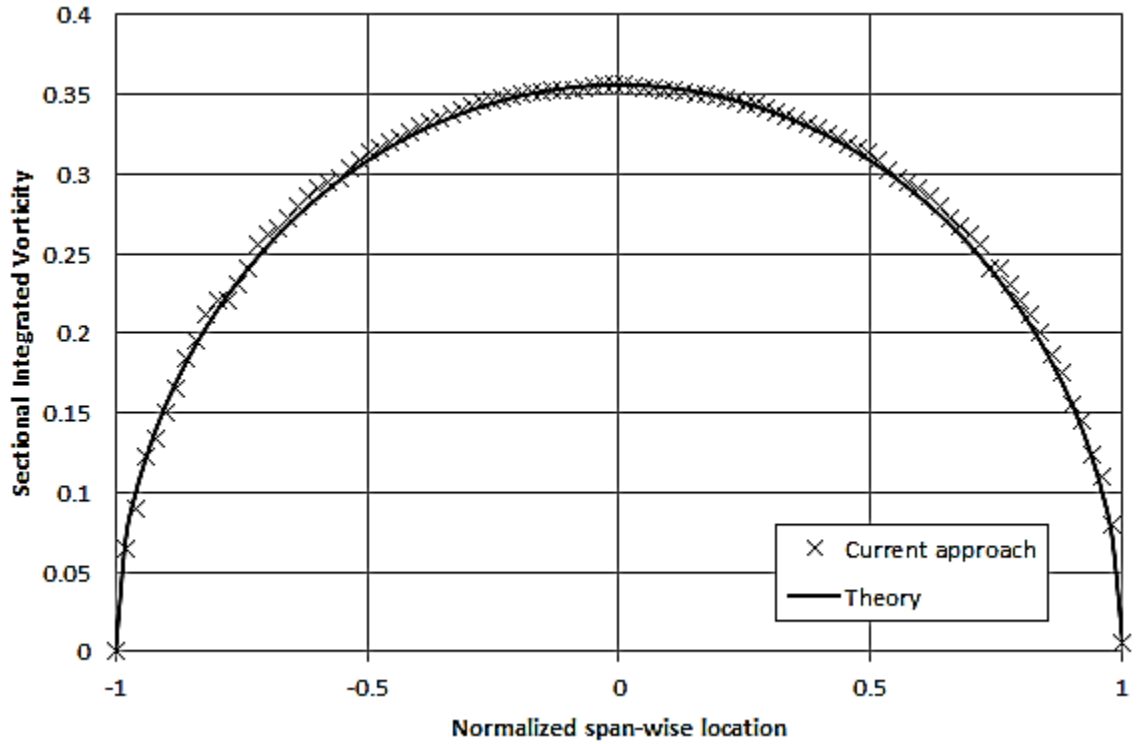
Examples of the composite span-wise distribution function are given for two geometries: an elliptical wing and the Boeing F-18A Hornet in order to illustrate the nature of the function and its relation to the underlying geometry. Figure-4.4.3 illustrates the results for the elliptical wing of aspect ratio 5.1 tested at 10° AOA. The composite span-wise sectional-lift distribution is shown in Figure-4.4.5(a). Given the elliptical nature of the wing plan-form, the expected nature of the circulation distribution is elliptical <sup>20</sup> in nature and matches the predicted data within numerical accuracy limits (Figure-4.4.5(b)). The current approach reduces to the concept of the lifting-line solution for flat wings in the limiting condition.



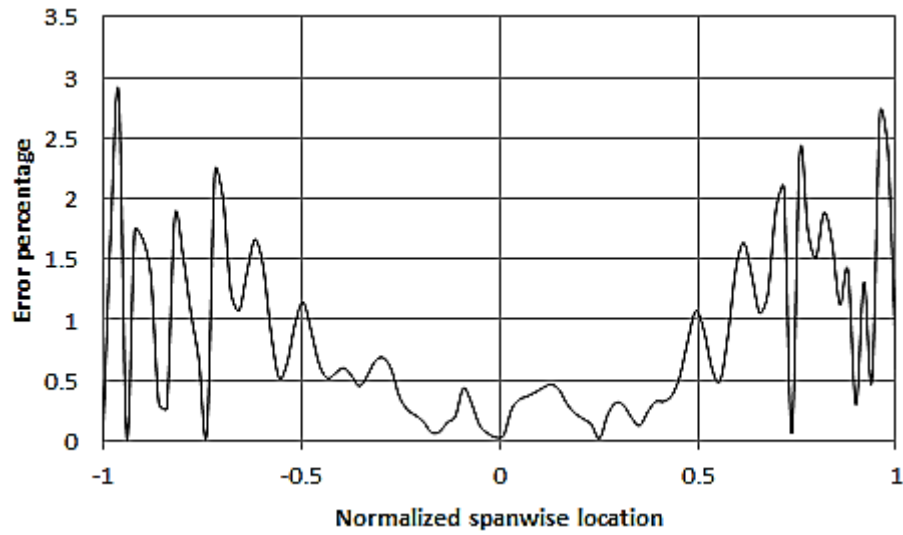
**Figure 4.4.3: The elliptical wing vorticity and wake solution**



**Figure 4.4.4: Integrated circulation loops around the elliptical wing**

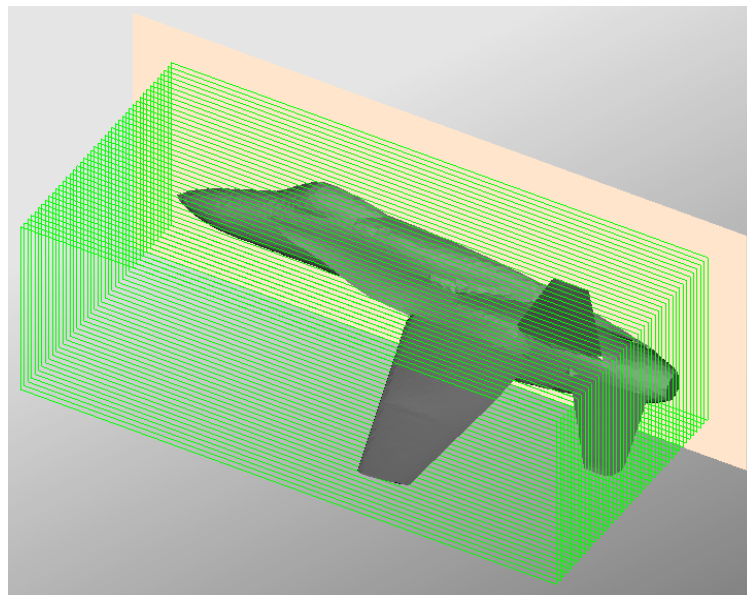


**Figure 4.4.5(a): Composite span-wise sectional lift coefficient for the elliptical wing**

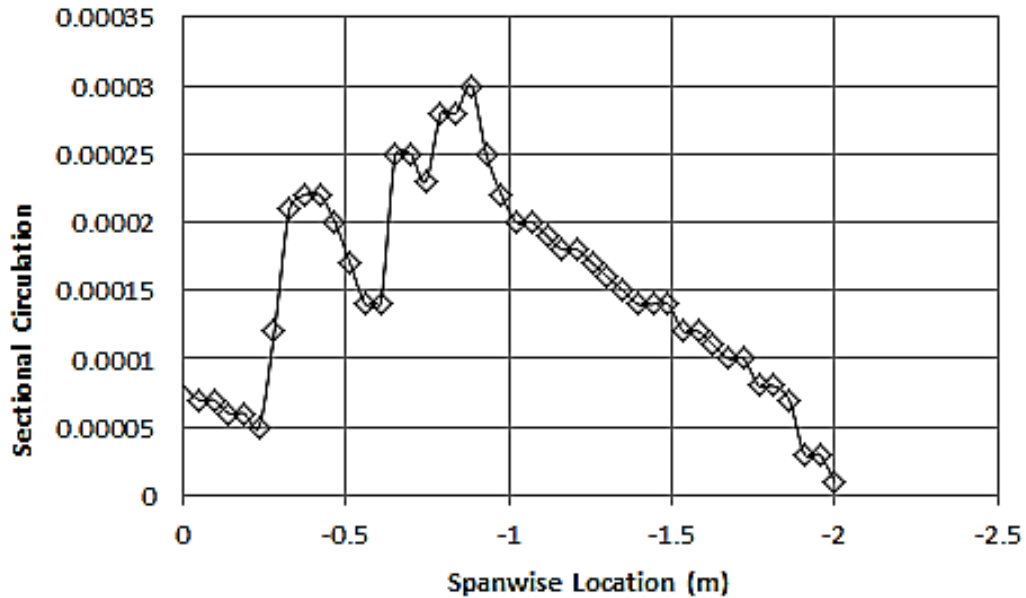


**Figure 4.4.5(b): Composite span-wise sectional lift coefficient for the elliptical wing**

Figure-4.4.6 illustrates the same process applied for the Boeing F-18A geometry. The geometry was evaluated at  $1^\circ$  AOA with a symmetry plane and velocity inlets for the engine intakes under the wing. The auto-mesher was used resulting in a fine mesh and smaller average edge-size. This led to the solver using 44 ICLs in semi-span direction as shown in Figure-4.4.6. The complexity of the combined main wing, stabilizers and LERXs is captured in the span-wise lift distribution shown in Figure-4.4.7. Note that the spikes and valleys in the distribution function correspond to the additive or subtractive effects of the horizontal and vertical stabilizers, the LERXs and then the fuselage near the span-wise zero location in Figure-4.4.7.



**Figure 4.4.6: Integrated circulation loops around the F-18A geometry**



**Figure 4.4.7: Composite span-wise sectional lift coefficient for the F-18A**

In numerical application, the ICLs are discretized using the average edge-size of the underlying surface mesh. The spacing between ICLs is also discretized using the same reference values. It is possible to include other discretization values and this can have a varying effect on the solution fidelity depending on the complexity of the geometry in question.

An example of a case-study is presented for a rectangle wing ( $AR=4$ ) for  $12^\circ$  AOA. The spacing was varied about the average edge-size by different multipliers and the overall coefficient of lift evaluated for each of these distributions. The results are plotted in Figure-4.4.8 and Figure-4.4.9. It can be seen from the figure that increasing the spacing between the ICLs beyond the average edge-size gradually reduces the fidelity of the solution where reducing the distance between ICLs (and hence increasing the total number of ICLs) improves the solution fidelity slightly.

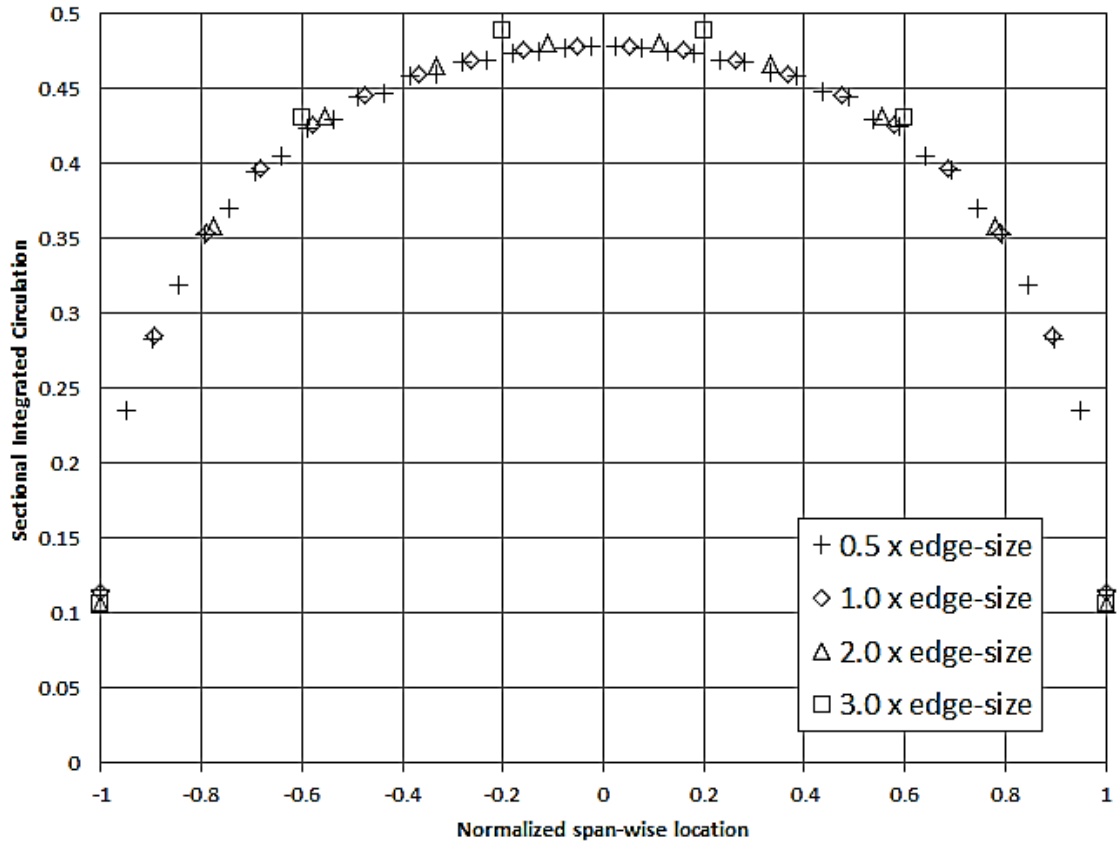


Figure 4.4.8: Span-wise integrated circulation for rectangle wing (AR=4, 12° AOA)

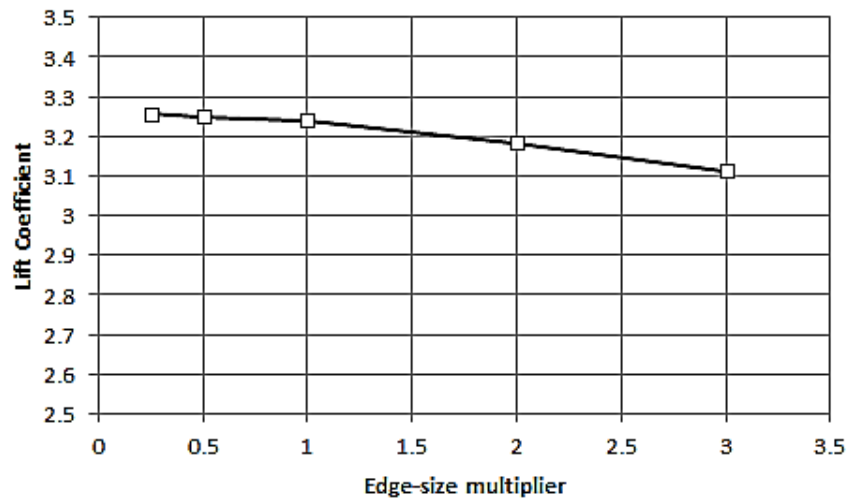


Figure 4.4.9: Lift coefficient as function of increasing spacing between ICLs for the rectangle wing (AR=4, 12° AOA)



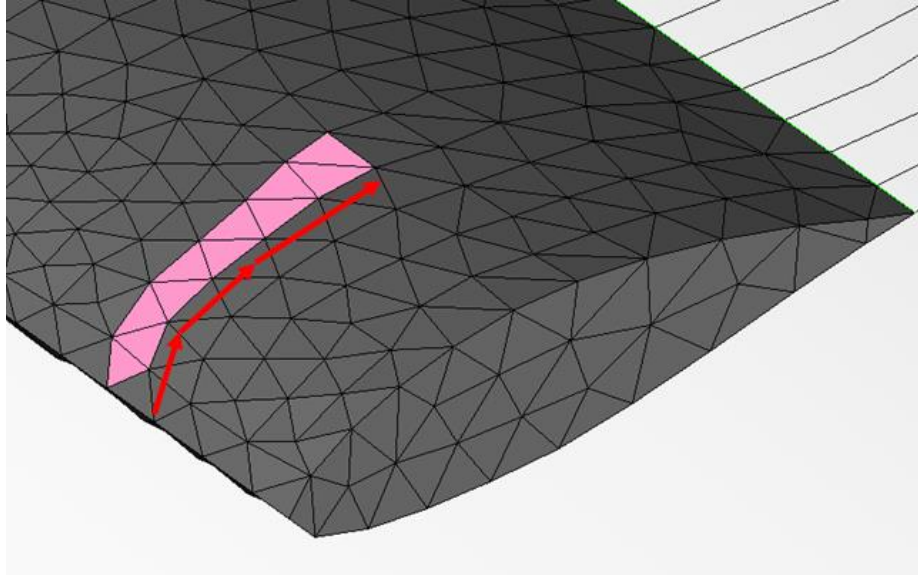
#### 4.4.2 SKIN-FRICTION DRAG EVALUATIONS

Evaluation of parasite drag in legacy potential-flow solvers has been based on the evaluation of the local Reynolds number for each facet <sup>2, 21-22</sup>. This local Reynolds number is then used as part of the semi-empirical skin-friction coefficient equations. The most popular of these equations for high speed subsonic aircraft was developed by Prandtl-Schlichting <sup>21-22</sup> in 1932 and was subsequently modified with the turbulent correction factor. This equation can be written as:

$$C_F = \frac{0.455}{(\log_{10} Re_x)^{2.58}} - \frac{1700}{Re_x} \quad (4.4.2.1)$$

The application of this equation in a pressure-solver was made possible by the combined use of structured meshes and pressure-field evaluations for calculation of aerodynamic loads <sup>8-11</sup>. The knowledge of local velocity and distance from the leading edge allowed local evaluation of the skin-friction coefficient for each facet that was then summed up for the net integrated value <sup>8-11</sup>.

With the current approach, the local velocity is not determined. However, the local distance from the leading edge can be determined by topological “walking” for each facet towards the closest leading edge in the direction of the free-stream (Figure-4.4.10).



**Figure 4.4.10: Topological “walking” used to determine B.L. growth lengths**

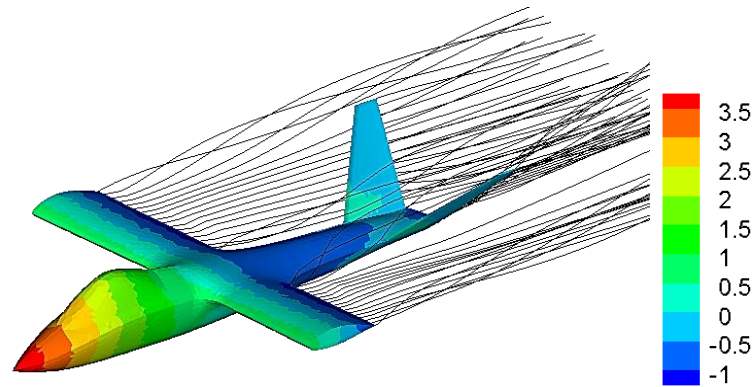
Additionally, the local vorticity is known in lieu of the local velocity. As such, the equation for the local Reynolds number is heuristically modified thus:

$$Re_x = \frac{\rho_\infty V_\infty \left( x + \frac{\gamma}{V_\infty} \right)}{\mu} \quad (4.4.2.2)$$

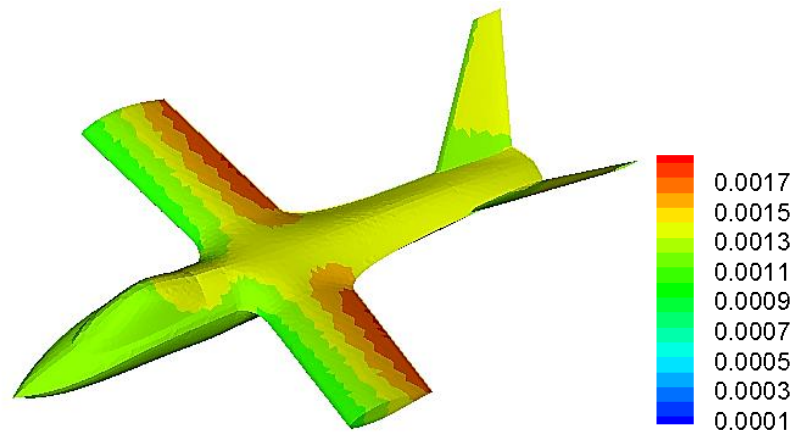
This model is applied after the determination of the local facet vorticity following convergence of the solution. The model follows certain heuristic arguments that were used in its formulation:

- a) When the facet is parallel to the flow, the vorticity on it is zero and equation 4.4.2.2 reduces to the standard formulation for a flat plate.

- b) When the facet is at the leading edge, the local vorticity dominates the skin-friction evaluation since the distance from the leading edge is a very small value
- c) The dependency of the local Reynolds number on the surface vorticity means that the distribution patterns for the skin-friction coefficient closely resemble the vorticity distributions. This is also expected from theoretical arguments. An example of this is shown in Figure-4.4.11 for the case of the Bell-Boeing Advanced Tilt-rotor geometry.



**Figure 4.4.11(a): Vorticity distribution for the Bell-Boeing tilt-rotor (5° AOA)**



**Figure 4.4.11(b): Skin-friction coefficient for the Bell-Boeing tilt-rotor (5° AOA)**

## **5. MESHER METHODOLOGY**

This Section outlines the automation adopted for the unstructured surface mesher developed for this effort. The quantification of various heuristics and empirical parameters that control the overall surface mesh and help convert an input geometry into a solver ready surface mesh are also discussed. Finally, the Section will outline the importance of sizing and quality control on the solver mesh and the inherent robustness of the current approach to stressful input geometries.

### **5.1 SIZE BASED MESHING**

One of the opportunities provided by unstructured meshes is the ability to control sizing and sizing gradients for the surface facets at the local and global levels. This is of great importance in CFD simulations because of the volumetric cell based solver physics. In potential-flow solvers, the importance of mesh facet sizing varies depending on the choice of solver. Pressure-solvers are significantly more sensitive to the sizing and sizing gradients very similar to CFD solvers. This is not surprising in lieu of the similar nature of their aerodynamic load evaluators. As mentioned in previous Sections, capturing the pressure field accurately is of paramount importance for such solvers and this is not

possible unless the mesh is appropriately refined based on the local surface curvatures. Furthermore, surface patches making up the geometry must be conformal along the interface edges. This is due to the greater sensitivity of such solvers to gaps or punctures in the surface topology. If the solver is fitted with a boundary layer model, no such gaps can be allowed and the mesh surface across patches must be conformal at all times. Such conformity and sizing can be enforced through the careful development of an underlying U-V mapped structured mesh and is the reason for widespread use of such surface meshes with legacy potential-flow pressure-solvers.

Vorticity-solvers are much more resilient to such limitations and afford greater flexibility in the sizing and sizing gradients. Effective solutions can be obtained even from meshes with sharply varying sizing gradients along the topology. Mesh conformity is not a necessary condition for solution fidelity. However, maintaining conformity along patch interfaces and a uniform sizing gradient greatly helps in faster solution convergence (boosted by lower oscillations in surface vorticity along solver partition interfaces as also because of higher quality facets).

The savings in computation time due to sized based meshing therefore translate directly to faster runs within an MDO environment and therefore merit further investigation. Despite these advantages, there are certain pitfalls that must also be investigated. The primary disadvantage of size based meshing for vorticity-solvers is that it can sometimes defeat the very advantage that makes this approach MDO friendly. This can happen by the liberal use of size-based meshing over the entire surface geometry, which can significantly increase facet count in regions of low curvature where the

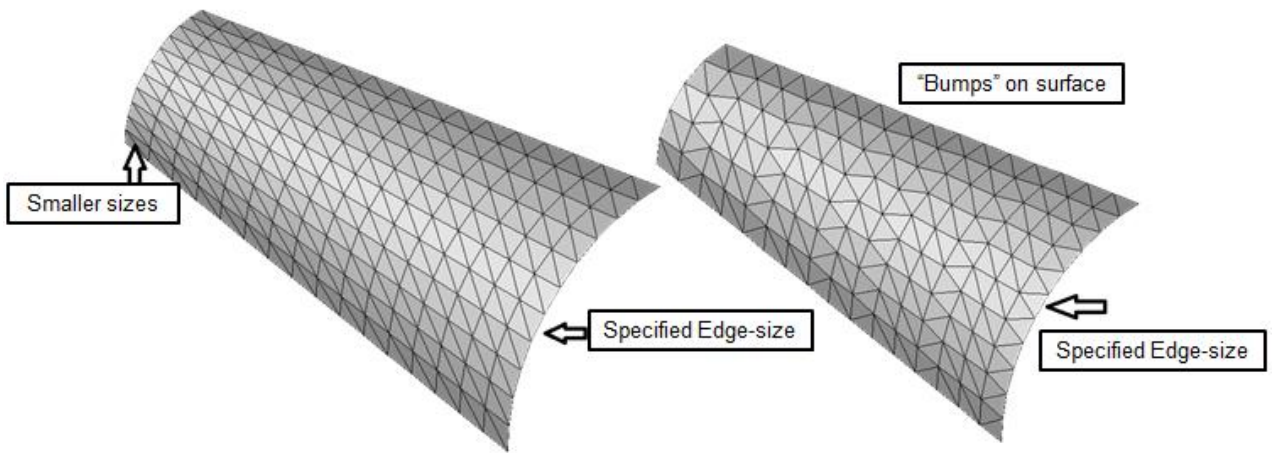
vorticity gradients are low and insignificant. Surface re-meshing, as brought about by the use of size based meshing is also particularly invasive to the smoothness of the initial surfaces and can create “bumps”. However the vorticity-solvers are generally insensitive to local surface bumps while the Method of Integrated Circulation concept allows it to smooth out the effects of such topological perturbations. Finally, unless care is taken and appropriate metrics used, redefining the surface tessellations can adversely affect surface curvature and must therefore be carefully employed.

Two case studies of sized based meshing applied to initial structured meshes elaborate on the above concepts. The mesher developed for this effort is discussed later but its results are used here for illustration purposes.

For the first study, a quadrant of a truncated cone was generated with a structured mesh made of up quad facets that was then split into triangles. The resulting “forced” unstructured mesh is shown in Figure-5.1.1. The initial sizing of the facets was set up so that the base of the cone had facets of appropriate edge-sizes. Note that this initial surface has all the inherent advantages and disadvantages of a mapped structured mesh. There are no surface perturbations and the feature lines are clean. Also note that the U-V mapping forces poor quality facets near the tip of the truncation edge of the cone and that the edge-sizes of the facets in this region are much smaller than the requested edge-size.

Consider now the re-tessellated surface generated by the size-based mesher developed for this effort (Figure-5.1.1). The sizing distributions are much more uniform across the surface and especially near the tip of the truncation edge. However, note that

the feature lines on this surface are distorted and as a result there are surface perturbations (this can be evaluated in Figure-5.1.1 by the lighting distribution on the two surfaces from a single point source away from the surface). A comparison of the average edge-sizes and the final facet-count is shown in Table 5.1.1. The reduction in facet count is more than 50% while the improvement in edge-sizes created by the mesher is within 5% of user requested values compared with 40% on the original surface.



**Figure 5.1.1: Size based meshing case study for a truncated cone surface. Initial surface mesh (left) and re-tessellated mesh (right)**

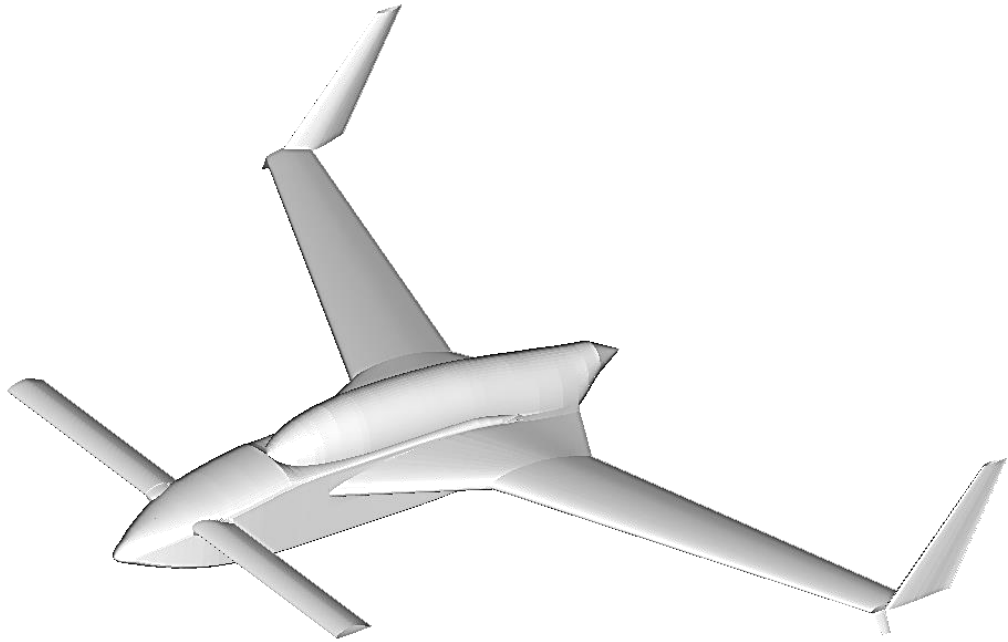
**Table 5.1.1: Size based meshing results for the truncated cone surface**

	Initial surface	Re-tessellated surface
Number of facets	352	160
Average edge-size (expected = 0.2m)	0.28	0.21

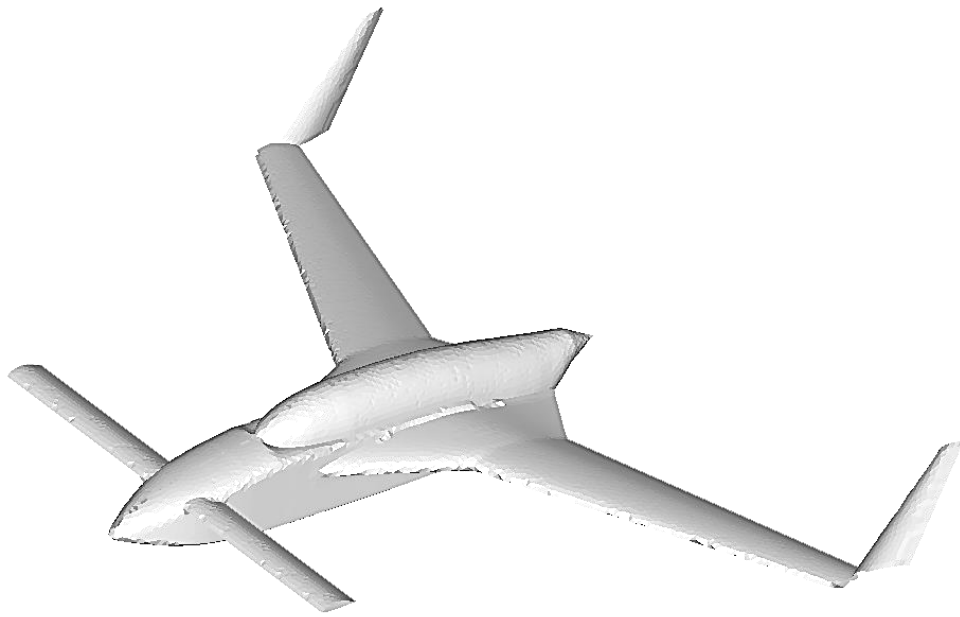
A second case study was made on much more realistic application geometry. The Rutan Varieze civilian aircraft was run through the mesher and the results for the aircraft are shown in Figures-5.1.2(a) and Figure-5.1.2(b). In Figure-5.1.2(a), the original surface prior to re-tessellation is shown. This mesh was generated from a forced conversion of the underlying structured mesh which resulting in the starting surface of 108,040 facets. The geometry once again had clean feature lines and little to no surface perturbations. The large number of facets for this surface resulted from the very large needle-shaped facets distributed over the entire surface to allow effective pressure-gradient capture from a potential-flow pressure-solver.

This surface was then run through the mesher with an aim to convert the surface into one more appropriate for a vorticity-solver. This meant that the sizing gradients could be improved in favor of reducing computation time and feature curves could be dispensed with where necessary. The result of this was process is shown in Figure-5.1.2(b). As can be observed from this figure, the overall “essence” of the geometry has been preserved by the mesher but at the cost of much more pronounced surface perturbations. Such a surface as shown in Figure-5.1.2(b) is not considered solver-ready for any pressure-field solver or any CFD solver but is usable by the current approach without loss of fidelity. The reduction in facet count is an indicator of reduced computation time outlined via Equation-3.3. The new facet-count is 10,036 facets (a reduction of over 90 %) compared with a structured mesh. The savings in computation time is over two orders of magnitude.



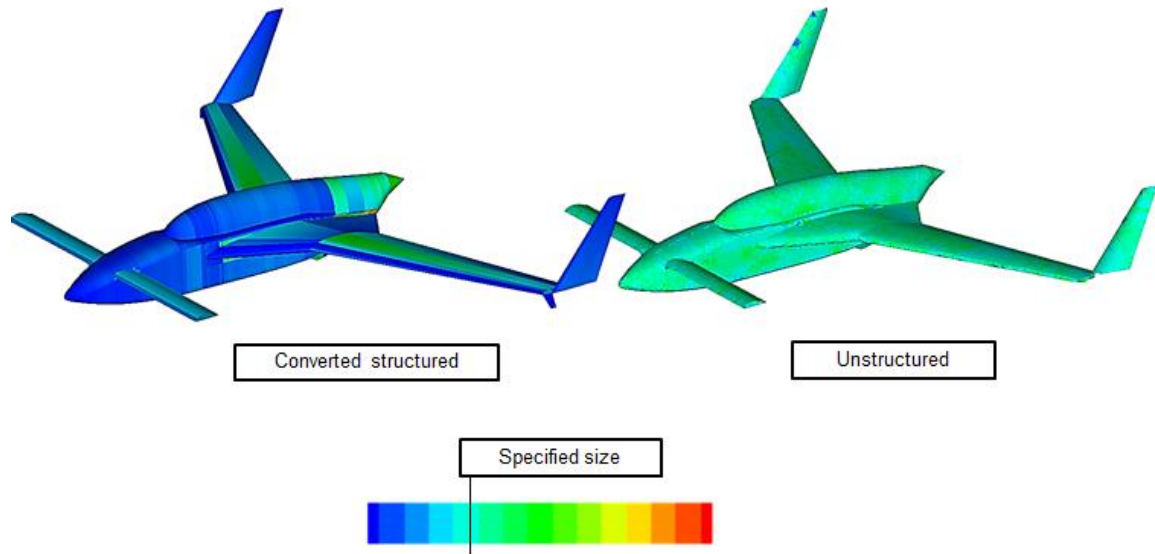


**Figure 5.1.2(a): Initial pressure-solver mesh for the Varienze (108,040 facets)**



**Figure 5.1.2(b): Re-tessellated vorticity-Solver mesh for the Varienze (10,036 facets)**

A further illustration of the sizing distribution and gradients for this case study can be made via Figure-5.1.3. Here, the contour distribution of the mesh facet sizes is shown for the initial and final meshes on the Rutan Varieze. The comparison is qualitative but apparent.



**Figure 5.1.3: Mesh sizing distribution and gradients**

## 5.2 MESH QUALITY DEFINITION

The current effort quantifies mesh quality at the facet level and evaluates them at an area-averaged level for the overall surface mesh. At the facet level, the quality criterion (developed by the author) is defined as:

$$Quality = \frac{Radius\ of\ circumcircle}{Radius\ of\ incircled\ circle} \quad (5.2.1)$$

The Circumcircle and Incircle for a triangular facet is readily defined in two dimensional Euclidean space based on the facet edge lengths and is thus applied to three-dimensional facets similarly. As a result of this, Equation-5.2.1 can be modified thus:

$$Quality = \frac{\left[ \frac{L_{12}L_{23}L_{31}}{4\sqrt{s(s-L_{12})(s-L_{23})(s-L_{31})}} \right]}{\sqrt{\frac{(s-L_{12})(s-L_{23})(s-L_{31})}{s}}} \quad (5.2.2)$$

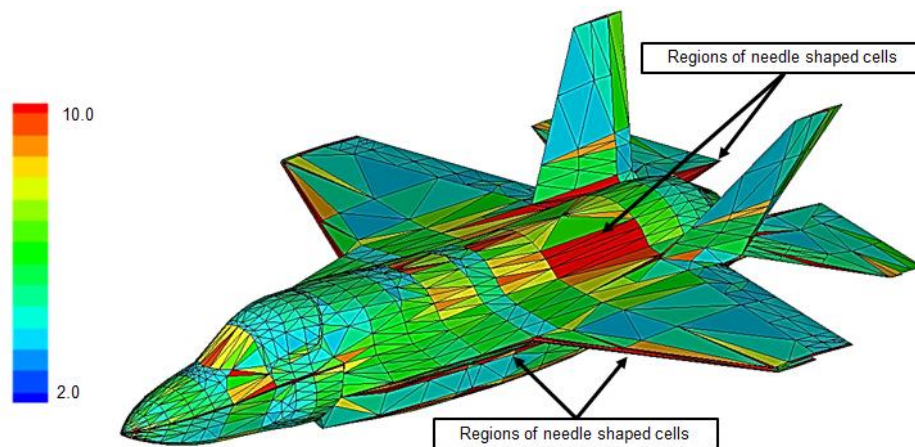
Where,

$$s = \frac{(L_{12} + L_{23} + L_{31})}{2} \quad (5.2.3)$$

The limiting value for the quality ratio described in Equation-5.2.1 is 2.0 for an equilateral triangle. Note that this value represents the confluence of solver vortex ring requirements in that an equilateral triangle has the most appropriate placements of the centroid-based control points and therefore the resultant positive effect on solver convergence and reduced computation time. This lower reference value therefore serves as the metric for “best-possible-quality” for a given facet on a surface.

As a result, the mesher looks to this metric to determine which facets must be culled from a quality standpoint during the re-tessellation phase. That is, the mesher determines a range of quality above this reference value which it considers as recoverable and amenable for improvements. All facets having quality values beyond this threshold are culled from the mesh via edge collapsing (to retain conformity of neighbors).

Since the quality of a facet is geometrically defined, visual evaluation of overall mesh quality serves as a useful step for the current solver process. An example is shown in Figure-5.2.1 for the Lockheed-Martin F-35 geometry. Run through the mesher prior to re-tessellation, the “quality-distribution” of the mesh can be visualized. In this way, mesh regions needing facet culling or re-tessellation can be identified and either manually repaired or run through the mesher.



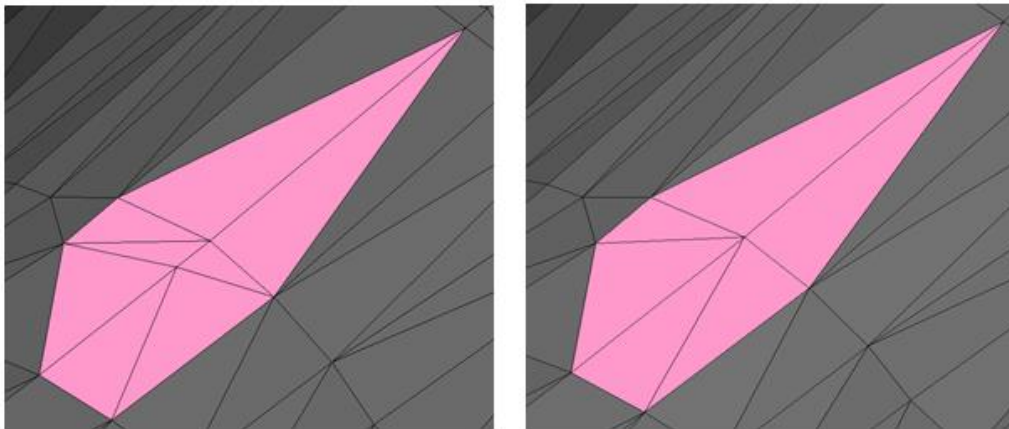
**Figure 5.2.1: Unstructured mesh facet quality distributions**

### **5.3 MESHING AUTO-REFINEMENT**

The automated unstructured mesher developed for this effort uses several invasive surface improvement techniques to render a size- and quality-controlled solver-ready mesh. The individual techniques applied at the facet or patch level are:

- a) Edge-collapsing
- b) Edge-Swapping
- c) Facet-splitting

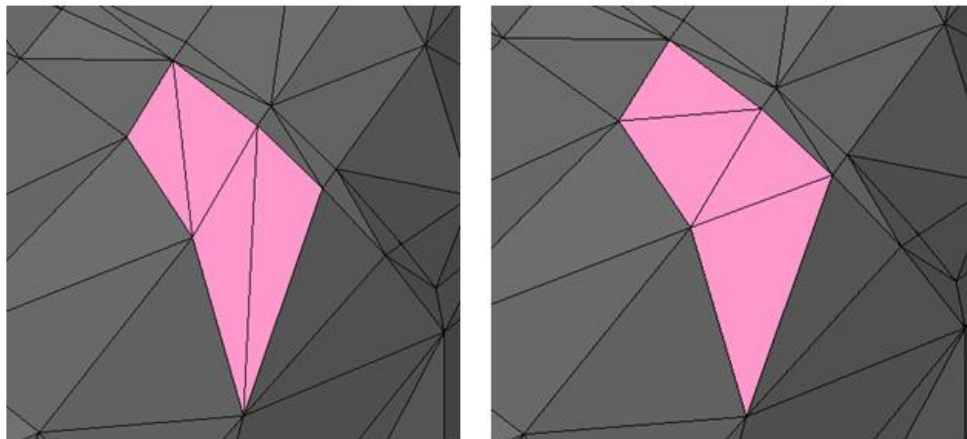
Edge-collapsing is applied on those facet edges that are deemed too small to allow recovery of facet sizing and/or quality. An example is shown in Figure-5.5. The smallest edge in the facet is identified and the corresponding vertex connectivity map is determined from the global topology. The two vertex pairs are then collapsed and the topology adjusted accordingly as shown in Figure-5.3.1. Edge-collapsing serves to eliminate very poorly sized facets or ones with very poor quality (above the mesher recoverable threshold) while also helping to increase the overall size of the surrounding facets. Of the three surface modification techniques used in the mesher, it is the only one that increases the sizes of facets and reduces overall mesh size. Checks are also made to ensure that edges marked as feature edges based on their neighborhood connectivity are not subject to edge collapsing. This is done to ensure that no overall loss of geometrical “essence” takes place.



**Figure 5.3.1: Edge collapsing on below-threshold facets. Before (left), after (right)**

Edge-swapping is a purely quality control technique and is applied iteratively in conjunction with the Edge-collapsing and Facet-splitting. The general principle is to

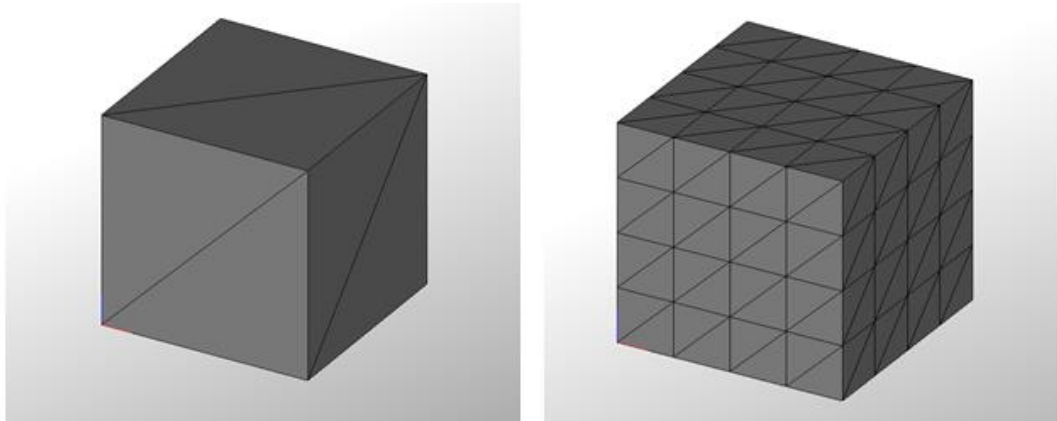
identify pairs of neighboring facets from the surface topology and identify their shared edge. If this edge can be flipped with resulting increase in quality of the two facets, the mesher allows this. Checks are made to evaluate the surface-normal angle across the shared edge and determine if this value is greater than the user defined threshold for active feature edge modifications. If so, the edge is not swapped and the feature edges are preserved. An example is shown in Figure-5.3.2 for two facet pairs. The resulting improvement in quality is visible. Note that the swapping subroutine actively looks to ensure that the final summed area of the two facets does not increase in order to prevent flipping edges that would lead to topologically inconsistent or self-intersecting facets in the post swapping phase. Note that edge-swapping is the only active quality-control technique in the mesher's tool set. All other techniques may improve quality based on secondary criteria, but otherwise enforce no checks for quality.



**Figure 5.3.2: Edge swapping on facet pairs, before (left) and after (right)**

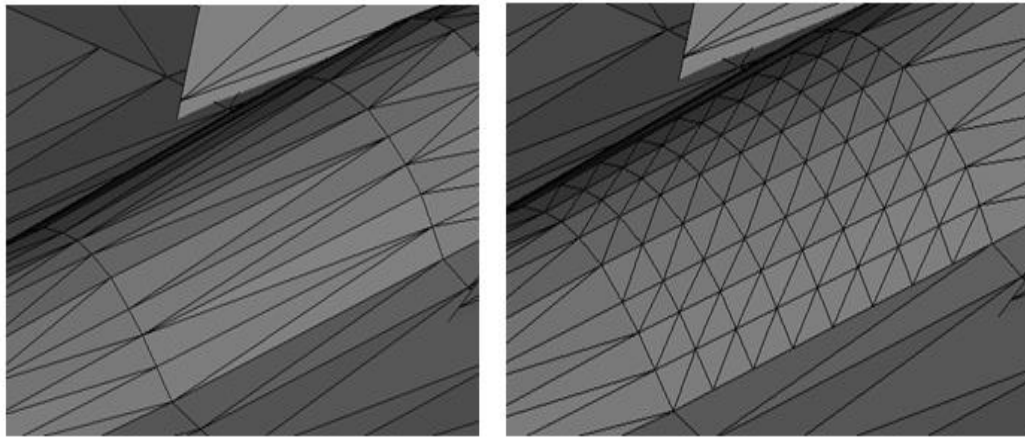
Facet splitting takes place in one of two ways: at the facet level or at the patch level. At the facet level, the mesher identifies facets that are above the allowable sizing

threshold and marks them for splitting. Topological sharing information for the three edges of the facet are recovered from the global map and the neighboring facets are also marked for splitting along the shared edge to enforce mesh conformity. Note that the splitting, whether at the facet level or the patch level, is designed to always enforce conformity on the surface unless none exists initially. Once this information is extracted, the mesher splits the edges in digital steps, i.e. the splits are dual along each edge and are not matched for exact sizing. This is enforced in order to prevent the formation of very poor quality, needle-shaped facets that can prove irrecoverable by the edge-swapping or edge-collapsing routines. Figure-5.3.3 shows an example of splitting applied by the mesher at the facet level for a simple cube designed with just 12 facets initially. Note the substantially higher facets in the resulting mesh in Figure-5.3.3. This is an inherent feature of facet splitting: there is always an increase in mesh size. As a result, only patches where the solver requires greater refinement are subjected to these routines. Note also that facet splitting does not respect feature edges in its effort to maintain mesh conformity.



**Figure 5.3.3: Individual facet-splitting, before (left) and after (right)**

Facet splitting at the patch level is more advanced and sizing-driven. In this routine, the mesher actively looks for topological information and uses this to identify patches of the initial tessellation that can be suitably split along the feature lines. Once identified, it splits the feature lines in exact sizing steps (unlike the digital splits used in the facet-level splitting) so that feature-aligned surface meshes can be extracted. This is illustrated for an application made to the F-35 upper-fuselage surface from Figure-5.2.1 and is shown in Figure-5.3.4. This is the only technique in the mesher's tool set to reduce overall feature-curve distortion in the final mesh compared to the initial surfaces.

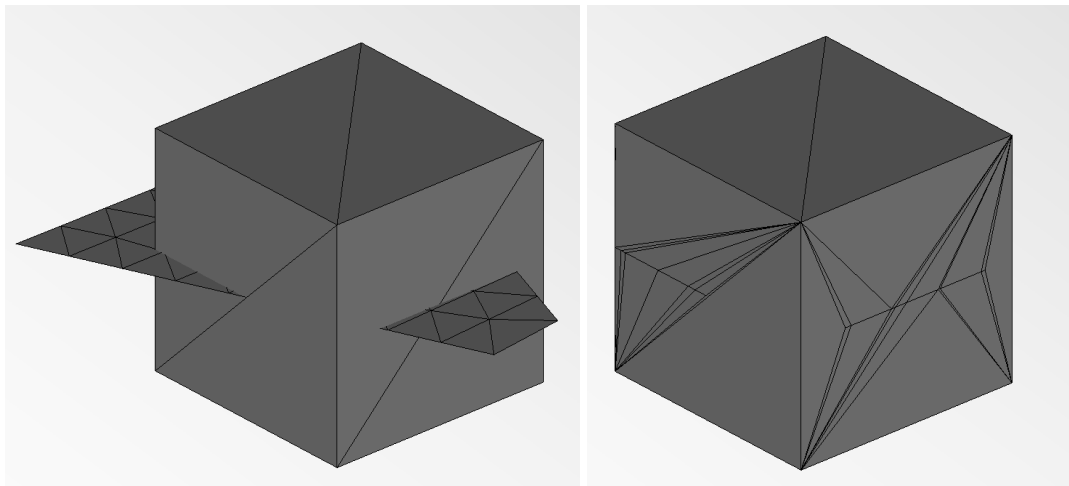


**Figure 5.3.4: Patch-splitting, before (left) and after (right)**

Pierced facets are corrected by the application of splitting at facet level for the two intersecting pairs. The splitting point inside the facet is determined by the intersection of the edges of the opposite facet with the plane of the other facet. Once this point is evaluated, it is added to the global vertex map and the facet is split into three new facets about this point. This process retains the overall mesh conformity and the global edge-connectivity map is added upon but otherwise not changed. This process is applied



on both facets of the pair and the process is repeated until no further piercings exist on the pair. An example of this is shown in Figure-5.3.5. Here, a flat finely-tessellated surface is shown cutting a coarsely tessellated cube. This results in numerous piercings as visible from Figure-5.3.5. The post piercing-correction results are also shown and the increased facet count can be seen as well as the cutting edges that are not visible. The new facets are mainly of poor quality and further iterations of the solver are suitable candidates for edge-collapsing and edge-swapping.



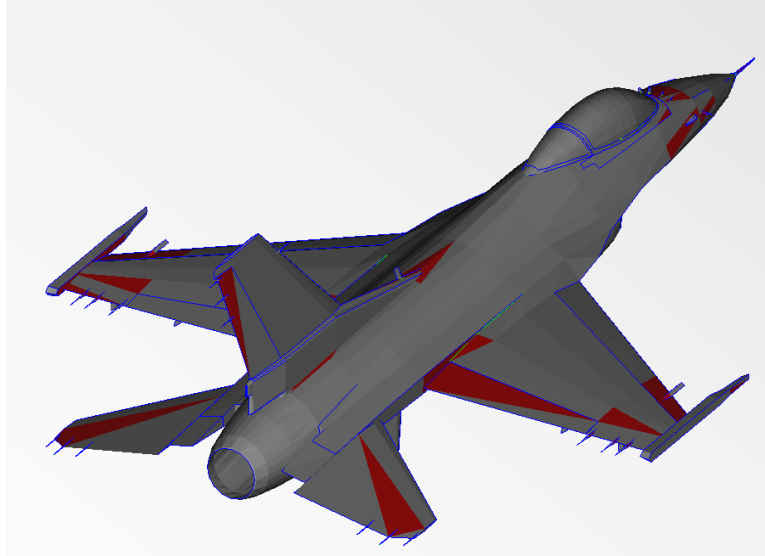
**Figure 5.3.5: Piercing correction, before and after (cutting surface hidden)**

All of these techniques are applied in an iterative loop that spans every facet on the original mesh and changes are recorded. The mesher continues to operate on the mesh until it records no further changes applicable to the geometry at which points it terminates operations and moves the mesh to the solver for pre-simulation diagnostics. The liberal application of this mesher on the entire geometry can result in significant mesh-size increases, distortion of feature-edges and de-featuring of smaller components (that fall

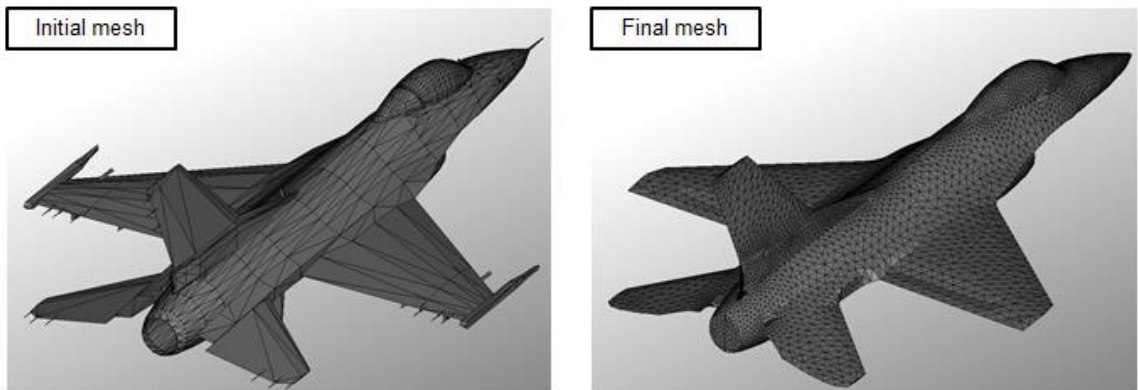
entirely below the user specified sizing threshold). This can, of course, be used to advantage so that non solver-ready meshes with high geometrical fidelity can be reduced to a solver-ready state around the essence of the overall geometry.

A case study of the auto-mesher is provided here for the F-16A geometry. Figure-5.3.6 shows the initial geometry as run through the solver diagnostics. Facets marked in red are pierced and can cause instant solver divergence. Blue edges are feature edges evaluated by the mesher through topological probing. Note that feature edges have been detected on this geometry even over relatively smooth surfaces because of the underlying non-manifold edges that are not visible in Figure-5.3.6. Edges that are non-manifold are instantly marked as feature edges by the mesher and as such either need to be removed manually or the auto-mesher can leave them in place. Green edges are free and represent the perimeter of free-edges and non-conformal interfaces. Facets marked in orange show proximal facets that will also cause instant solver divergence.

It is easily visible from Figure-5.3.6 that the initial mesh is highly articulated and contains a lot of electronics antennae and other empennages that are below the sizing threshold of the rest of the geometry and can be dispensed with (each of these empennages can cause solver divergence since they are not topologically consistent with the rest of the geometry; i.e. they pierce other facets or are in numerically bad proximity of each other). This repair can be accomplished via laborious manual means or through the use of the automated mesher developed here. The results are shown in Figure-5.3.7 and the quantitative mesh comparison is presented in Table 5.3.1.



**Figure 5.3.6: F-16A initial geometry pre-solver topological probing results**



**Figure 5.3.7: F-16A case study results for automated mesh refinements**

**Table 5.3.1: F-16A automated mesh refinement results**

	Initial Mesh	Re-tessellated Mesh
Number of facets	4504	13964
Average facet quality	8.3491	2.2253
Worst facet quality	4579.36	106.52

## 5.4 PATCH INTERFACES AND CONFORMITY

For this effort, patch interface conformity is defined as the connectivity of facets at the feature edges of a surface topology. These interfaces are created during the CAD phase of the geometry design process but can also result from user defined feature curves as well as brought about within an MDO environment by the intersection of topologically independent bodies such as wing-fuselage intersections, wing-pylon intersections etc. Applied to potential-flow solvers at the surface level, they are the equivalent of CFD solver volume mesh partition interfaces and have philosophically similar effects on the solver convergence.

These interfaces are handled in one of two ways:

- a) Mesh side handling
- b) Solver side handling

In the mesh side handling, the mesher (volumetric for CFD and surface for potential-solvers) is responsible for generating conformal meshes along the interfaces. The solver simply inherits the connectivity information from the mesher during the solution phase. Maintaining conformity on the mesher side enforces mesh quality penalties that can adversely affect the solution.

In the solver side handling, the interface conformity is ignored by the mesher and therefore meshes the topological patches in a standalone manner. The solver is then left to determine the cross-connected facets and neighborhood connectivity. This can result in increased computation time and increased solver oscillations depending on the quality of

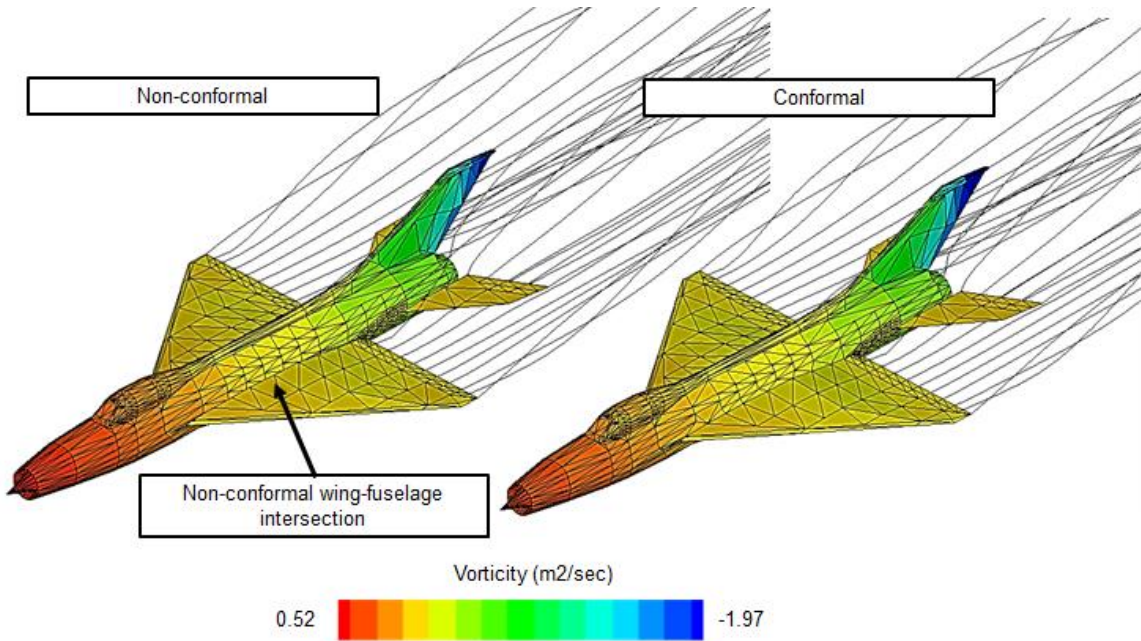
the interfaces. The choice of either of these two approaches to interfaces affects the solver significantly.

The vorticity-solvers and the Method of Integrated Circulation are largely insensitive to the surface presence or absence of mesh conformity at the patch interfaces with regard to the final aerodynamic loads. However, the solver convergence (and thus computation time) is still dependent on mesh conformity.

The solver developed for this effort handles the patch interface conformity on the solver side. As such, the mesher is allowed to identify and mesh patches in standalone format. This reduces the complexity of the application of this approach to an MDO environment whilst improving flexibility in the design space since each patch can be meshed individually. Given the first-order approach of the vortex-ring solver applied to the facets, it was determined that the solver is generally stable around non-conformal patches under controlled sizing gradients (applied by the surface mesher described in previous Sections).

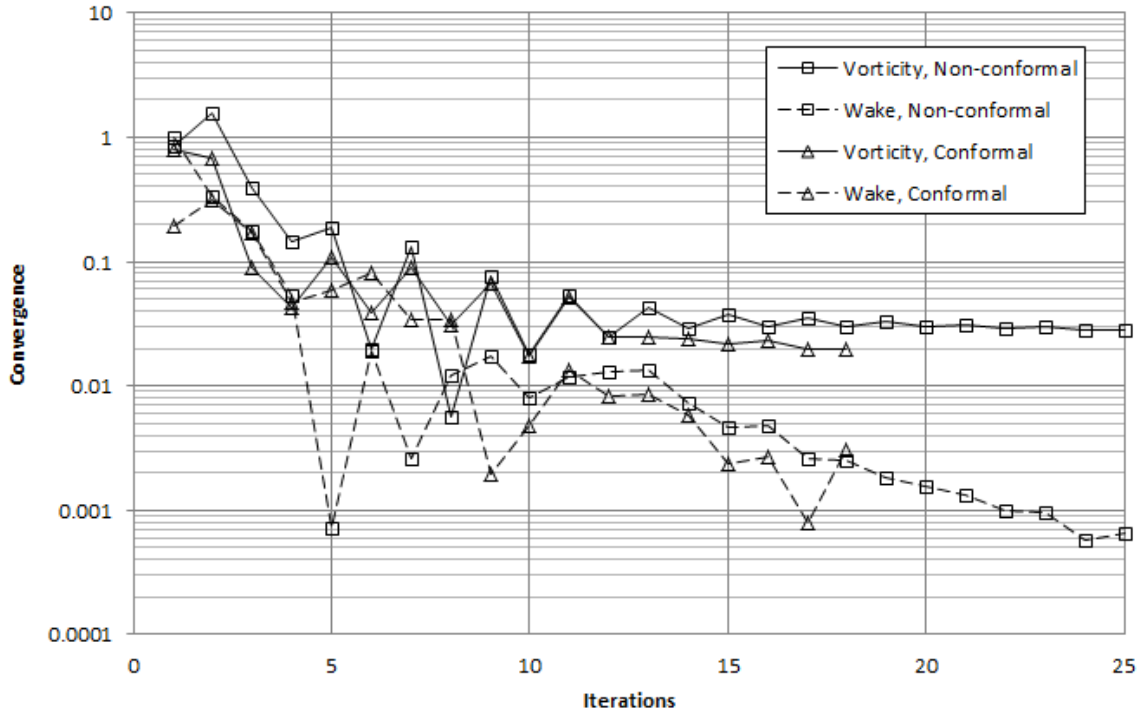
A case study for the stability of the solver under conformal and non-conformal patches is presented here to illustrate the effects. The Soviet Mig-21 interceptor geometry was chosen for this study as shown in Figure-5.4.1. The wing surface was made conformal for one case and non-conformal for the second case (Figure-5.4.1) and both meshes were run through the current solver under identical conditions. The solution for vorticity and the relaxed wake strands were seen to converge in both cases and were found to be very close to each other (see Figure-5.4.1). The final solution therefore had

no apparent effect by the conformity of the wing-fuselage intersection. The integrated aerodynamic loads evaluated using the Method of Integrated Circulation seems to confirm this assumption in that far field effects are indeed diminutive on the final results.



**Figure 5.4.1: Mig-21 mesh conformity case study results**

However, such visual inspection of the vorticity distribution and the qualitative comparison of the integrated loads can be deceptive to the overall effect on the solver. To illustrate this, the convergence history for the above case study is provided in Figure 5.4.2. Plotted as a function of iterations, the dramatic effect of mesh conformity is easily visible on both the vorticity convergence as well as the wake-strand stabilization. From the figure it can be seen that the conformal design undergoes substantially more damped oscillations compared to the non-conformal mesh and the convergence takes place at a more rapid rate.



**Figure 5.4.2: Mig-21 case study convergence history results**

Similar results have been obtained from other geometries as well that suggest that solver stability is adversely affected by the non-conformity of patch interfaces but otherwise have little effect on the overall fidelity of the integrated aerodynamic loads.

## **6. SOLVER EFFICIENCY ENHANCEMENT**

The basic vorticity-solver methodology utilizing structured meshes does not offer significant flexibility to improve its basic performance. However, with the current approach making it possible to apply unstructured meshes, several important techniques have been developed to take advantage of the increased solver and mesh flexibility to improve solver efficiency, stability and to reduce computation times. The novel techniques developed for this effort are:

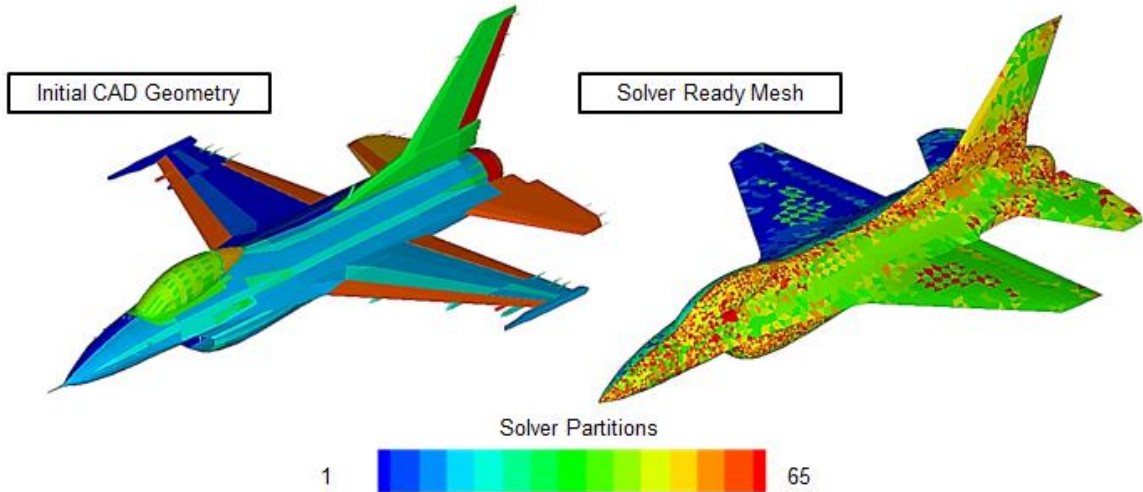
- a) Fluid-Dynamic Mesh Repartitioning
- b) Persistent Geometrical Induction Matrix

### **6.1 FLUID-DYNAMIC MESH REPARTITIONING**

The unstructured surface meshes post CAD-driven tessellation and post auto-meshing (as described in Section-5 of this document) can render a diverse spread on the surface facet distributions with regard to numerical space. In other words, the unstructured nature of the mesh operations, whether part of CAD design, manual surface repair or the auto-mesher, leads to facets that are numerical neighbors being widely separated in physical space. This is illustrated in Figure-6.1.1 for clarity. The F-16A



geometry used here is illuminated for numerical neighbors using the contour coloring and shows the mesh post-CAD and post-auto-mesher.



**Figure 6.1.1: Rationale behind FDMR**

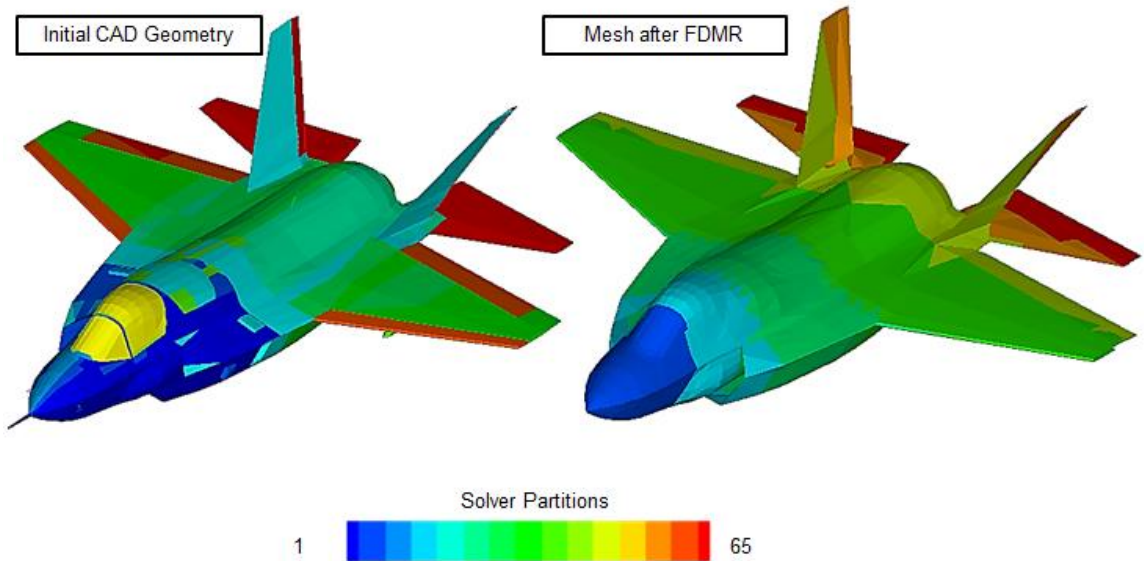
The widely spread nature of the numerical neighbors are readily appreciated from Figure-6.1.1. When passed to the solver in this condition, the iterative patches are scattered and have physical neighbors that are not part of the current iteration's solution (and hence have vorticity values frozen from the previous iteration). As a result, the solver undergoes substantial stress. Once the solver stabilizes and the oscillations dampen out, the vorticity and wake solutions converge quickly to their final form.

It has been observed from several geometries that the oscillation phase of the solver for a typical mesh as shown in Figure-6.1.1 constitutes about 40-50% of the total solver iteration time. If these oscillations could be damped out prior to start of solution, the resulting convergence would be substantially faster and more stable. Further, if the solver partitions could be organized so that partitions containing the leading edges and

sharp nose sections were evaluated first, the “pass-on” effect of the solver during each iterative loop could lead to quicker convergence as well.

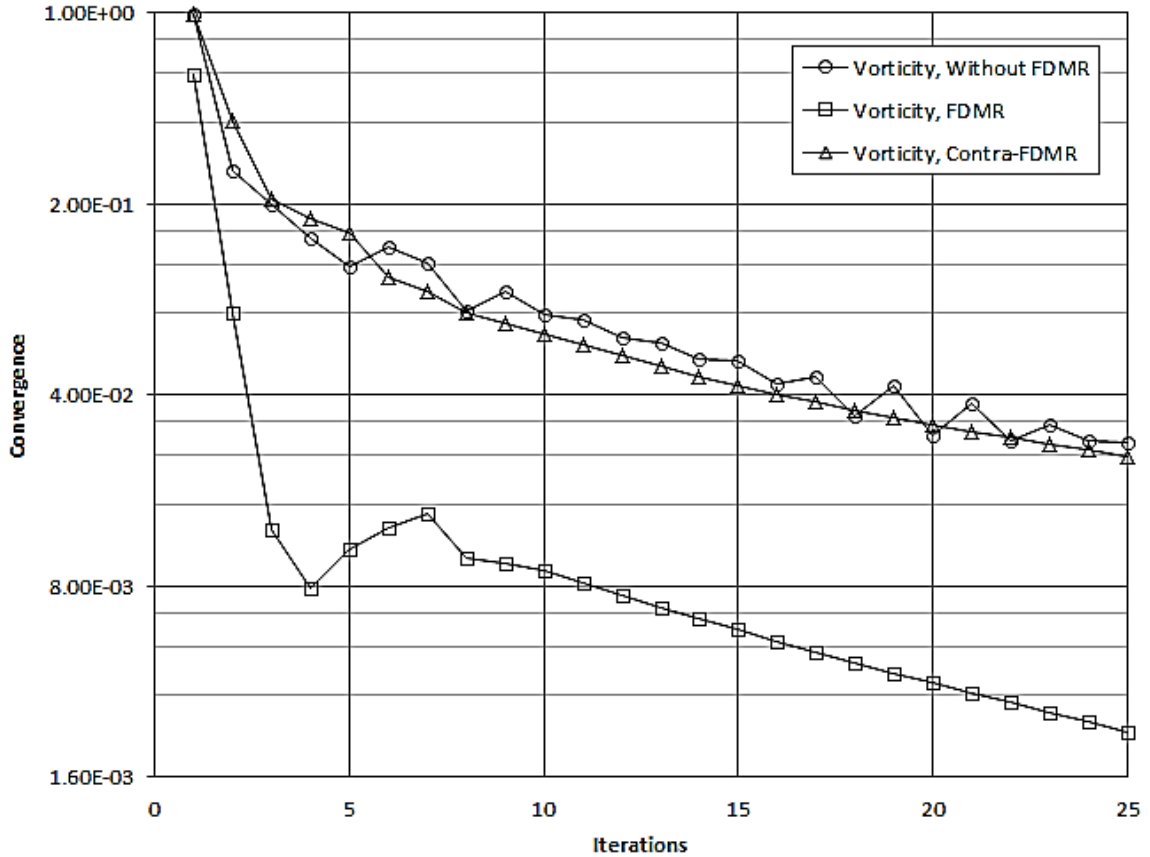
To resolve this issue, the Fluid-Dynamic Mesh Repartitioning (FDMR) concept was developed and applied to the solver. This process repartitions the solver-ready mesh developed by the auto-mesher or provided by the user and reorganizes the numerical distribution of the mesh to match the free-stream direction. The sorting algorithm used here is a standard bubble-sort routine and the direction of the bubble-flow is in the free-stream direction. In some cases it is beneficial to sort the partitions in the direction normal to the flow and this has also been developed and added to the solver as a user defined option called Contra-FDMR.

Figure 6.1.2 shows a case study for the application of the FDMR concept on the Lockheed-Martin F-35 geometry. This geometry was passed directly from the CAD to the solver post-FDMR (i.e. no auto-meshing refinement was used). As a result, it retains the original CAD tessellations during the solver run. This was done to ensure that the only change between various runs was the numerical re-ordering of the solver partitions. An example of FDMR application to the geometry is given in Figure-6.1.2 for flow in the vehicle X direction (long axis). The comparison of the solver partitions is clear and it is seen that the FDMR partitions provide the required “pass-on” numerical behavior that is anticipated to give desired reductions in solver computation times.



**Figure 6.1.2: The F-35 geometry before and after FDMR**

The geometry was run through the solver without FDMR, with FDMR and with Contra-FDMR for otherwise identical conditions. The vorticity solution for the geometry with FDMR is shown in Figure-6.1.3 and shows a clean convergence. Figure-6.1.3 shows that the FDMR solver convergence is an order of magnitude lower than the standard mesh and the Contra-FDMR results. In terms of computation time, the results are even more significant. The FDMR solution reaches the same convergence level of the non-FDMR solution (reached in 25 iterations) within 4-5 iterations. That is a reduction of ~80% in overall computation time during vorticity convergence phase of the solution. It should be noted that the loads evaluation phase for all three cases would remain unaffected by the application of FDMR. Also, FDMR imposes its own time penalty based on the mesh size. It is usually between 1-5% of the overall solver run time. Combining all of these factors, the effective reduction in solver time for the FDMR case is between 60-70%. This value is still very significant and bears importance during MDO runs.



**Figure 6.1.3: Convergence history for the F-35 solution (AOA = 5°, 1 m/sec flow)**

## 6.2 PERSISTENT GLOBAL INDUCTION MATRIX

As outlined in Section-4, the velocity induction coefficient matrix is identical during all iterations in the absence of geometrical warping or morphing. Under these conditions, it is possible to reduce the evaluation time for this matrix by storing it in the solver memory allocations after the first complete iteration. This approach reduces the computational expense for the solver and it merely has to access the predetermined memory locations to evaluate the geometrical induction coefficient of a given facet-

bound vortex ring on any other facet. This coefficient is then multiplied by its vorticity value from the latest iteration.

Note that this does not apply to the relaxed wake strands since their spatial relaxation does not allow persistent storage of their induction effects of the geometry control points. The wake coefficients are evaluated incrementally despite the associated computational expense. The wake evaluations therefore become the limiting condition for the scalability of this feature for large geometries.

## 7. BOUNDARY PHYSICS

The unstructured nature of the surface mesh used in the current effort makes the application of external boundary physics on local regions a simple process. Single facets can be marked individually or as part of topological patches. For the solver developed as part of the current effort, there are four main boundary conditions and associated external physical conditions that are applicable on the surface facets:

- a) Slip walls
- b) Velocity Inlets
- c) Trailing edges
- d) Symmetry

### 7.1 SLIP WALLS

Slip walls are the default boundary conditions for all facets in the solver for the current effort as a carryover from all legacy potential-flow solvers. The current application takes the form of the Neumann condition <sup>2, 14</sup> on the control point of each facet (placed at the centroid of the facet). This condition states that the fluid flow normal to the direction of the facet at the control point must be zero. Unlike the other sister

condition for slip walls based on the Dirichlet condition <sup>14</sup>, the Neumann condition remains applicable for both thick and thin surfaces and thereby helps maintain solver robustness <sup>14</sup>. Legacy pressure-solvers were usually provided with the Dirichlet condition as the default slip-wall condition given the requirement for thick closed surfaces for pressure evaluations <sup>8-11</sup>. Mixed-condition solvers were also used. However, vorticity solvers applying combination of thick and thin surfaces have always necessitated a Neumann condition approach <sup>2</sup>. It also has a simpler implementation.

The main disadvantage of the Neumann condition is the inability to carry or enforce any boundary layer information and thereby serves to decouple the viscous and inviscid effects, necessitating the requirement for a decoupled skin-friction load evaluation scheme, as outlined in Section-4.

## **7.2 VELOCITY INLETS**

Velocity inlets are simulated on the user-marked surface facets by relaxing the Neumann condition <sup>14</sup> on the control points. The relaxation of the Neumann condition allows the modeling of various types of inlet flows such as compressor fan faces of podded engines, air-intakes on the fuselage and cooling-intakes at various points on the geometry <sup>14</sup>.

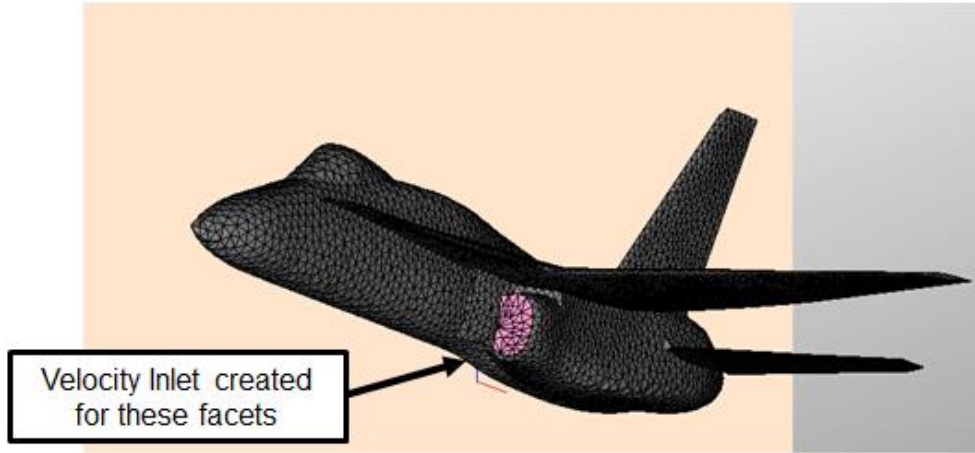
The Neumann condition is provided the required velocity increment magnitude and direction vector components during its application in the solver. As a result, the solver ensures that no surface normal component of velocity other than the predetermined

external value can exist on the marked facets. It achieves this by modifying the local vorticity.

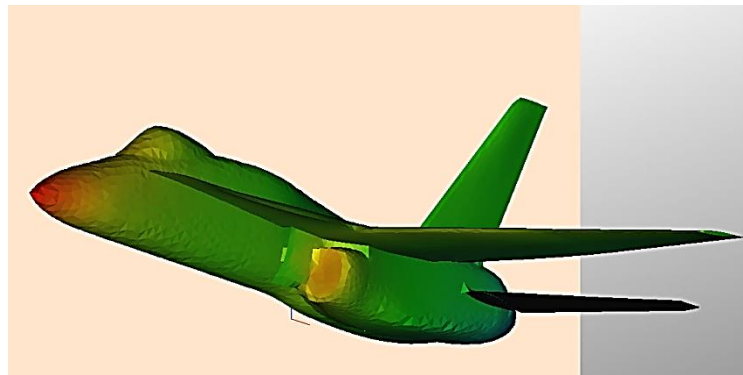
Note that the Neumann condition only allows the modification and control of the surface normal velocity on the facet. It does not enforce any control on the local tangential components of the velocity (borrowing from its slip-walls lineage). As a result, relaxation of the Neumann condition to simulate velocity flow can only accurately portray the normal velocity components whilst ignoring the tangential components.

An example of the application of velocity inlets is provided for the case of the Boeing F-18A Hornet (Figure-7.2.1). The geometry is modeled about the symmetry plane and has facets marked across the engine inlet next to the fuselage as shown in the figure. Two cases were run with the solver for this geometry for an AOA of  $0^\circ$ , symmetry plane applied and 25 iterations limit. The first case did not simulate the velocity inlet and treated the facets on the engine inlet as standard slip-walls (Figure-7.2.2). The second case relaxed the Neumann condition on the marked facets and simulated flow moving through the intakes (Figure-7.2.3). Notice the difference in the local vorticity distribution for the two cases. The under-wing vorticity is significantly affected by the presence or absence of the velocity-inlet and this translates to small changes in the integrated aerodynamic loads. The overall effect varies from geometry to geometry and must be exercised accordingly.





**Figure 7.2.1: The F-18 geometry with velocity inlets and symmetry plane**



**Figure 7.2.2: The F-18 geometry without Neumann relaxation**



**Figure 7.2.3: The F-18 geometry with Neumann relaxation**

### 7.3 TRAILING EDGES

Facets at the trailing edges of wings, fins and stabilizers are marked accordingly so that the solver can determine the wake nodes on the mesh (as discussed in Section-4). As such, they remain the most important aspect of all potential-flow solvers, regardless of the solver methodology. The solver is set up to check marked facets for trailing edges as well as doing a global search on the surface mesh to determine trailing edge facets. In the former approach, the user marks the facets on the mesh manually to treat them as a trailing edge. This has the advantage of giving extensively customized solutions. However, for an MDO environment, this is clearly unsuited. Further, lacking the U-V mapping of structured meshes, the current solver cannot rely on pre-marked information for determining trailing edges on the mesh.

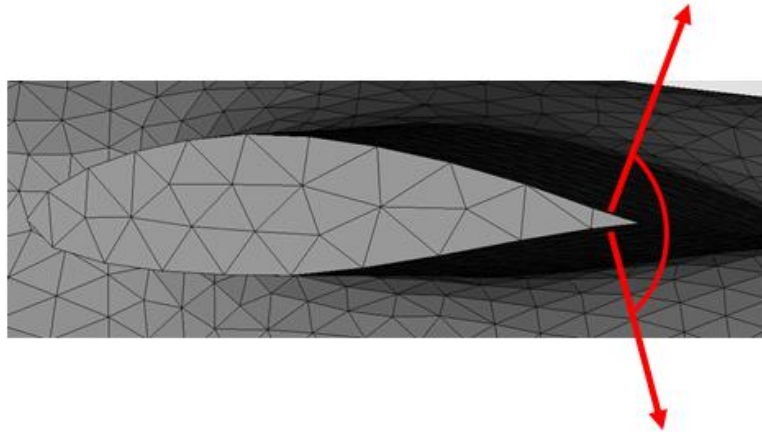
As mentioned in Section-4, the primary advantage available from the relaxed strand model is the ability to evaluate wakes at the level of strands. As a result, the determining characteristics of the trailing edge facets can also be evaluated at the facet level in the mesh and then pooled together to determine the wake nodes.

The current approach is based on the formulation of trailing edge metrics. The solver checks the facets in the entire mesh to determine which of the facets pass the testing criteria for trailing edges and automatically marks them as trailing facets. There are two main criteria for a facet to be a trailing edge:

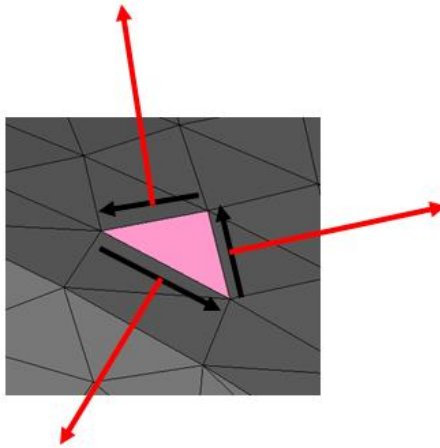
- a) The facet must have at least one free edge or have a neighbor where the angle across the shared edge is greater than the wake definition threshold angle set

by the user (the stable value for this angle is  $90^\circ$  based on experimentations; Figure-7.3.1).

- b) The “centrifugal-normal” for the edges that pass selection criteria (a) must make an angle with the free-stream velocity vector greater than the trailing-edge marker angle (the stable value for this angle is  $75^\circ$  based on experimentations). The centrifugal normal vector for an edge is the vector normal to the edge direction (along the winding of the facet) in the plane of the facet with a direction away from the facet centroid (Figure-7.3.2).

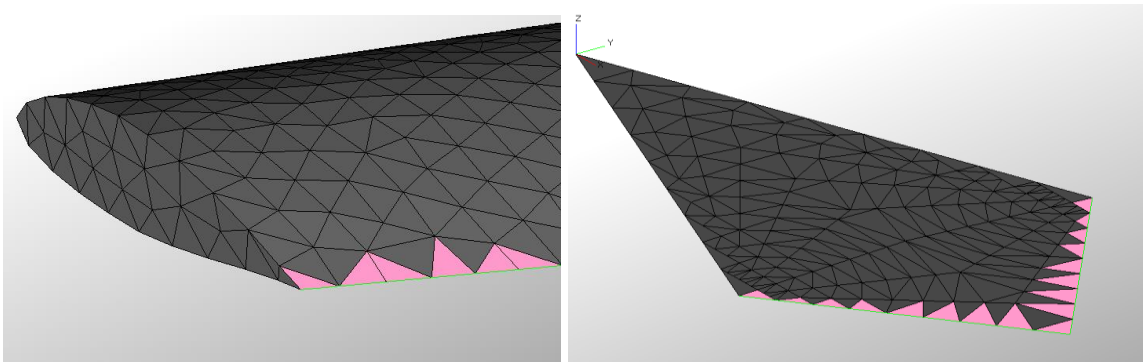


**Figure 7.3.1: Condition-1 for auto-detecting “thick” trailing edges marked by the solver using topological feature angles**



**Figure 7.3.2: Centrifugal normal vectors for a general facet**

Facets that meet the above two conditions are processed for the wake generation criteria including the detection of possible wake nodes along the exposed trailing edge vertices (Figure-7.3.3).

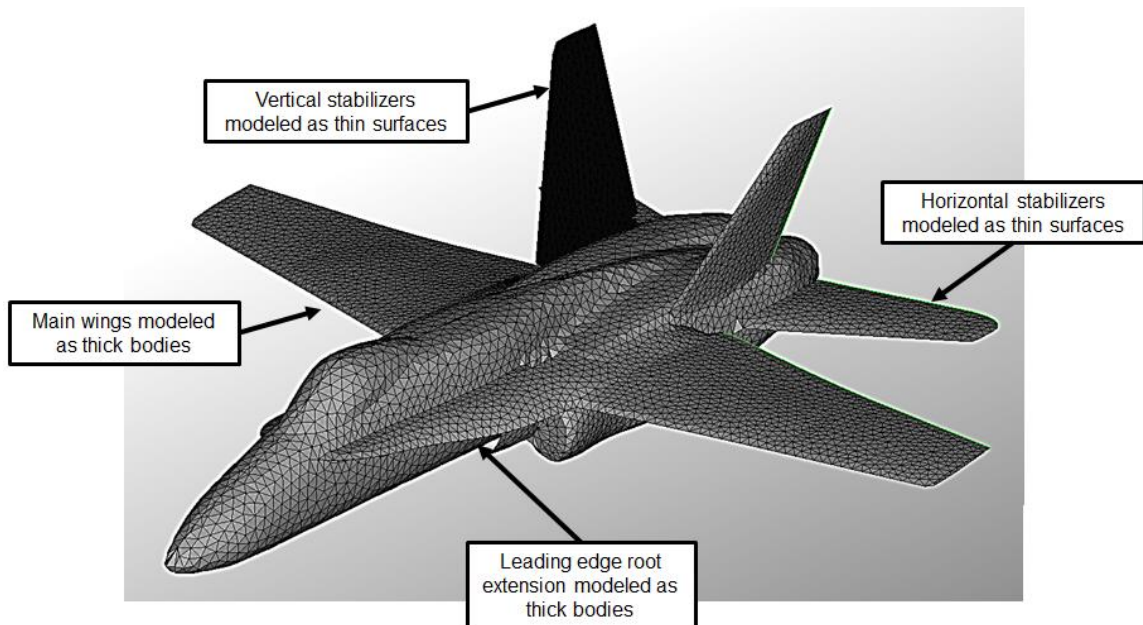


**Figure 7.3.3: Typical results of trailing edge auto-detection. Thick wing example (left) and thin wing example (right)**

It should be noted that the solver is not restricted from marking more than one edge as a trailing edge. It is conceivable that up to two edges of a facet could simultaneously be trailing for a certain orientation and hence will both pass through the

auto-detection routines. The solver takes the existence of this possibility into account and lets the wake node detection routines determine whether a facet can have more than two wake nodes. In this aspect, the current approach attains a significant advantage compared to the pressure-solvers.

The ability to detect thick and thin edges using the two selection criteria allows the solver to maintain robustness with both types of surfaces. This is yet another advantage of the current methodology. This allows the user to improve the efficiency of the solver by marking thick surfaces with very low thickness values as thin and eliminate an entire patch of facets from the solver without affecting the solver results. An example of such an approach is shown in Figure-7.3.4 for the Boeing F-18A Hornet. It has a geometry modeled from a combination of thick and thin surfaces whilst retaining overall geometrical (and hence the solution) fidelity.



**Figure 7.3.4: The F-18 geometry manifold and non-manifold surfaces**

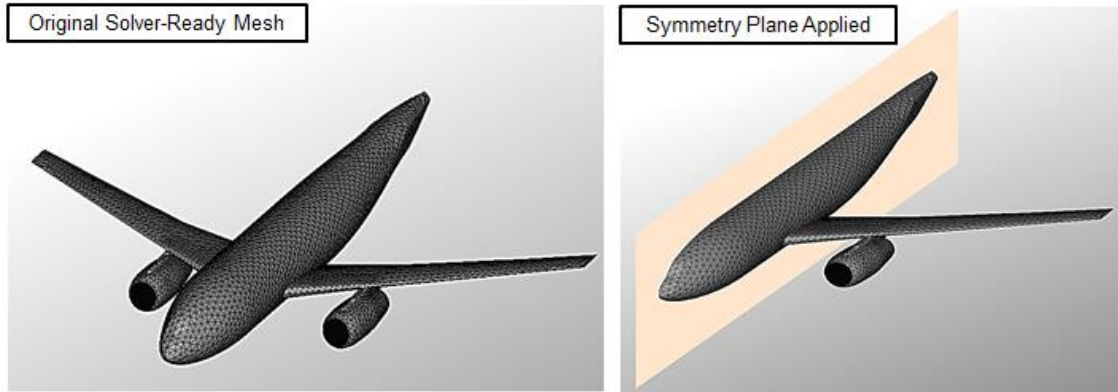
## 7.4 SYMMETRY PLANES

For most standard applications, it is possible to identify a symmetry plane for the geometry with a normal vector perpendicular to the direction of the free-stream flow. This is especially true in those cases with non-morphing and non-rotating geometrical sections. Such cases can take advantage of the symmetry plane boundary condition. The use of these planes significantly reduces the solver workload as well as memory requirements (in order to apply the persistent geometrical induction matrix feature discussed previously) by transferring over half of the mesh (and hence half of the wake) to the common induction matrix evaluated at the beginning of the solver iterations.

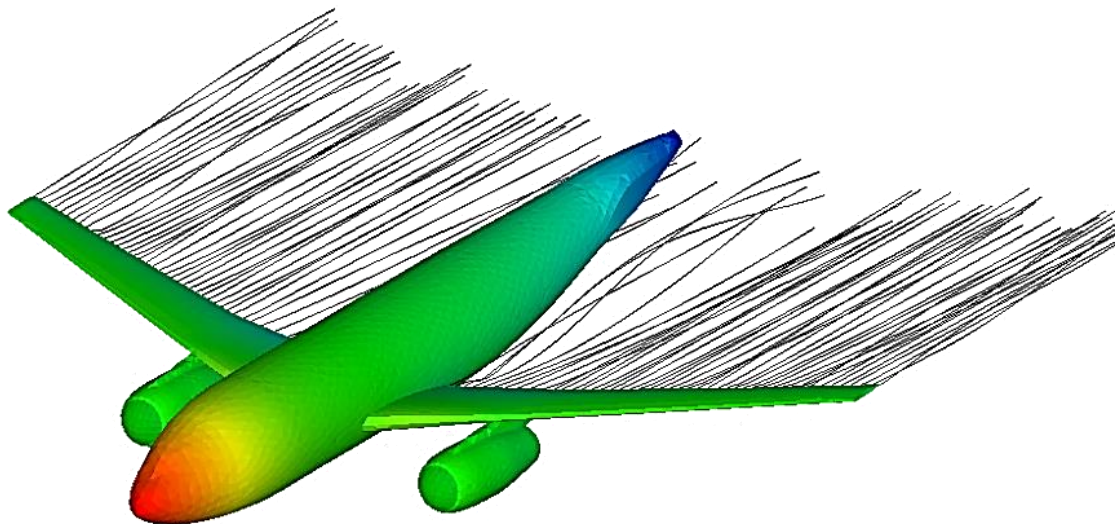
When this condition is applied about the symmetry line, the facets retained are treated as though having a mirror image about the symmetry plane. Note that their winding direction does not change but the storage of vertices is modified accordingly. This allows the facet to induce its own effect spatially but also add to that the induction of its reflection on the other side of the symmetry plane.

An example of the CPU savings from adopting this boundary condition is illustrated for the case of the DLR-F6 wing-fuselage geometry as shown in Figure-7.4.1. The original geometry has 11,038 facets. The effective facet count after symmetry plane has been applied is thus 5,519. Two case runs were done on the solver, with and without symmetry planes for  $0^\circ$  AOA and 1 m/sec free-stream flow. Both geometries were passed through the FDMR routines. The trailing-edges were detected automatically and the geometry was modeled entirely as thick. The resulting wake-field is shown in Figure-

7.4.2 and the results of the two runs are shown in Table 7.4.1 for the individual components that make up the overall solver CPU time.



**Figure 7.4.1: The DLR-F6 geometry with and without symmetry BC**



**Figure 7.4.2: The DLR-F6 geometry with symmetry BC (AOA = 0°, 1 m/sec flow)**

It is easily established that the savings in CPU time for the overall run are substantial (64.46%) for the geometry run with the symmetry condition. Note also the reductions in FDMR time, global geometric induction matrix allocation and storage times

and the solver iterations time for the same number of iterations. There is reduction in all of these components, illustrating the advantage given by the symmetry plane condition.

**Table 7.4.1: DLR-F6 solver results comparison (AOA = 0°, 1 m/sec flow)**

	No Symmetry BC	Symmetry BC
Number of facets	11,038	5,519
FDMR Wall Time (seconds)	2.58	1.02
Induction Matrix Evaluation Wall Time (seconds)	186.51	83.85
Solver iterations Wall Time (seconds)	454.14	143.70
Total Wall Time (seconds)	643.23	228.57



## 8. RESULTS

This Section details the results of the solver runs made for the two main testing suites developed for this effort. The preliminary testing suite deals with simple geometries and shapes designed to test basic features of the solver. These include geometries such as spheres, flat wings etc. These geometries allow easy diagnosis of problem areas and help in identification of the limits of the current approach. They are also very useful to highlight the advances made by the current approach over legacy approaches and also to validate the new approach on well-defined theoretical cases.

The advanced testing suite deals with real world applications and involves testing the solver under stressful mesh conditions (such as large mesh sizes, non-manifold surface intersections and complicated auto-mesher runs), with multi-physics external flows and testing limits of the physics of the potential flow philosophy. The geometries used here are real-world aerospace designs. Results have been collected for the aerodynamic load coefficients, solver run times, convergence histories and mesh statistics (wherever required).

## 8.1 PRELIMINARY TESTING SUITE

The list of geometries tested in this group includes:

- a) Rectangle wings (various aspect ratios)
- b) Delta wings (various aspect ratios)
- c) 75° / 65° Double-delta wing
- d) 70° Arrow wing
- e) 70° Diamond wing
- f) 45° Sweep wing
- g) Unit sphere and other axisymmetric bodies

### 8.1.1 RECTANGLE WINGS

Rectangle wings with aspect ratios of 3.0, 4.0 and 7.0 were designed and an unstructured mesh was generated for each of them. The geometry was designed as thin, free-surfaces. The results are shown in Figure-8.1.1 and Figure-8.1.2. The test results were compared with the original experimental results of Prandtl <sup>23</sup> from 1921 and are plotted alongside for comparison in Figure-8.1.1 and Figure-8.1.2.

The results show generally good adherence to experiments with higher fidelity results for lower angles of attack. This is not unexpected since the new approach simplifies to the standard lifting line approach for rectangle wings in the extreme limiting conditions as disused in Section-4. At higher angles of attack, additional fluid physics such as flow separation begin to take place at higher angles and consequently affect the

lift and drag coefficients. The linear prediction curve then deviates from the experimental results as the experimental lift reduces while the experimental drag increases. Under these conditions the predictions overshoot the measured results.

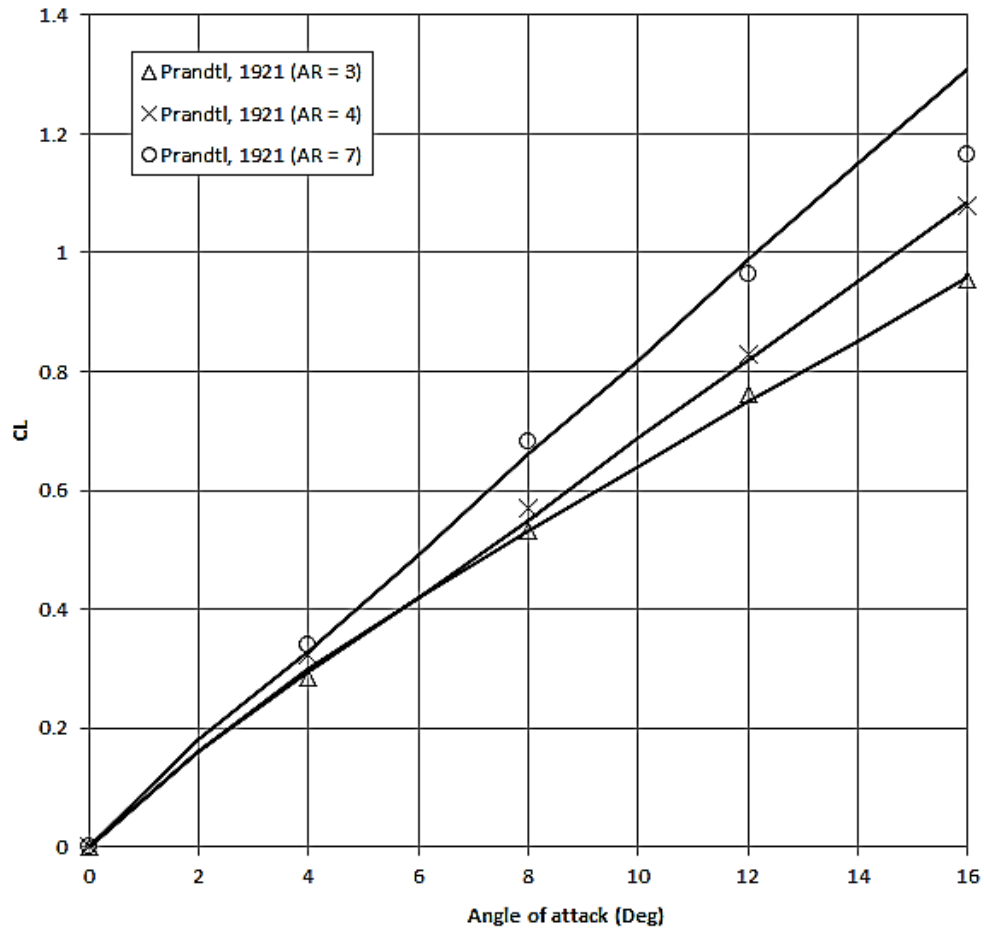


Figure 8.1.1: Lift coefficient versus angle of attack for rectangle wings<sup>23</sup>

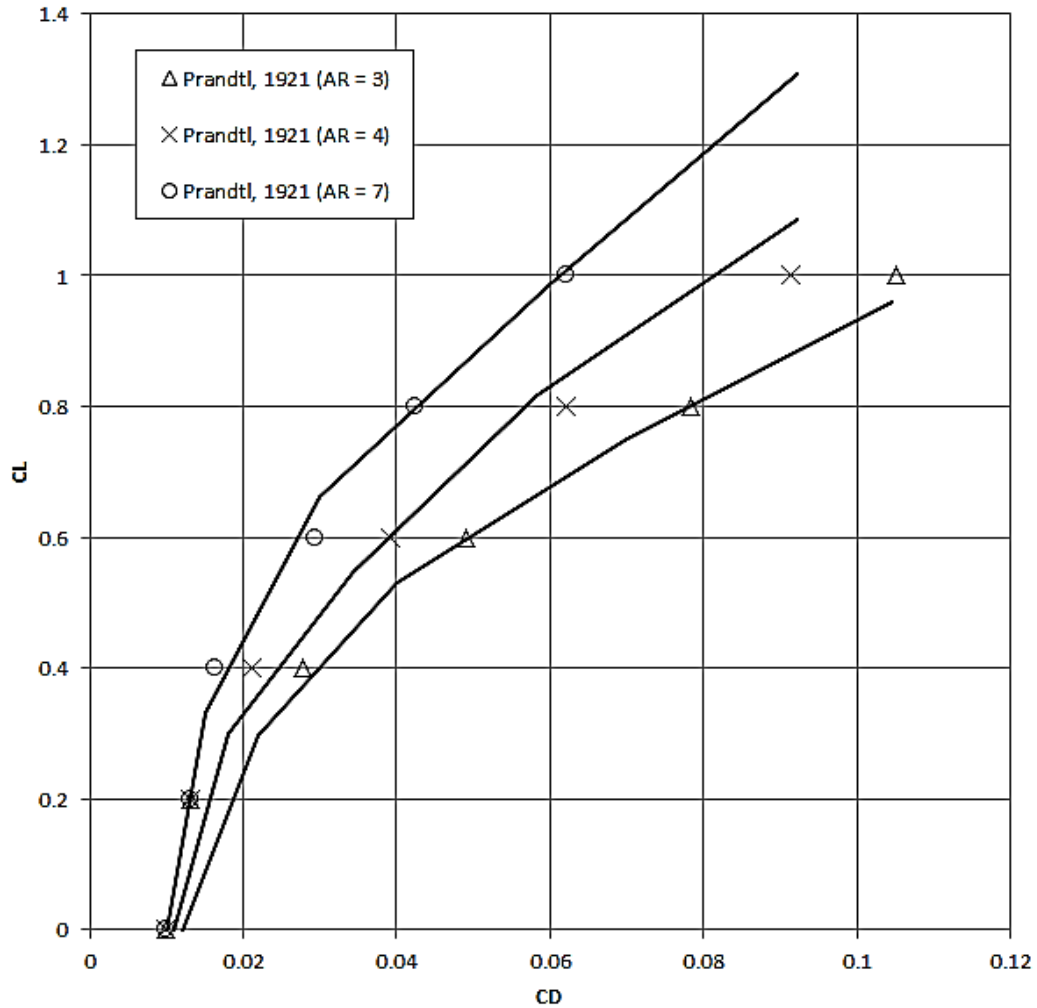


Figure 8.1.2: Lift versus Drag for rectangle wings<sup>23</sup>

### 8.1.2 DELTA WINGS

Unlike rectangle wings, delta shaped wings offer severe challenges for a vorticity-solver given the complicated physics even at modest angles of attack. Unless special care is taken to model the leading edge separation of vorticity as an additional relaxed wake proposed by Katz and Plotkin<sup>14</sup>, or more advanced vortical sheet models are used such as

that done by Burkhalter and Purvis<sup>24</sup> for thin wing surfaces, the standard vorticity-solver is unable to capture the effects for delta wings except at low angles of attack<sup>2</sup>. This approach enhances the application of such solvers to unstructured meshes and has devised load evaluation and wake generation techniques unique to such meshes, but otherwise remains true to the original philosophical concept. As a result, it is theoretically unexpected that the evaluation of loads over pure delta wings would be better.

This is outlined in Figure-8.13 for delta-wings with various aspect ratios. The experimental data was evaluated for a Reynolds number of  $7 \times 10^5$  by Schlichting and Truckenbrodt<sup>25</sup> in 1969. The experimental wings were modeled for thickness  $t=0.12c$ .

It is easily seen that the experimental data is non-linear as a result of the physics involved. The current approach however can only match the linearized physics, and hence fits a linear lift-curve slope through the required data. The linearization deviates from experimental results at higher angles of attack and the error between prediction and experiments is significant within that region.

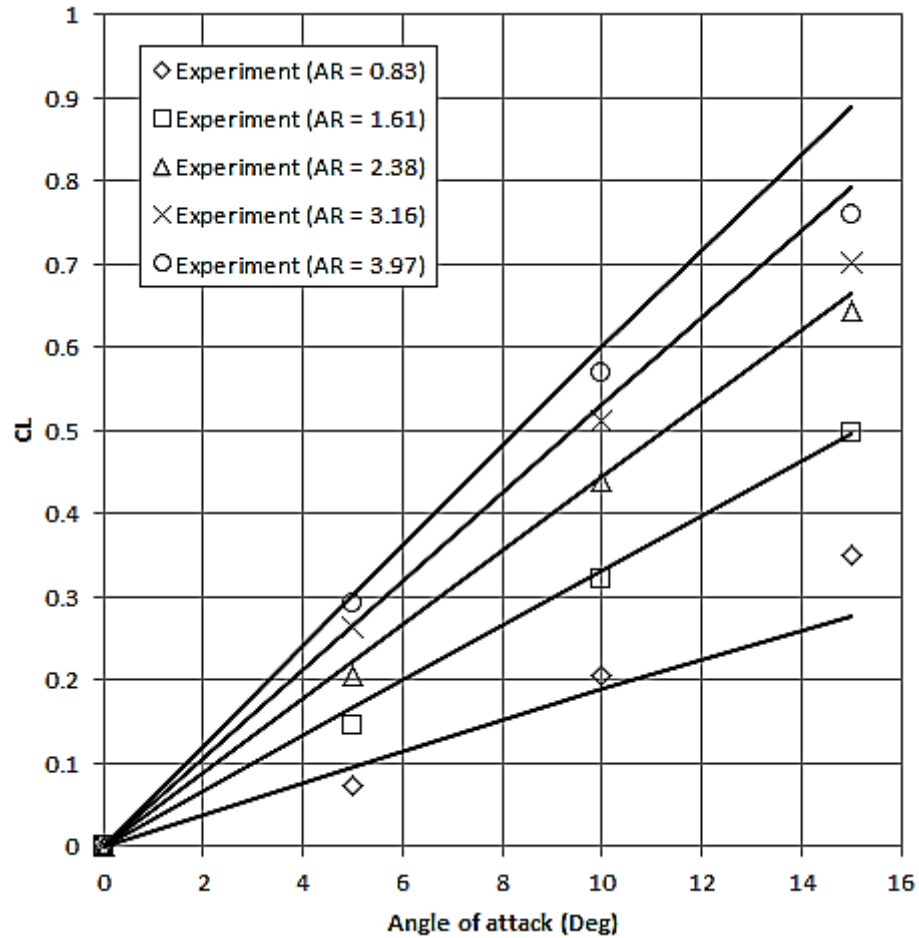
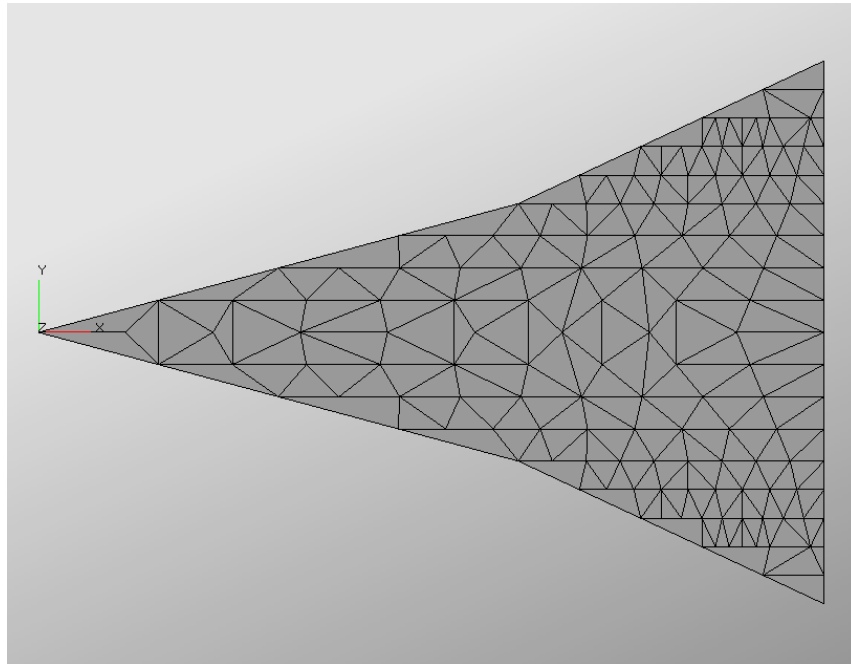


Figure 8.1.3: Lift coefficient versus angle of attack for delta wings <sup>25</sup>

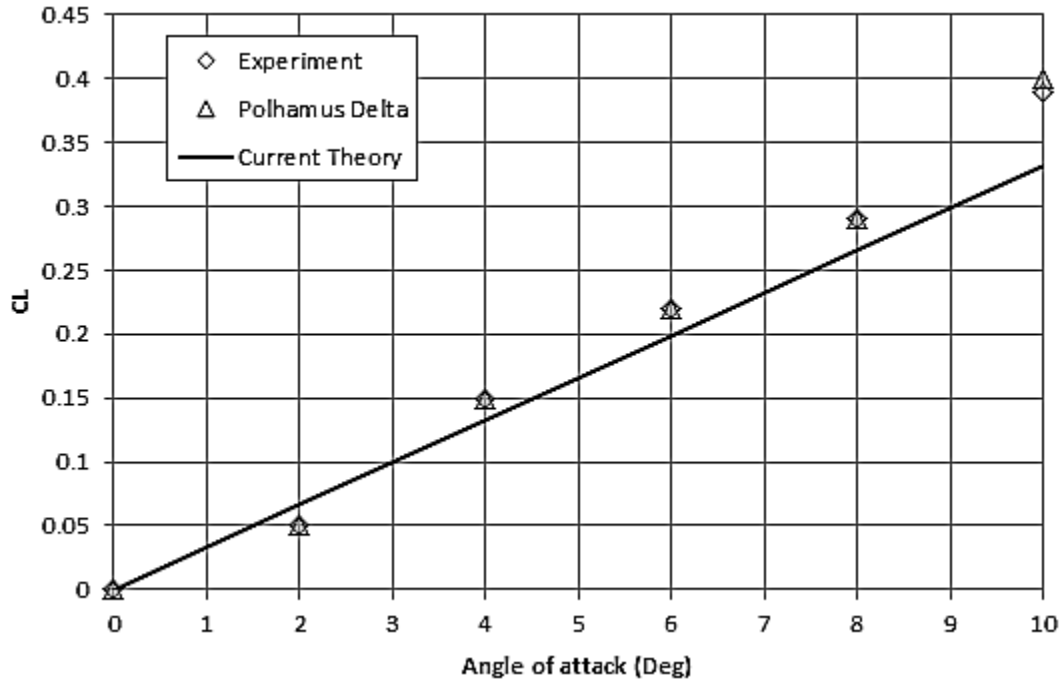
### 8.1.3 DOUBLE DELTA WING 75° / 65°

In addition to the standard delta wings described in Section-8.1.2, several more wing shapes modified from the standard delta were also tested. One such model was the 75° / 65° double-delta wing. The plan-form of this geometry is shown in Figure-8.1.4 post auto-mesher. The nature of the unstructured mesh is also visible in this figure.

The results for this geometry were tested against the experimental results of Wentz and Kohlman<sup>26, 29</sup> from 1968. Additionally, the predicted data was also compared with the empirical equations based on the leading-edge suction analogy developed by Polhamus<sup>27, 28</sup> for high sweep angle wings. The comparisons are shown in Figure-8.1.5. It is seen that the behavior of the predicted data is similar to that of delta wings in that the current approach attempts to put a linearized prediction curve against non-linear data. The general fitting of the predictions is good for lower angles of attack similar to that for delta wings.



**Figure 8.1.4: Unstructured mesh on the 75°/65° double-delta**



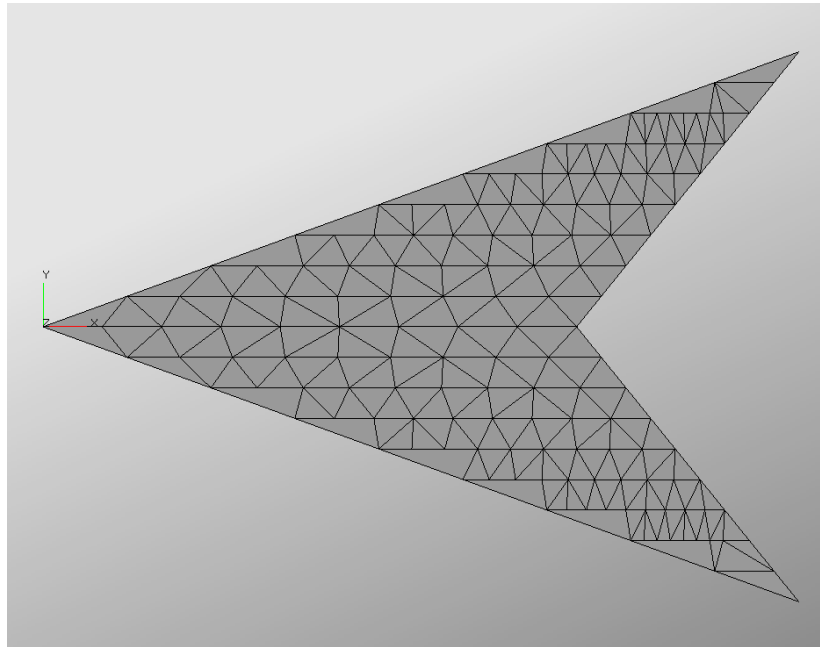
**Figure 8.1.5: Lift coefficient versus angle-of-attack for the 75°/65° double-delta** <sup>26-28</sup>

#### 8.1.4 ARROW WING 70°

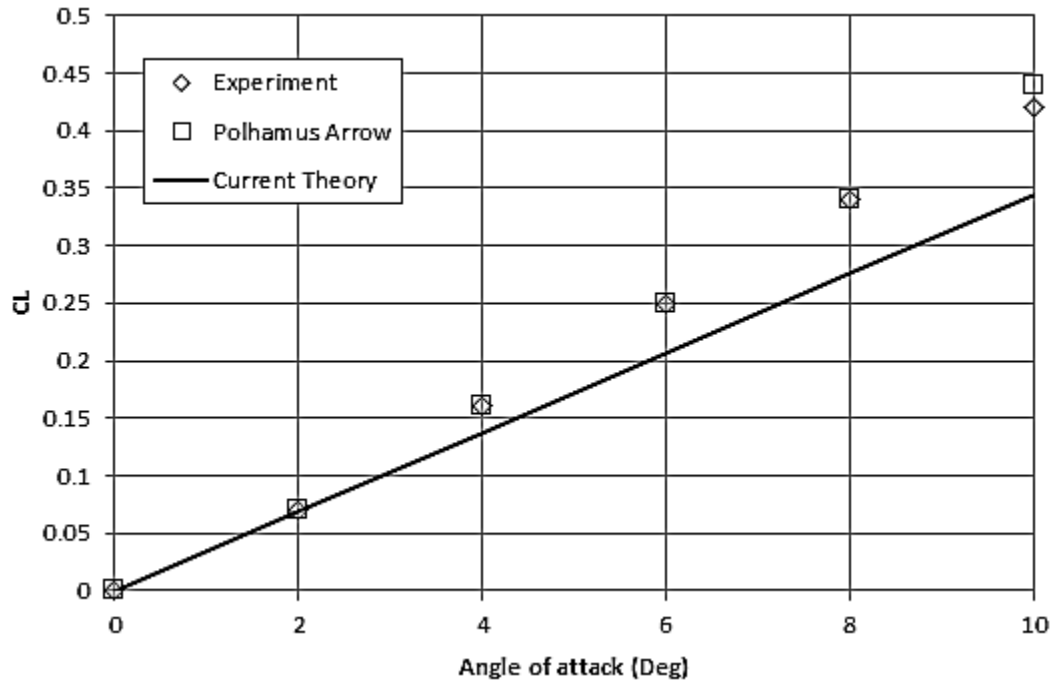
Another geometry modified from the original delta wings is the 70° delta wing as shown in Figure-8.1.6 post auto-mesher along with an overlay of its non-conformal unstructured mesh. The predictions for this wing are compared with the experimental results of Wentz and Kohlman <sup>26</sup> from 1968. Similar to Section-8.1.3, the Polhamus arrow empirical model <sup>27-28</sup> predictions have also been overlaid on the data for comparison. The arrow wing predictions are substantially poor compared with the double-delta and the delta wings even for the limited range of angles as seen in Figure-8.1.6. The predictions are comparable to the experimental results for angles up to 4°. The substantial non-linearity of the arrow wing even at lower angles renders any such



potential flow approach ineffective. Burkhalter and Purvis have managed to get superior results compared to the current approach for such wing geometries using continually distributed vortical sheets.



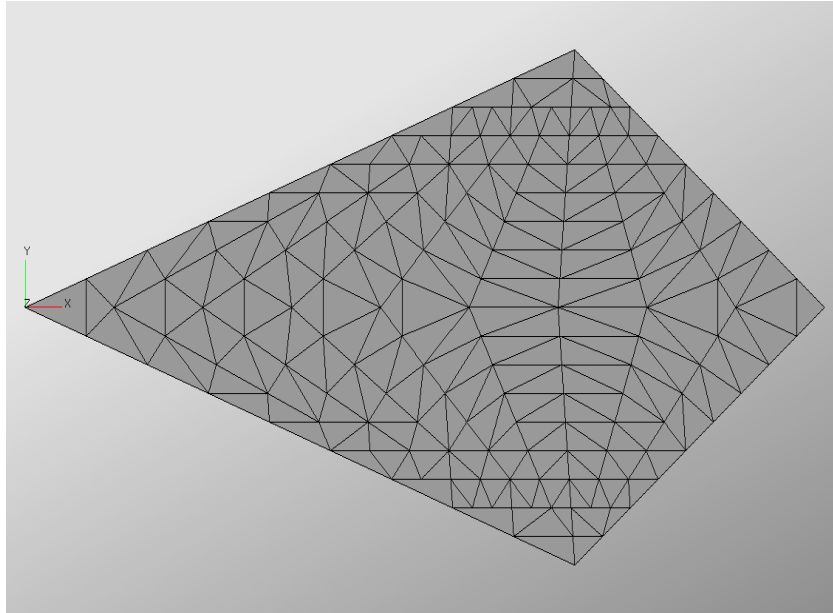
**Figure 8.1.6: Unstructured mesh on the 70° arrow wing**



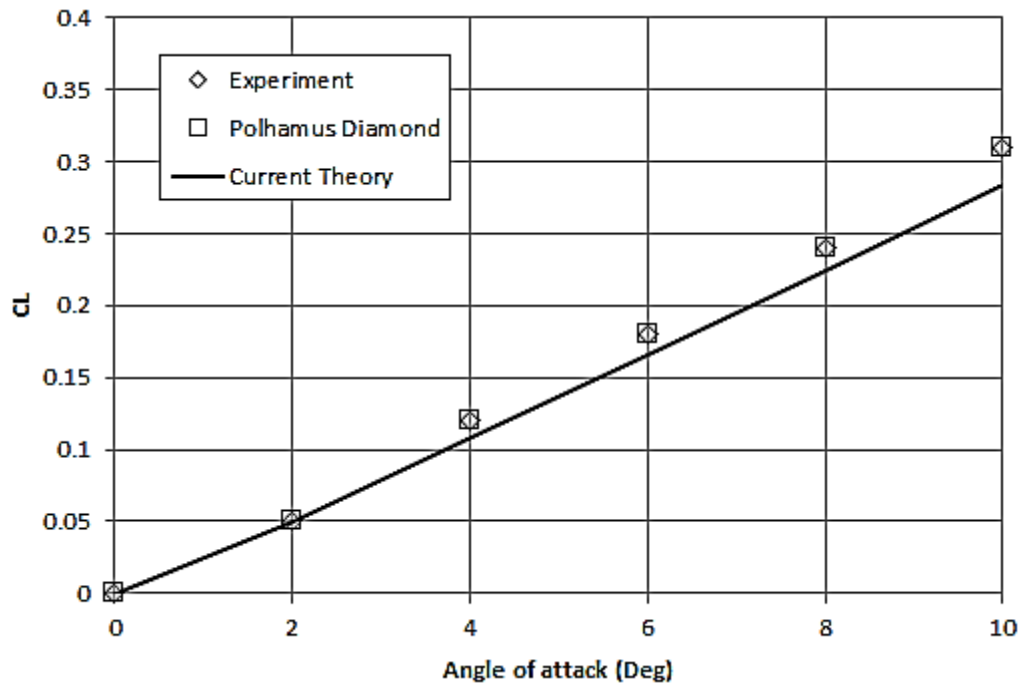
**Figure 8.1.7: Lift coefficient versus angle-of-attack for the 70° arrow** <sup>26-28</sup>

### 8.1.5 DIAMOND WING 70°

The final modification to the delta wings was in the form of a 70° diamond wing as shown in Figure-8.1.8 post auto-mesher. The experimental data used for comparison for this geometry also follows from the work of Wentz and Kohlman <sup>26</sup> from 1968. The Polhamus equations <sup>27-28</sup> modified for a diamond wing are also used for comparison. The results are shown in Figure-8.1.9. The general comparison is good for the range of angles shown with the predictions underestimating the measured data. This is a trend that has now been established as common for all the delta-wing variations seen previously and in contrast to those for the rectangle wings, which always overestimated measured data at higher angles of attack.



**Figure 8.1.8: Unstructured mesh on the 70° diamond wing**

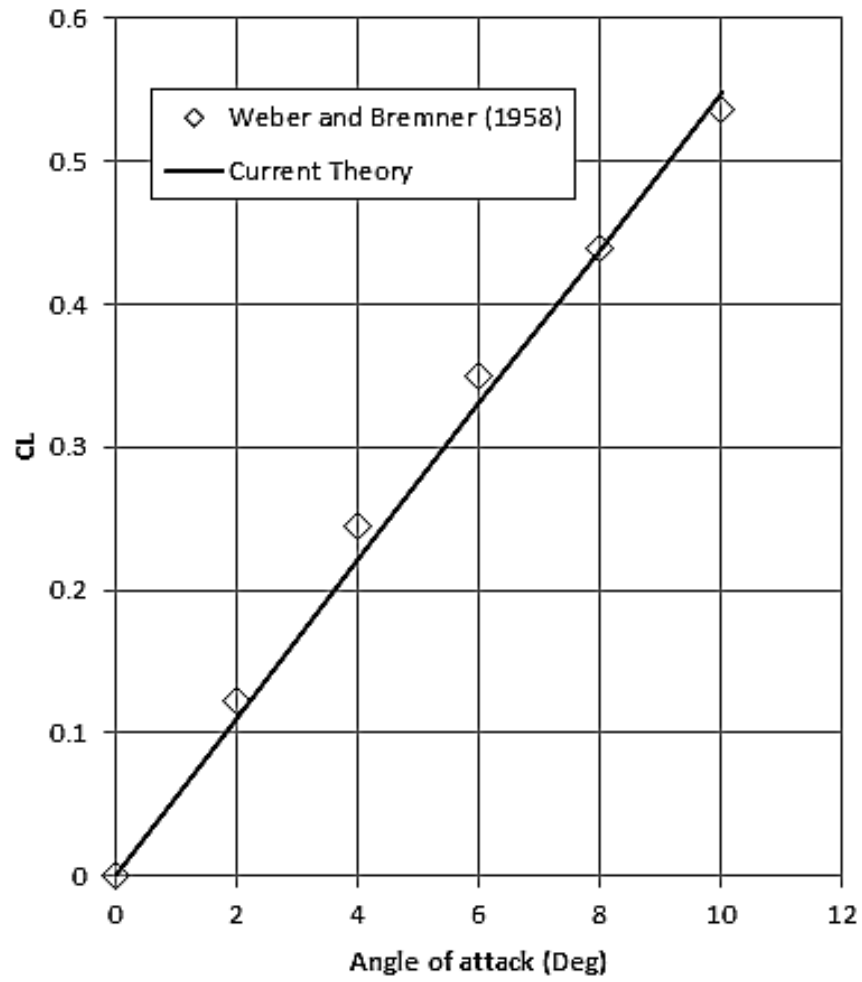


**Figure 8.1.9: Lift coefficient versus angle-of-attack for the 70° diamond wing**

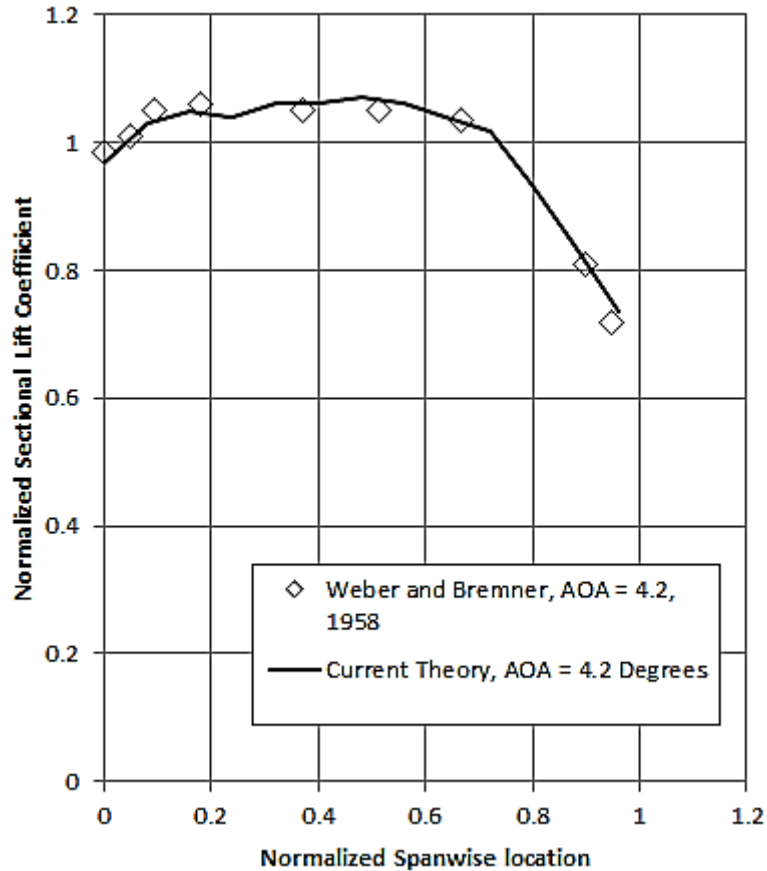
### 8.1.6 SWEEP WING 45°

In contrast to high taper-ratio wings such as the delta, diamond and arrow wings, the current approach predicts very good results for low taper-ratio geometries and does so for other geometries that have physics that are captured by vorticity-solvers. This was seen in the limiting case for a rectangle wing but is not limited to such basic geometries. The case presented here for a 45° sweep wing is another example where the current approach predicts the aerodynamic performance in close accordance with measured data.

The geometry used here is a standard zero taper-ratio rectangle wing with an aspect ratio of 5.0 and 45° sweep on the leading and trailing edges. The experimental data used for comparison here is obtained from the work of Weber and Bremner<sup>30</sup> in 1958 and is shown in Figure-8.1.10 and Figure-8.1.11. Figure-8.1.10 shows the very close approximation of the predictions alongside the measured lift coefficient for the wing for the entire range of measured angles of attack. Similarly, the sectional lift coefficient plotted as a function of span-wise location also shows remarkable estimations alongside the measured data. Note that the sectional lift distribution is nonlinear for such wings<sup>30,31</sup> but the current approach is able to match that nonlinearity accurately. Some noise is seen inside the experimental data as well as the characteristic numerical noise in the sectional circulation data of the current approach (which manifests itself as the sectional lift coefficient using the Kutta-Joukowski formulation).



**Figure 8.1.10: Lift coefficient versus angle-of-attack for the 45° sweep wing**

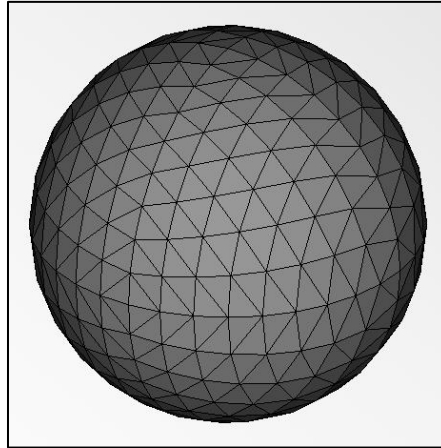


**Figure 8.1.11: Normalized sectional lift coefficient for the 45° sweep wing**

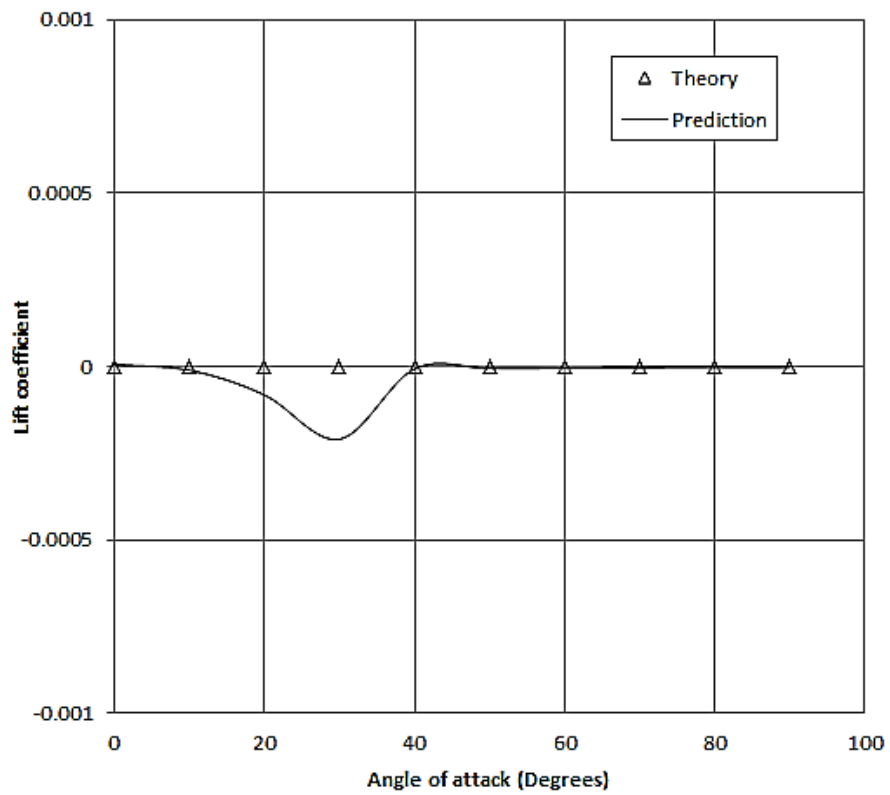
### 8.1.7 UNIT SPHERE AND AXISYMMETRIC BODIES

In addition to the standard wing geometries, axisymmetric bodies were also tested for the classical potential flow results using the current solver. Two of these bodies are presented here: a sphere and an ogive body. The sphere was designed with a unit radius and meshed unstructured as shown in Figure-8.1.12. The theoretical lift coefficient from such a body without wake description is zero. The geometry was tested from 0° to 90° angle of attack and the results are shown in Figure 8.1.13. The overall prediction is found to be well within acceptable limits around the theoretical prediction. There is some

numerical error obtained from both the solver as well as the facetization of the analytical sphere surface and its effects are random as seen in Figure-8.1.13.

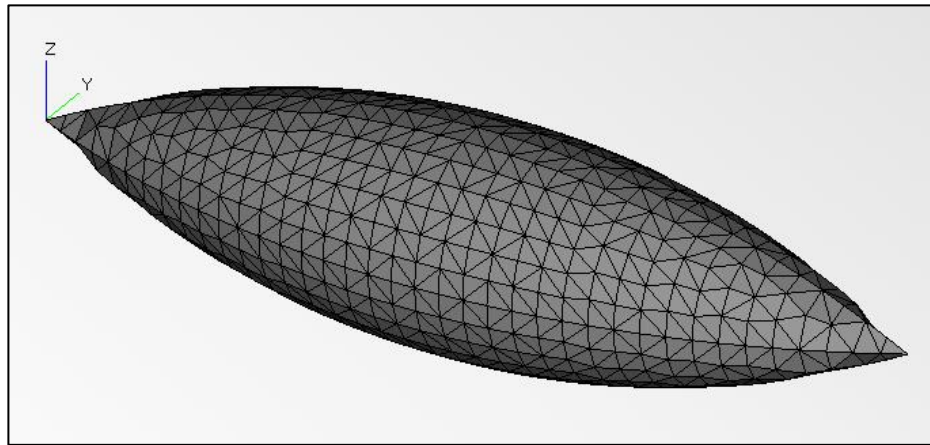


**Figure 8.1.12: The unit sphere**



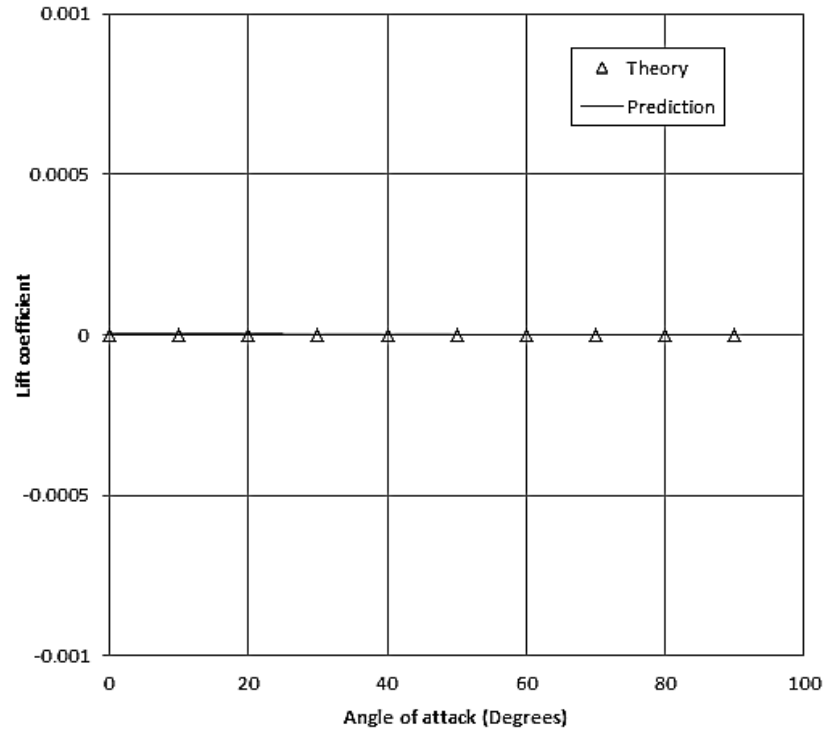
**Figure 8.1.13: Lift coefficient versus angle of attack for the sphere**

The ogive body used for the tests is shown in Figure-8.1.14. This body was generated with a center body radius of 0.5 meters and an overall length of 4.0 meters. The unstructured mesh generated on this body shows some of the limitations of the unstructured mesh discussed in previous Sections. The nose of the ogive is poorly discretized by the mesher but has very good quality facets. This is helpful to the solver but poor for overall mesh fidelity. Using an structured mesh for such a body would have increased the mesh fidelity by retaining the appropriate feature curves but would have resulted in solver oscillations and higher numerical errors the loads. As such, the predicted results for the lift coefficient are much better than the sphere as seen in Figure-8.1.15. The numerical error is significantly lower and overall drives the point that the current approach requires only the “essence” of the underlying geometry for it to provide accurate results. This will prove very useful in the following Sections that deal with the advanced geometries.



**Figure 8.1.14: The ogive axisymmetric body**





**Figure 8.1.15: Lift coefficient versus angle of attack for the ogive body**

## 8.2 ADVANCED TESTING SUITE

The list of geometries tested in this group includes:

- a) General-Dynamics F-16A
- b) McDonnell Douglas F-18A
- c) DLR-F6 commercial transport aircraft
- d) NASA Canard Fighter
- e) Rutan Varietee

### 8.2.1 GENERAL-DYNAMICS F-16A

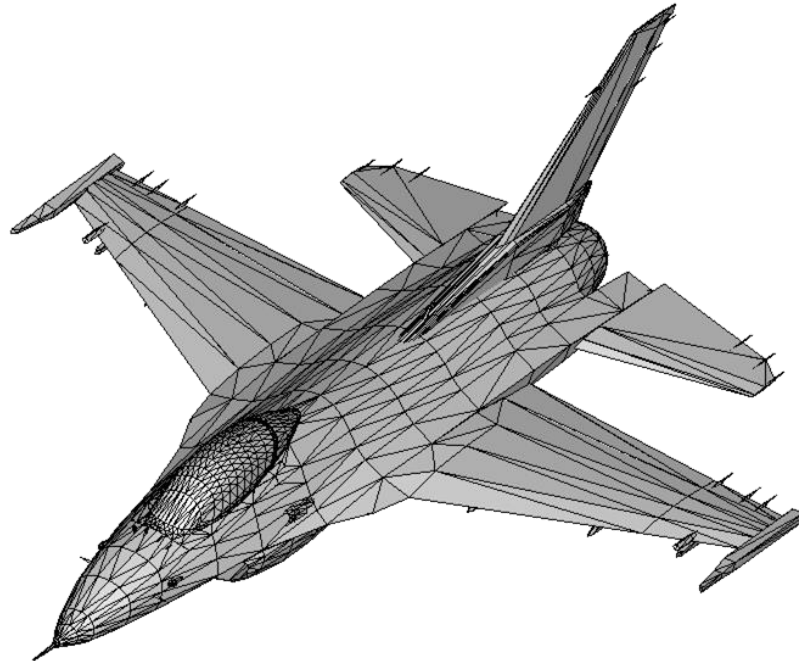
The General-Dynamics F-16A Fighting Falcon was originally designed as a multipurpose strike aircraft in the lightweight fighter category in the 1970s<sup>32</sup>. Its original design objective was superior subsonic-cruise lift-to-drag ratio to allow effective mission radii with a variety of air-to-ground and air-to-air weapons. The design (Figure-8.2.1) has leading edge extensions for the main wing to provide controlled vortex lift and increase lift at high angles of attack<sup>32</sup>. The single vertical stabilizer was chosen over the two surface options unlike the F-15 to reduce buffeting from strake vortices at high AOA. The engine intake is placed below the nose to avoid gun gas ingestion<sup>32</sup>. The overall wing-body intersection is blended for superior aerodynamic performance and increased internal volume for fuel (and hence range).



**Figure 8.2.1: The Lockheed-Martin F-16<sup>32</sup>**

This design was tested using the current approach by the conversion of a flight-simulation geometry used by NASA into a solver ready mesh. The original geometry is

shown in Figure-8.2.2. This represented the starting point in the conversion process for the solver.



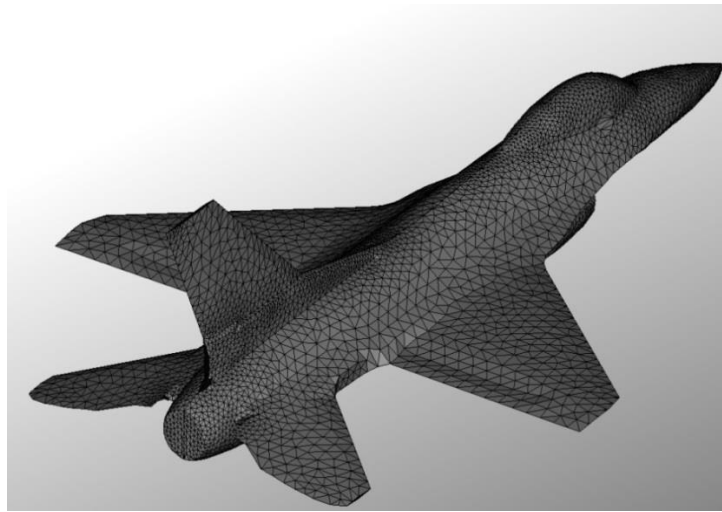
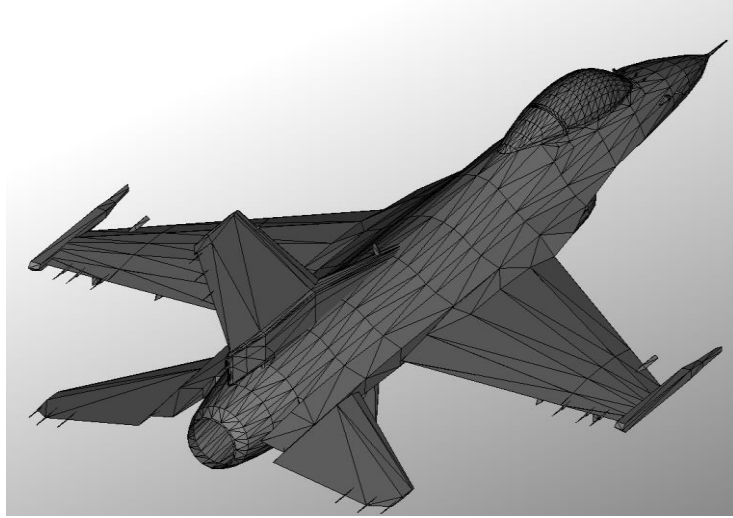
**Figure 8.2.2: Original geometry for the F-16A post CAD design**

This geometry was then cleaned up using manual repair tools to remove geometrical features that would not contribute to the solver fidelity but would otherwise increase the overall mesh size and induce solver instability and increase overall computation time. The features removed in the manual repair process included the weapons stores, the under-wing pylons, the gun ports and the electronics antennae. The engine intake and exhaust were closed off with additional facets and the internal geometry removed. Further, the thin nature of the horizontal stabilizer allowed for the collapsing of the upper and lower surfaces into a mean camber line thin surface.

Following the geometrical de-featuring, the model was passed through the auto-mesher to improve overall mesh quality and sizing gradients. The need from this can be seen from the auto-mesher results shown in Figure-8.2.3. Additional feature curves were marked manually to ensure the auto-mesher did not de-feature the surface curvature in the wing-body intersection regions as well as the cockpit-fuselage intersection regions (this allowed the preservation of the classic bubble-shaped canopy lines of the F-16). Note from Figure-8.2.3 that the auto-mesher induced surface perturbations at several points on the geometry. These were manually cleaned after the meshing process.

The user of the auto-mesher increased facet count substantially as seen from the statistics in Table-8.2.1. However, the overall mesh quality was much improved. This was considered a good tradeoff between computation time and solution fidelity.

The boundary physics for the geometry included slip walls at every point on the geometry except for the intake and exhaust facets that were marked as velocity inlets and outlets. The trailing edge detection was manual and the facets were marked prior to solver initialization. The geometry was evaluated using a symmetry plane.

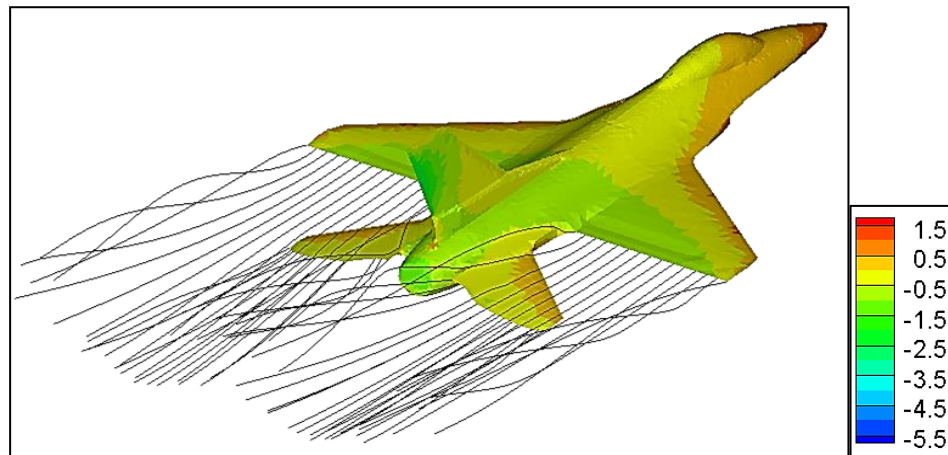


**Figure 8.2.3: Auto-mesher results on the F-16A**

**Table 8.2.1: Auto-mesher results for the F-16A**

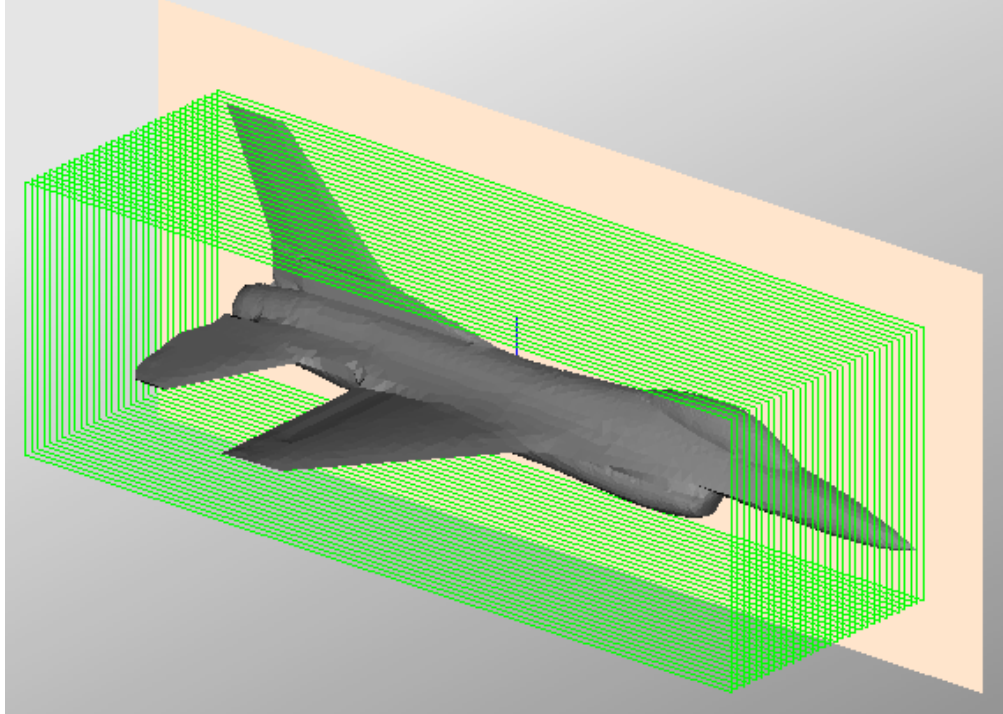
	<b>Initial Geometry</b>	<b>Refined Mesh</b>
Number of facets	4504	13964
Average facet quality	8.3491	2.2253
Worst facet quality	4579.36	106.52

The solver was run for a  $10^{-4}$  convergence setting and the Trefftz plane was marked 6 meters behind the horizontal stabilizer trailing edges. This was considered a safe point for the termination of the wake as a tradeoff between wake fidelity and computation time. The results of the vorticity-solver for the geometry are shown in Figure-8.2.4 for an AOA of  $10^\circ$ . The wrap-around of the main-wing wake strands near the fuselage is easily visible.

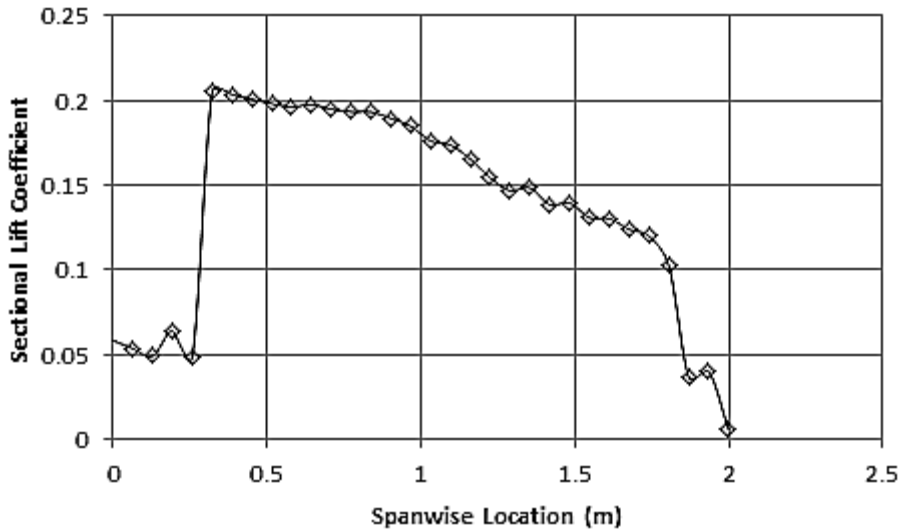


**Figure 8.2.4: Vorticity distribution on the F-16A**

The aerodynamic loads were evaluated using 32 ICL planes (using the average edge-size of the mesh) as shown in Figure-8.2.5. The sectional lift-coefficient as a function of span-wise location for an AOA of  $10^\circ$  is shown in Figure-8.2.6. The loss of lift caused by the presence of the fuselage is easily visible as well as the additive effects of the horizontal stabilizer.



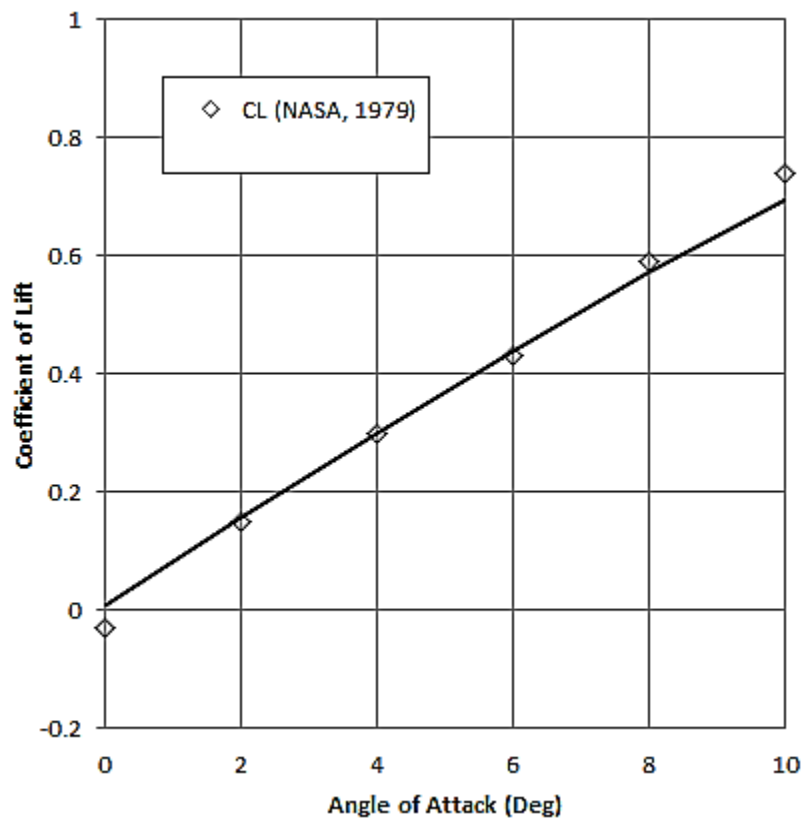
**Figure 8.2.5: ICL distribution about the symmetry plane on the F-16A**



**Figure 8.2.6: Lift coefficient span-wise distribution about the symmetry plane**

The integrated lift data was compared with experimental results obtained by Nguyen et.al <sup>32</sup>, in 1979 as part of a NASA effort. The flight conditions were highly

compressible at Mach 0.9 and the predicted results were corrected for compressibility using the correction terms described in Section-4. The final results are shown in Figure-8.2.7. The overall comparison is found to be very favorable along the measured data. There is some deviation at the lower and upper spectrum of the data as a result of slightly lower slope of the predicted lift-curve. It is reasoned that this is the result of inaccuracies in the source geometry.



**Figure 8.2.7: Lift coefficient versus angle of attack for the F-16A** <sup>32</sup>

The integrated drag predictions were compared with measured data obtained from the work of Webb and Kent <sup>33</sup> in 1977 and were also collected for Mach 0.9 in flight. The predicted data fits well within the flight-test data but the induced drag is seen to be



underestimated at higher angles of attack. The effects of compressibility on the predicted data could also be a factor in this discrepancy.

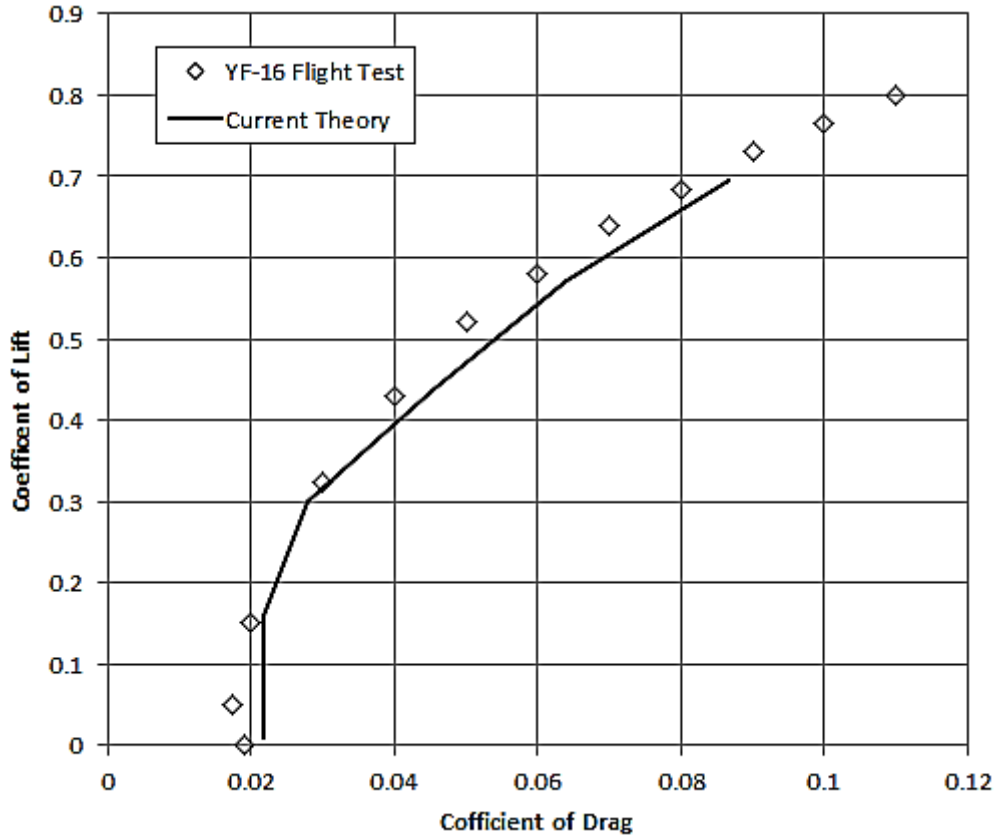


Figure 8.2.8: Lift versus drag for the F-16<sup>33</sup>

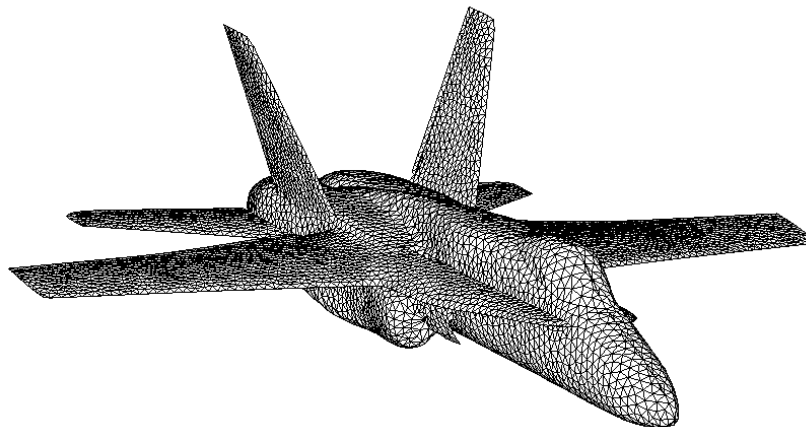
### 8.2.2 MCDONNELL-DOUGLAS F-18A

The second advanced geometry used in this testing suite was the McDonnell-Douglas F-18A (Figure-8.2.9). The pre-solver processing applied on this geometry was very similar to that adopted for the F-16 including the manual de-featuring of extraneous external and internal geometry objects prior to auto-meshing. The final mesh is shown in Figure-8.2.10. The geometry was modeled with 7,857 facets about the symmetry plane.

Given the nature of the experimental results available for comparison, the engine intakes and exhausts were not modeled with a velocity inlet boundary. The horizontal and vertical stabilizers were converted to thin surfaces but the main wing was modeled as a thick body merged with the similarly thick LERXs (visible in Figure-8.2.10).



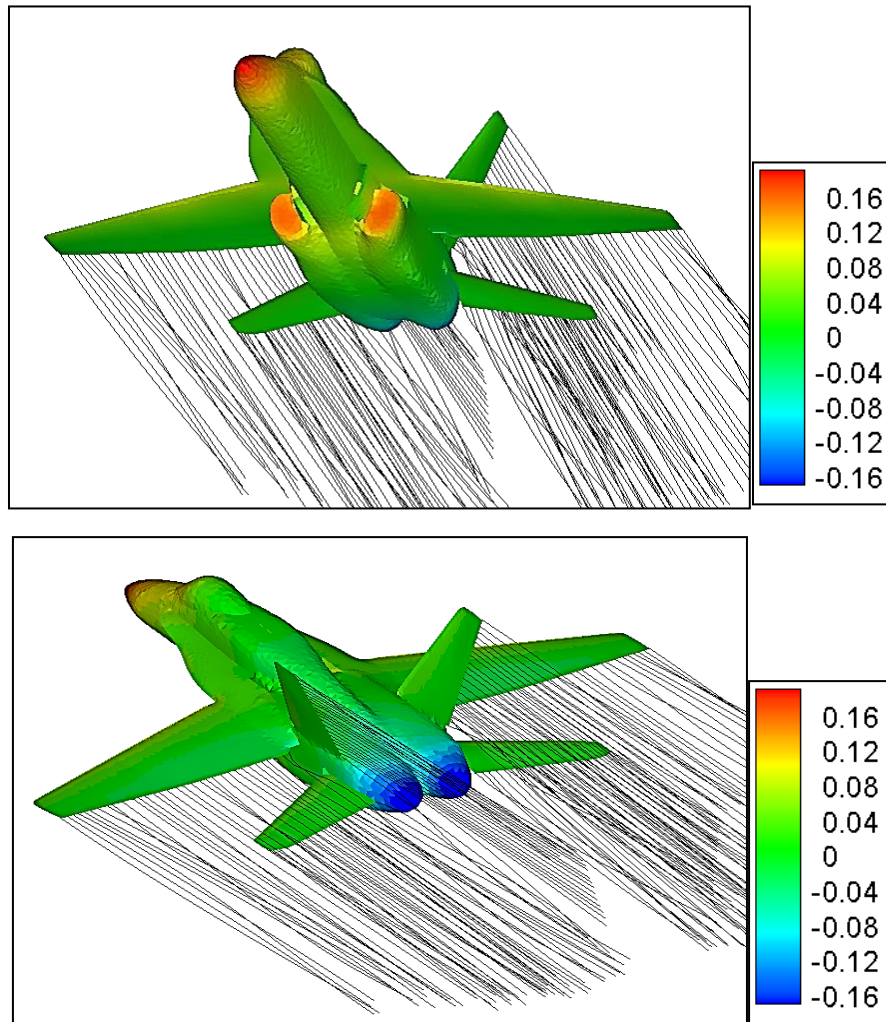
**Figure 8.2.9: The McDonnell Douglas F-18**



**Figure 8.2.10: Auto-mesher results for the F-18A**

The solver was run for a  $10^{-4}$  convergence setting and the Trefftz plane was marked 5.0 meters behind the horizontal stabilizer trailing edges. This was considered a

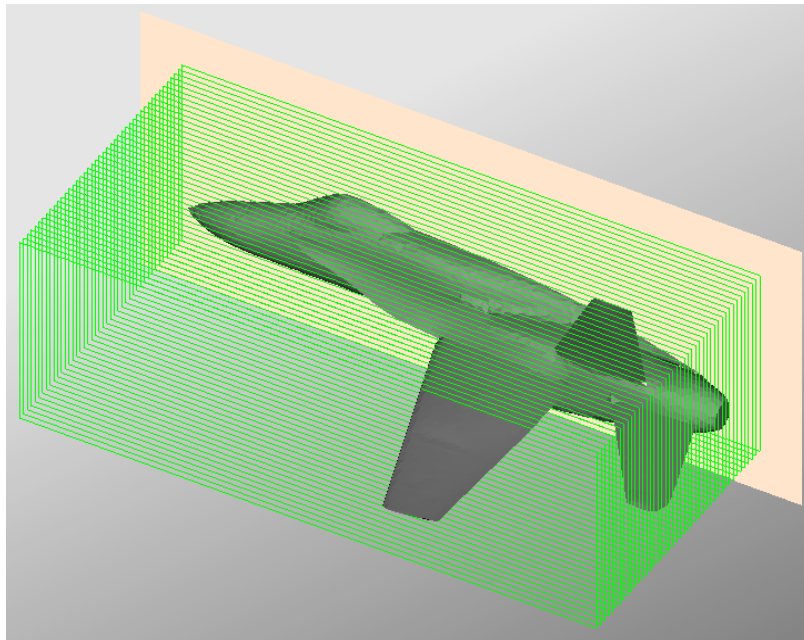
safe point for the termination of the wake as a tradeoff between wake fidelity and computation time. The results of the vorticity-solver for the geometry are shown in Figure-8.2.11 for an AOA of  $1^\circ$ .



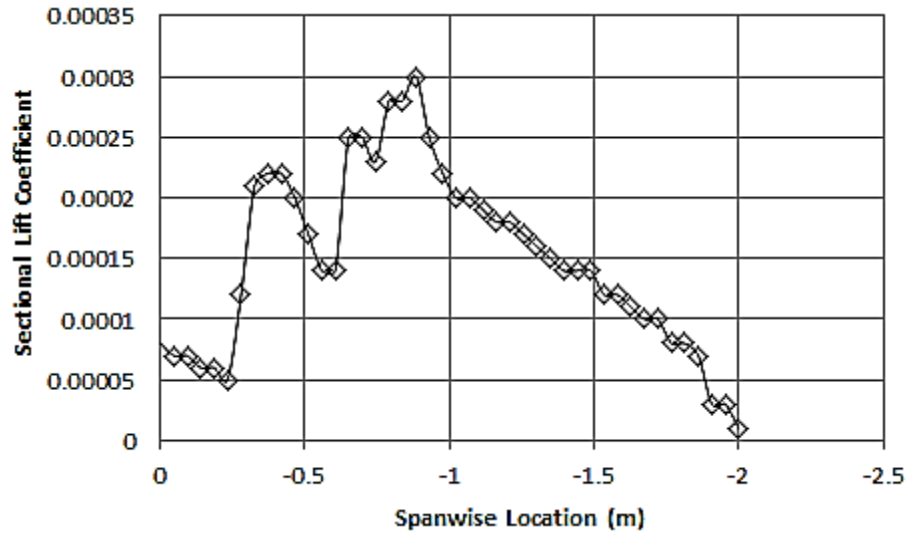
**Figure 8.2.11: Vorticity distribution on the F-18A (vorticity units =  $\text{m}^2/\text{sec}$ ; AOA =  $1^\circ$  and free-stream velocity = 1 m/sec)**

The aerodynamic loads were evaluated using 44 ICL planes (using the average edge-size of the mesh) as shown in Figure-8.2.12. The sectional lift-coefficient as a function of span-wise location for an AOA of  $1^\circ$  is shown in Figure-8.2.13. The figure illustrates the important additive and subtractive effects between the main wing, the

stabilizers and the fuselage. Moving from the wingtip inwards, the additive effect of the horizontal stabilizer is seen initially, spiking the overall sectional lift. This is then subtracted from by the tilted vertical stabilizer which caused the downward spike. The LERXs then add to the lift and the curve spikes up again and finally drop off above the fuselage, which does not produce any useful lift at such low angles.



**Figure 8.2.12: ICL distribution about the symmetry plane on the F-18A**

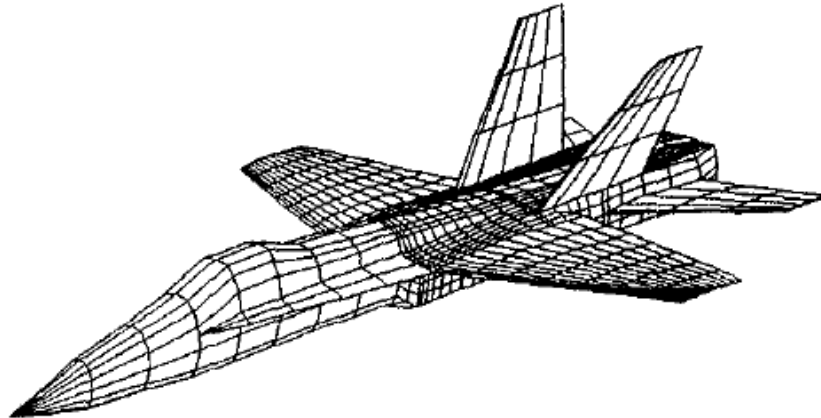


**Figure 8.2.13: Lift coefficient span-wise distribution about the symmetry plane**

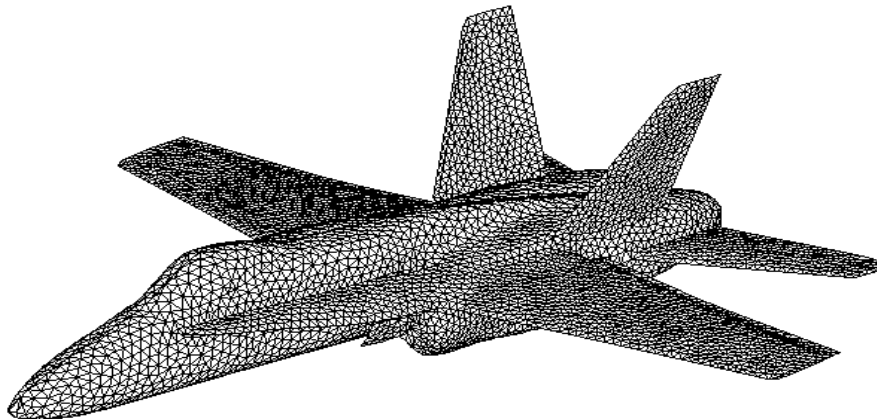
The integrated lift data was compared with experimental results obtained by Banks<sup>34</sup> in 1988. The experimental conditions were slightly compressible at Mach 0.33 and a Reynolds number of  $5.33 \times 10^6$ . The predicted results were corrected for compressibility using the correction terms described in Section-4.

At this point it is perhaps useful to compare the results of the experimental results of Banks with the vorticity solver used by Browne and Katz<sup>35</sup> in 1990 as well as the current solver developed for this effort. The geometry used by Browne and Katz is shown in Figure-8.2.14(a). The fidelity of the geometry is quite poor compared with geometry used for the current solver (Figure-8.2.14(b)). This is understandable given the computational hardware limitations in 1990. However, moving past the geometrical fidelity, several philosophical differences can be outlined between the past work and the current methodology. Primarily, the model developed by Browne and Katz<sup>35</sup> is a structured mesh with quad valency panels. This is in line with the approach taken by

legacy Vorticity solvers. For geometries as advanced as the F-18, this imposes a penalty with the solution fidelity compared with the unstructured mesh used for the current approach.



**Figure 8.2.14(a): Panel model of the McDonnell Douglas F-18A<sup>35</sup> with 1,336 panels (effective 2,372 facets for current solver)**



**Figure 8.2.14(b): Unstructured model of the McDonnell Douglas F-18A with 7,853 facets used in the current solver**

This penalty between the two solver approaches is shown in the comparison of the results next to the NASA Ames wind-tunnel results by Browne and Katz<sup>35</sup> in 1990 (Figure-8.2.15 and Figure-8.2.16). As seen from the lift data, the comparison of both solvers is favorable at lower angles of attack ( $AOA < 12^\circ$ ). However, the vorticity-solver

used by Browne and Katz <sup>35</sup> under-predicts the induced effects while the current approach over-predicts the induced loads beyond AOA=15°. The drag results amplify this discrepancy substantially in favor of the current approach. It is seen from Figure-8.2.16 that the current approach predicts the drag results much better than the older solvers at higher AOA. Even so, beyond AOA=20°, the current approach also begins to under predict the drag as nonlinear physics become dominant on the upper surfaces of the F-18 LERXs.

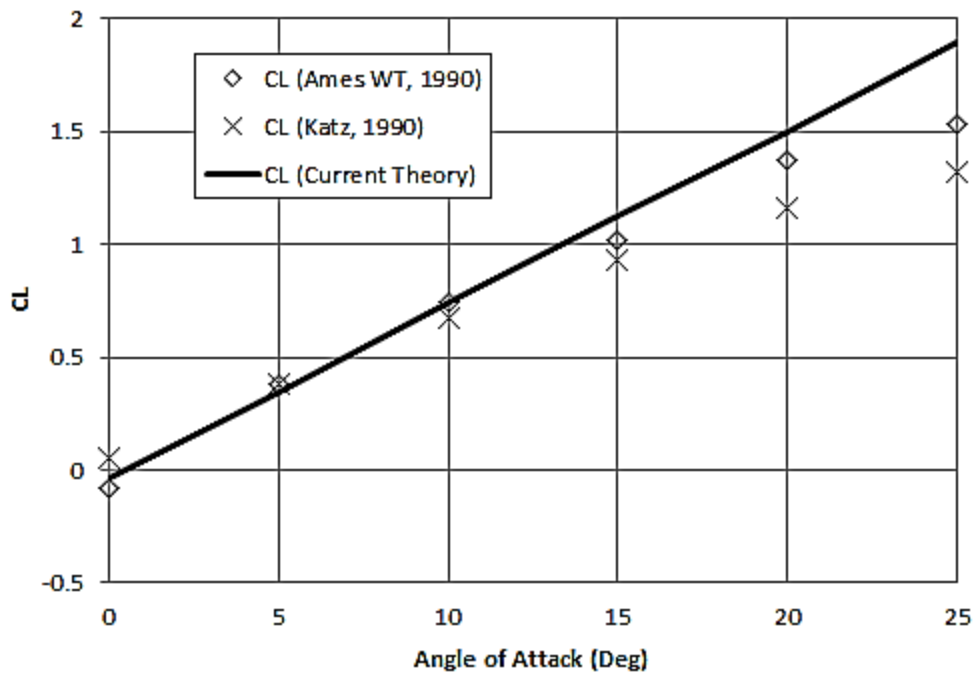


Figure 8.2.15: Lift coefficient versus angle of attack for the F-18A <sup>34-35</sup>

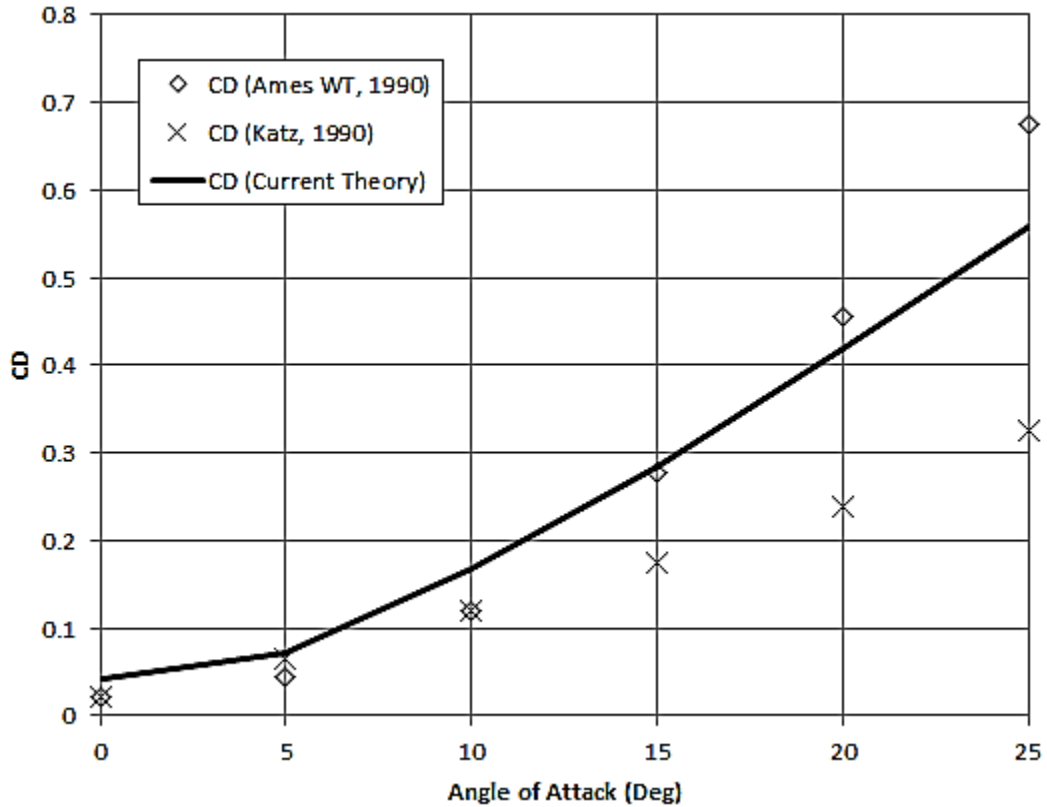
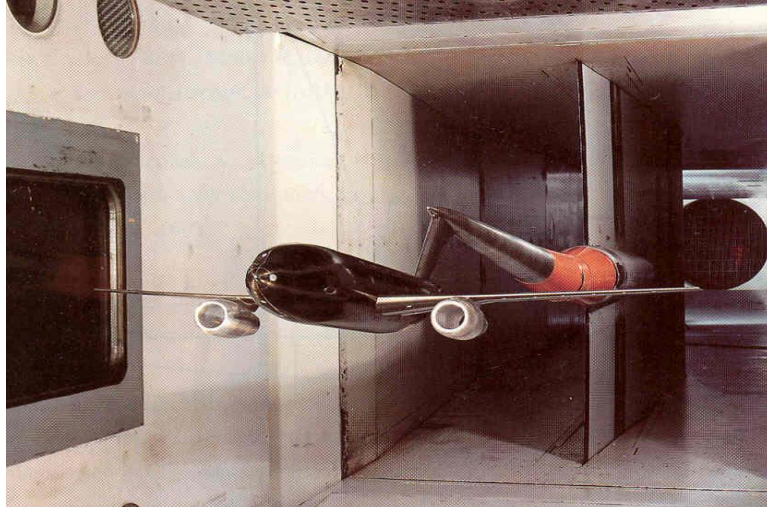


Figure 8.2.16: Lift versus drag for the F-18A <sup>34-35</sup>

### 8.2.3 DLR-F6 COMMERCIAL TRANSPORT AIRCRAFT

The third geometry used in this testing suite was the AIAA DLR-F6 (Figure-8.2.17) <sup>36-38</sup>. The pre-solver processing applied on this geometry differed from the previous cases in that the original CAD tessellations did not have to be de-featured or any components removed other than the volumetric bounding box. This is because the DLR-F6 geometry is part of the AIAA drag prediction workshop <sup>36-38</sup> testing suite and as such is designed in CAD for use in CFD. Unlike the F-16 and F-18 geometries, which were designed as visualization meshes for simulators, this geometry is a converted CFD <sup>36</sup> model used in the current solver.

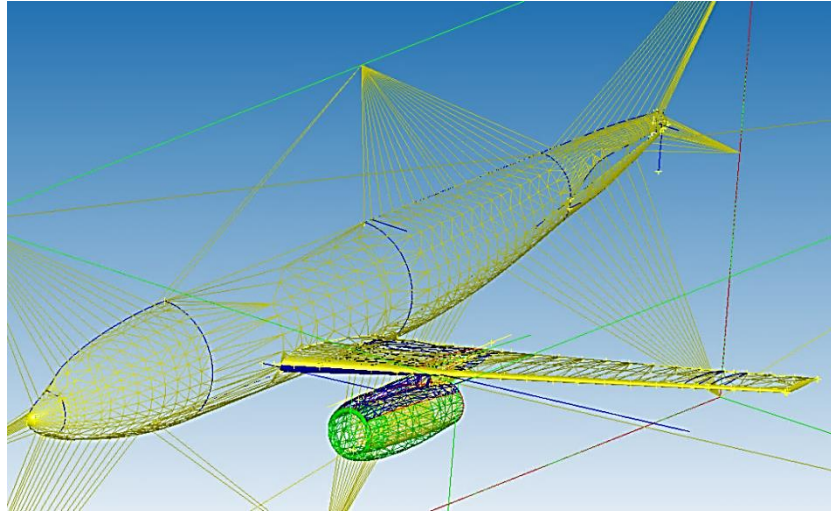




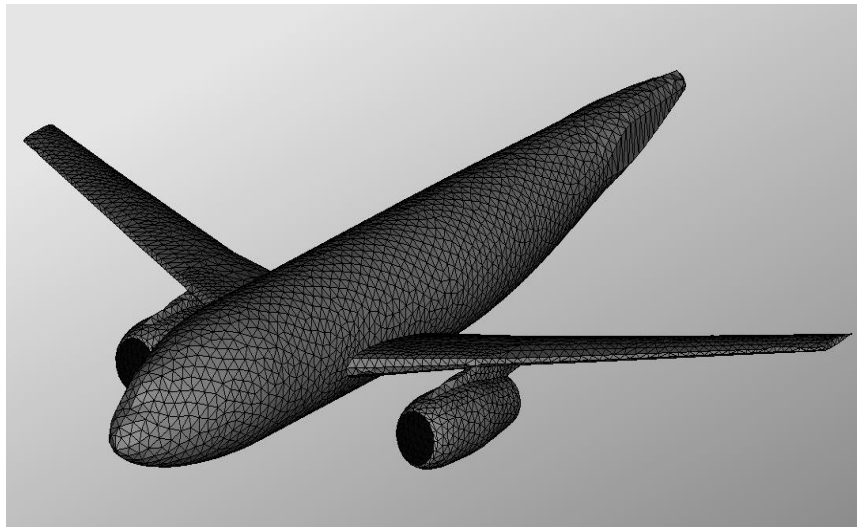
**Figure 8.2.17: The DLR-F6 geometry on a mount in the DLR wind-tunnel <sup>36</sup>**

The initial CAD tessellations are shown in Figure-8.2.18. The geometry is already in an unstructured state and is used as such. The initial mesh was topologically consistent and the bounding box mesh about the symmetry plane was split accordingly <sup>36</sup> as seen in Figure-8.2.18. This was subsequently removed for testing in this approach since volume meshes were not needed. The presence of underlying CAD also allowed for a unique test of the CAD projection abilities of the auto-mesher which resulted in a good quality, solver-ready surface mesh as shown in Figure-8.2.19. This final mesh had 11,038 facets about the symmetry plane.

On the DLR-F6, the podded engine is hollow inside and follows the model design for the wind tunnel <sup>37-38</sup> (Figure-8.2.17). This negated the need for velocity inlets to be applied on the engine intake and exhaust. Further, the geometry is simplified and there are no horizontal and vertical stabilizers on the geometry. No surfaces had to be collapsed to remove their thicknesses. True to the original CFD geometry, the model was tested with a symmetry plane boundary.



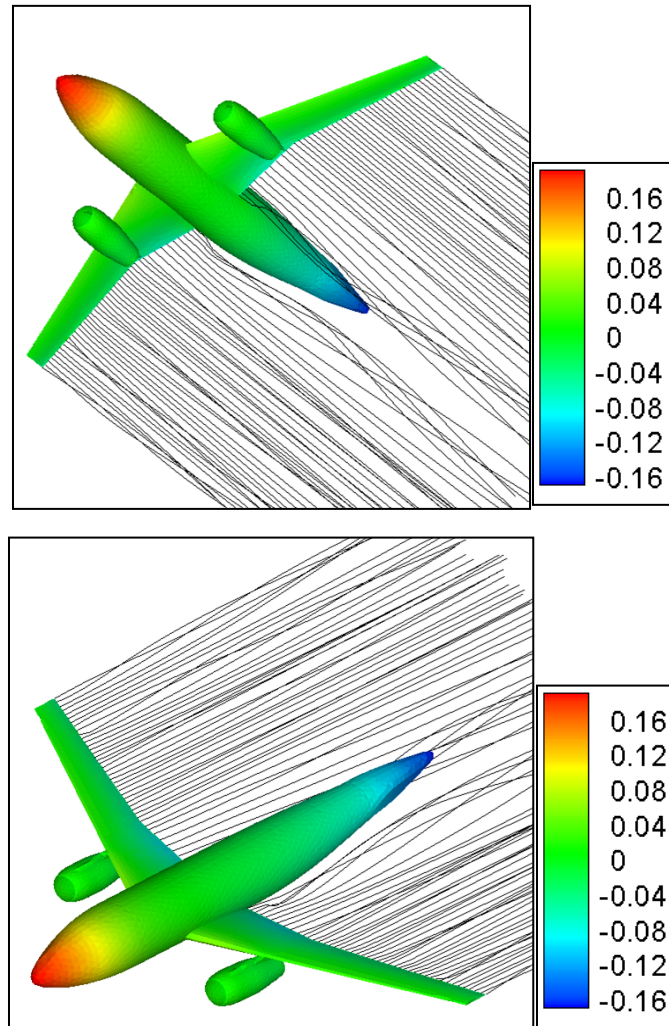
**Figure 8.2.18: Initial AIAA CAD tessellations for the DLR-F6** <sup>36</sup>



**Figure 8.2.19: Auto-mesher results for the DLR-F6**

The solver was run for a  $10^{-4}$  convergence setting and the Trefftz plane was marked 1.0 meters behind the horizontal stabilizer trailing edges (the model overall length is 1.188 meters using the scaling applied on the wind-tunnel geometry). As such the Trefftz location corresponded to roughly one body length past the last facet on the mesh and this was considered a safe point for the termination of the wake as a tradeoff

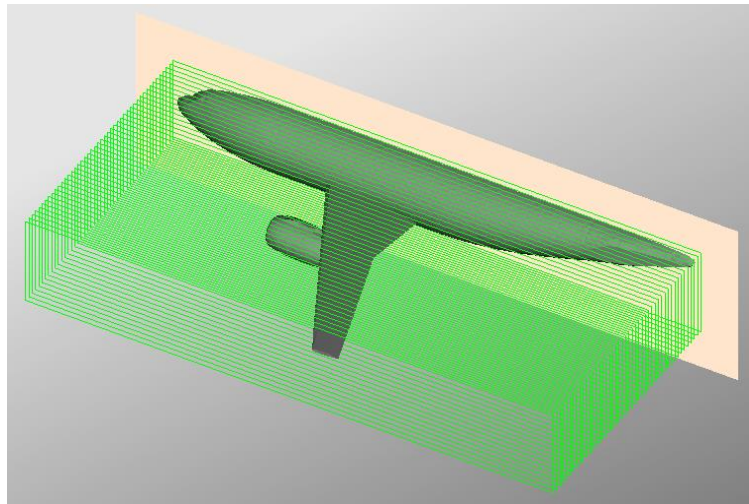
between wake fidelity and computation time. The results of the vorticity-solver for the geometry are shown in Figure-8.2.20 for an AOA of  $1^\circ$ .



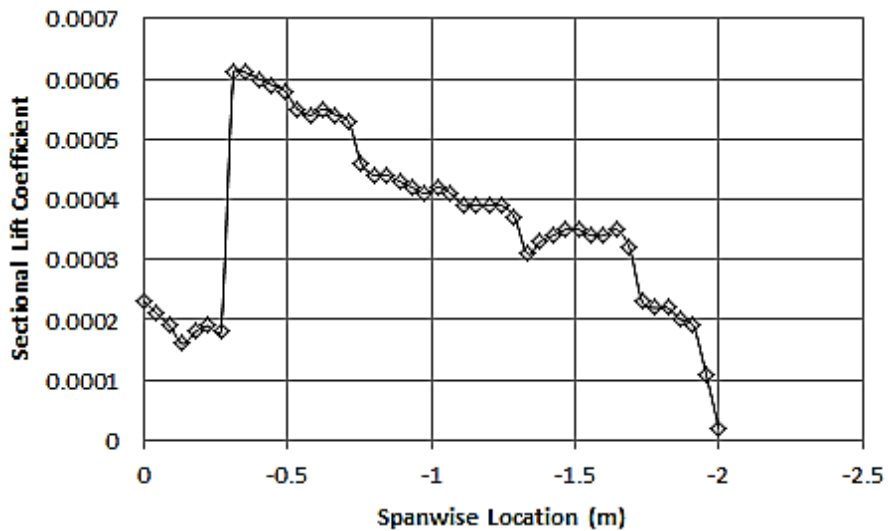
**Figure 8.2.20: Vorticity distribution on the DLR-F6 (vorticity units =  $\text{m}^2/\text{sec}$ ; AOA =  $1^\circ$  and free-stream velocity = 1 m/sec)**

The aerodynamic loads were evaluated using 46 ICL planes (using the average edge-size of the mesh) as shown in Figure-8.2.21. The sectional lift-coefficient as a function of span-wise location for an AOA of  $2^\circ$  is shown in Figure-8.2.22. As for the case of the F-18, the figure illustrates the important additive and subtractive effects

between the main wing and the fuselage. However, unlike the F-18 geometry, the much larger numerical noise is seen in the ICL vorticity distributions as a result of the low-density mesh on the wing, which results in much larger surface perturbations. These can be improved with further local refinements of the mesh but as is outlined later, the effect on the results is expected to be minimal.



**Figure 8.2.21: ICL distribution about the symmetry plane on the DLR-F6**



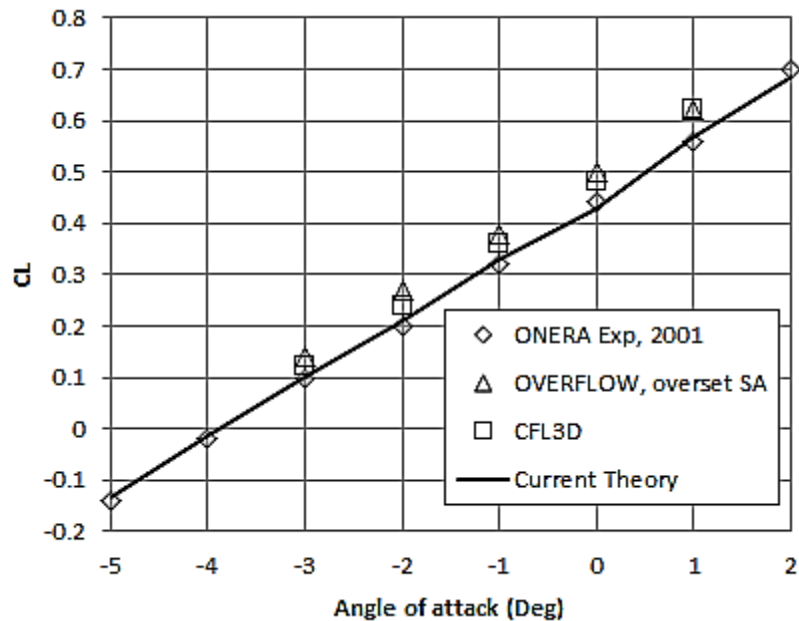
**Figure 8.2.22: Lift coefficient span-wise distribution about the symmetry plane**

The integrated lift and drag data was compared with experimental results obtained in the ONERA <sup>36</sup> tunnels (in 2001) as well as with two Navier-stokes solvers (CFL-3D and OVERFLOW <sup>37-38</sup>) using two “medium” meshes and Overset mesh solvers <sup>36-38</sup>. The choice of using medium meshes as a comparison point with CFD is deliberate. It is well understood that the above solvers working with “fine” meshes result in superior results than the relatively coarser “medium” meshes. However, it was discovered that the tessellations used in the current approach are similar to the “coarse” meshes used in CFD solvers, even though from potential-flow standpoint, they represent very refined meshes. DLR-F6 results with similar coarse CFD meshes could not be obtained <sup>36</sup>. However, medium mesh results are available <sup>36-38</sup> and used here to illustrate the philosophical differences in requirement between the two approaches using the same underlying geometry.

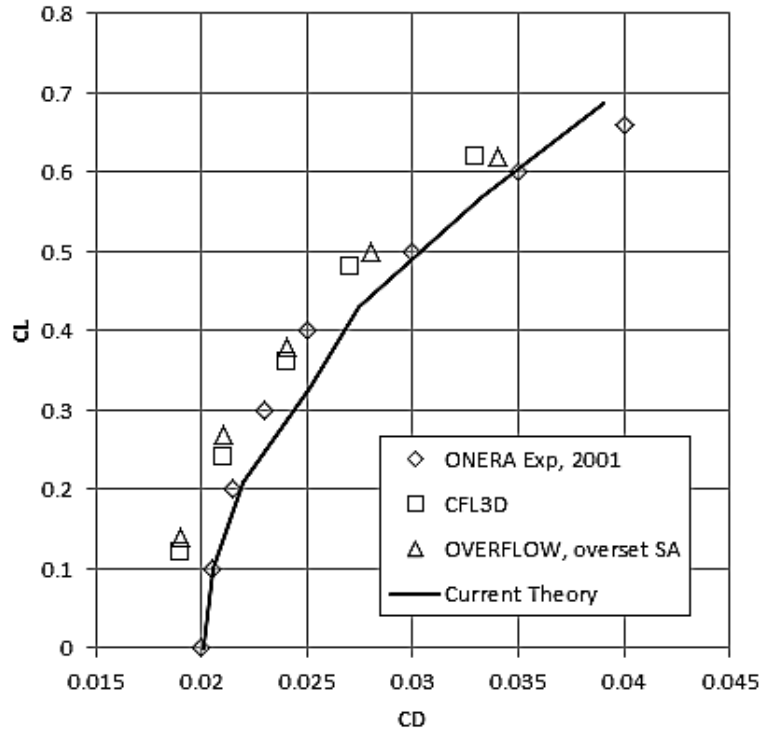
The experimental conditions as well as those used in the CFD solvers were highly compressible at Mach 0.75 <sup>37</sup>. In both these cases, the wing pressure distribution showed supercritical performance and normal shock spikes were measured on the surfaces during experiment <sup>36</sup>. This remains out of bounds of the limit of the potential flow solvers. However, using the precedence from the F-16A tests described previously, the tests on this geometry were continued after including the corrections for compressibility as described in Section-4.

The overall comparison is found to be very favorable alongside the measured data as seen in Figure-8.2.23 and Figure-8.2.24. The lift data shows the Navier-Stokes solvers with medium meshes overestimating the measured data while the current solver is able to

match closer. The perturbations in the evaluated data for the current approach are due to the perturbations noted in the ICL vorticity distributions in Figure-8.2.22 and are entirely numerical errors. The drag data in Figure-8.2.24 shows similar trends as the lift data for the two Navier-Stokes solvers<sup>36-38</sup>. The hypothesis by the authors is that there is an over-prediction in the surface pressures as a result of the coarser mesh. Since the aerodynamic loads are evaluated by pressure-summations, both the lift and drag are seen to be over-predicted. The drag predictions by the current approach shows a little discrepancy once again brought out by the vorticity perturbations in the ICL (drag data between 0.025 and 0.03 is perturbed adversely). Given the outer bounds of the physics involved, the current approach manages to get relatively close matches with the measured data.



**Figure 8.2.23: Lift coefficient versus angle of attack for the DLR-F6<sup>36-38</sup>**



**Figure 8.2.24: Lift versus drag for the DLR-F6** <sup>36-38</sup>

### 8.2.4 NASA CANARD FIGHTER

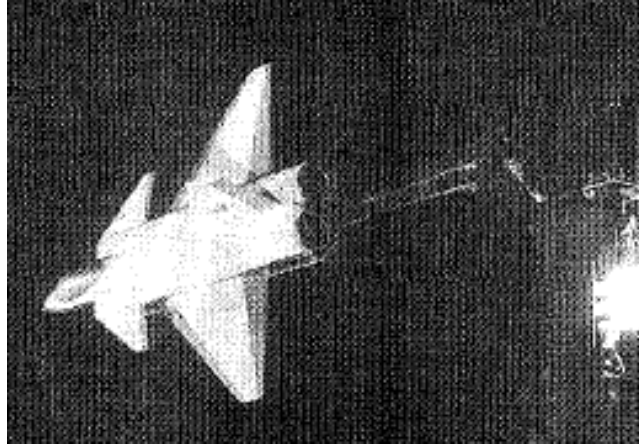
The final two geometries in this testing suite deal with stress testing of the relaxed wake-strand model. Both geometries have canards and main wings which involve wake interference in the loads as well as complex wake relaxation. The similarity between the geometries ends there, however. The NASA Canard Fighter <sup>39</sup> was designed in 1983 purely as a testing model for improving the understanding of such canard-wing interactions. It has a configuration of a high subsonic combat aircraft similar to the Swedish Viggen and in recent years the French Rafale and the European Eurofighter designs <sup>39</sup>. Unlike these real world geometries, however, the NASA design was a

simplified model with straight rectangular section and primary focus was given only to the canard-wing aerodynamics<sup>39, 41-43</sup>.

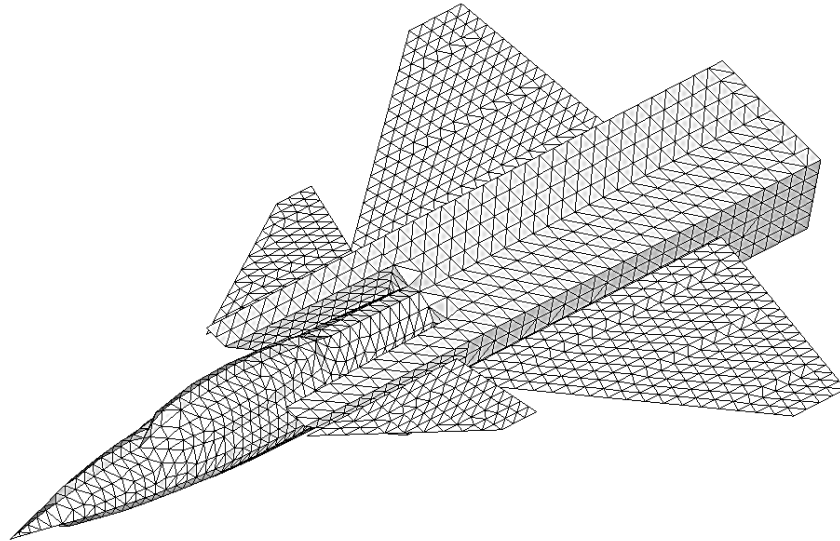
The model of the Canard-Fighter (shown in Figure-8.2.25 as mounted in the Ames wind tunnels) used for this effort was reconstructed from the detailed dimensions and scaling information provided in the work by Stoll and Koenig<sup>39</sup> at NASA Ames in 1983. This reconstructed geometry is shown in Figure-8.2.26 following auto-meshing cleanup. The mesh had 2,383 facets about the symmetry plane and was specifically modeled to include thin main wings and canard. This is in accordance with the very low thickness of the NACA 65A004 airfoil<sup>40</sup> used on the NASA model for both the surfaces. The savings in computation time as a result of this action is significant (a reduction of 23% was observed). And as outlined later, the effect on the final results is not noticeable.

Also in accordance with the wind tunnel model<sup>39</sup>, the engine intakes and the base of the model were modeled with facets but otherwise were not included with a velocity inlet boundary. The wing and canard mesh patch interface with the fuselage is non-conformal and was not fixed. The solver adapted to this condition without any oscillations given the general alignment of the facets in those regions.





**Figure 8.2.25: The Canard Fighter model on a mount at NASA Ames tunnel <sup>39</sup>**



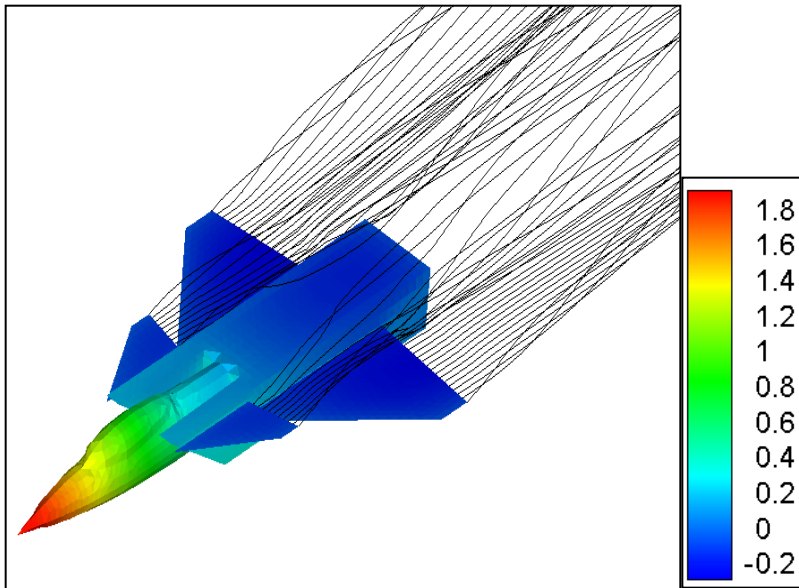
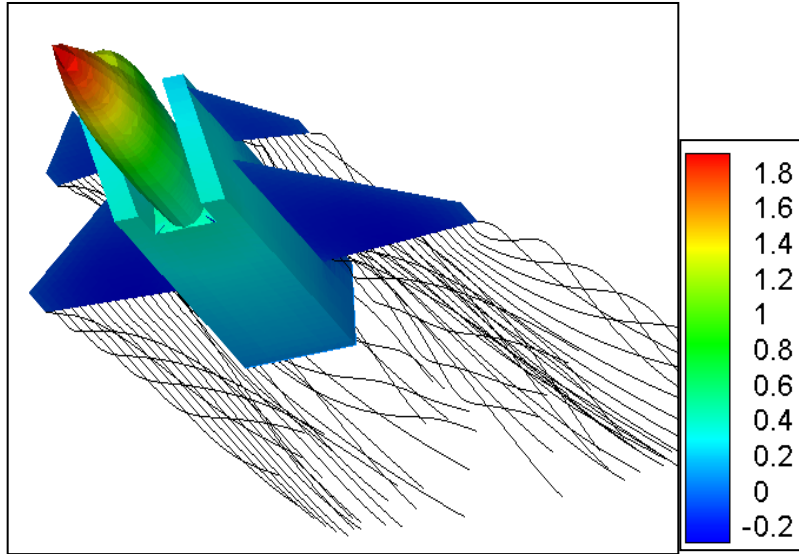
**Figure 8.2.26: Auto-mesher results for the Canard Fighter post CAD design**

The solver was run for a  $1e-4$  convergence setting and the Trefftz plane was marked 5.0 meters behind the main wing trailing edges (the model overall length as used in the solver is 5.053 meters). As such the Trefftz location corresponded to roughly one body length past the last facet on the mesh and this was considered a safe point for the termination of the wake as a tradeoff between wake fidelity and computation time. The

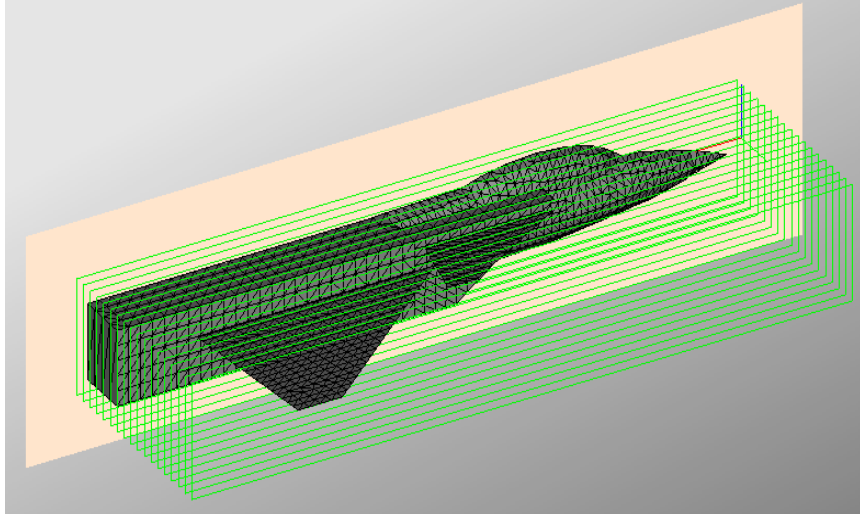
results of the vorticity-solver for the geometry are shown in Figure-8.2.27 for an AOA of  $10^\circ$ .

Figure-8.2.27 is also a good illustration of the stability of the wake strands. For this angle of attack, the wake from the canard moves above the main wing by a significant distance. This is partly due to the higher elevation of the canard relative to the wing and partly because of the fluid flow angles. The high incidence of the flow also creates powerful tip vortices for both the canard and the main wing and is easily seen in Figure-8.2.27. The vortices leaving the canard are under the influence of the main wing and the fuselage and as a result they do not spiral in a manner similar to the main wing strands. The strand emanating from the canards near the root are also pulled above and around the top surfaces of the fuselage as a result of the local vorticity gradients caused by the sharp angles on the geometry.

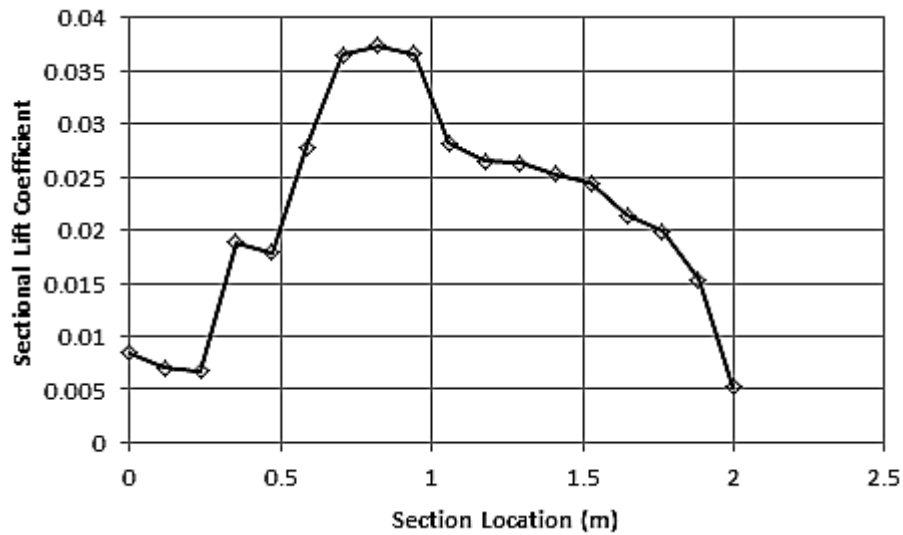
The aerodynamic loads were evaluated using 18 ICL planes (given the relatively coarse mesh) as shown in Figure-8.2.28. The sectional lift-coefficient as a function of span-wise location for an AOA of  $10^\circ$  is shown in Figure-8.2.29. As for the case of the F-18, the figure illustrates the important additive and subtractive effects between the main wing, canard and the fuselage. Given the simplified geometry involved, there is very little numerical noise in the ICL distribution.



**Figure 8.2.27: Vorticity distribution on the Canard Fighter (vorticity units =  $\text{m}^2/\text{sec}$ ; AOA =  $10^\circ$  and free-stream velocity = 1 m/sec)**



**Figure 8.2.28: ICL distribution about the symmetry plane**



**Figure 8.2.29: Lift coefficient span-wise distribution about the symmetry plane**

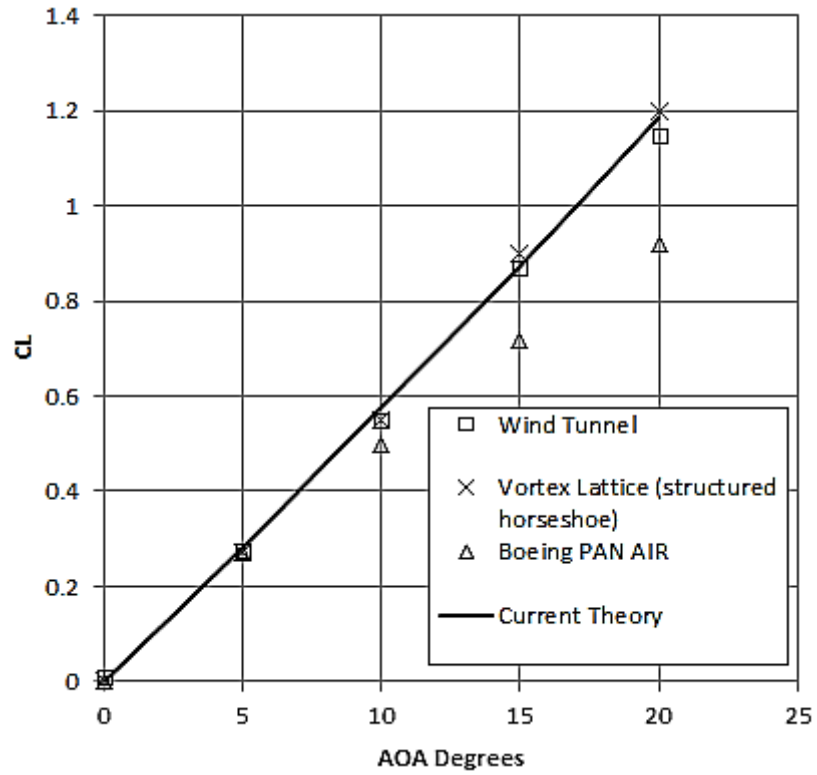
The integrated aerodynamic loads were compared with the experimental results obtained from the work of Stoll and Koenig<sup>39, 41-43</sup> at NASA Ames in 1983. Further comparison is made with the nonlinear Vortex Lattice Solver (VLS) used at Ames<sup>44</sup> by the authors, as well as with the Boeing PANAIR pressure-solver also used to compare

with the experimental results <sup>45</sup> (Figure-8.2.30). The VLS solution simplified the geometry by removing the fuselage and extending the wing and the canard to the centerline of the geometry <sup>44</sup>. It also maintained the surfaces as thin. The PANAIR solution modeled the geometry entirely as a closed and manifold body for a pressure-integrated solution <sup>45</sup>.

As seen from Figure-8.2.30 for the lift evaluations, the current approach closely approximates the wind-tunnel results. The VLS predictions are also close to the measured values. However, the PANAIR results deviate quickly from the measured data with increasing angle of attack. The PANAIR deviation is expected <sup>45</sup> given the low facet count and poor geometric fidelity of the model: as outlined earlier in this document, pressure-solvers remain limited by the fidelity of the mesh used. However, the reason the non-linear VLS solution matches experimental data so accurate is because it models the rollup of free vortices similar to the current approach's relaxed strand model <sup>44</sup>. More importantly, the model has the ability to simulate vortex separation from leading, side and trailing edges using free-separation vortices composed of discrete vortex segments <sup>44</sup>. The current approach does not model free-vortex separation from the leading edges and this remains one of the major future objectives. The accuracy of the Ames non-linear VLS solver is a classic example of enhancing the simplified VLS solvers for improved accuracy and physics <sup>14, 44</sup>.

Figure-8.2.31 illustrates the results for the model with and without the canards on the geometry. The comparisons here are restricted only to the wind-tunnel results given the absence of similar data from the NASA references cited earlier. The current approach

accurately captures the presence and absence of the canard from low AOA up to the point of vortex breakdown above the main wing surfaces.



**Figure 8.2.30: Lift coefficient versus angle of attack for the Canard Fighter** <sup>44-45</sup>

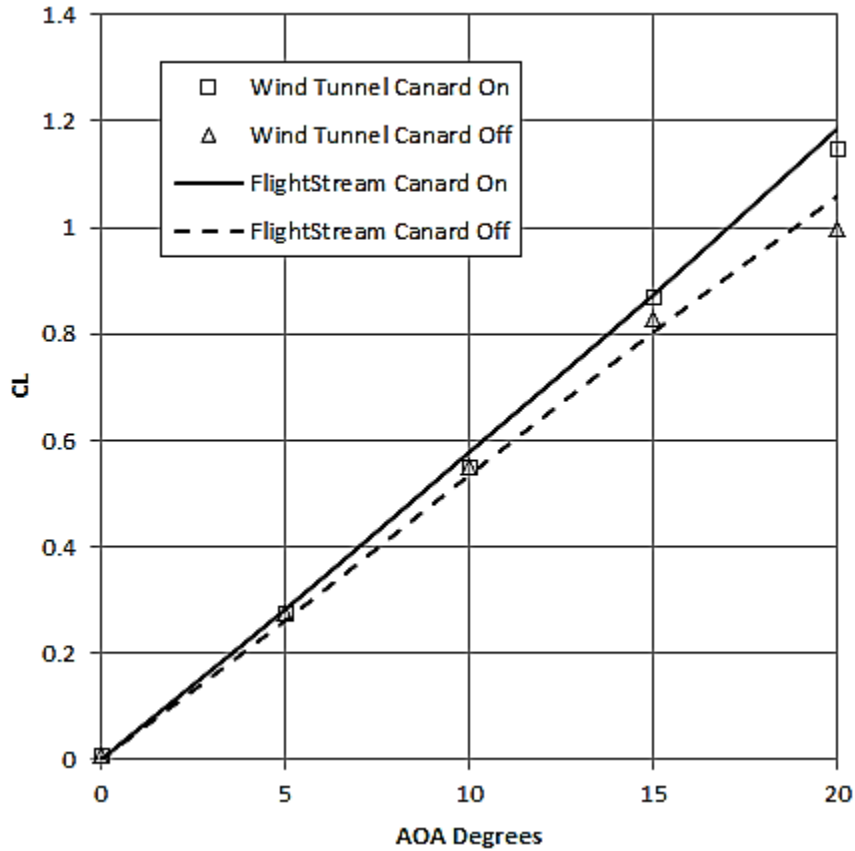


Figure 8.2.31: Lift versus angle of attack for the Canard Fighter <sup>44-45</sup>

### 8.2.5 RUTAN VARIEZE

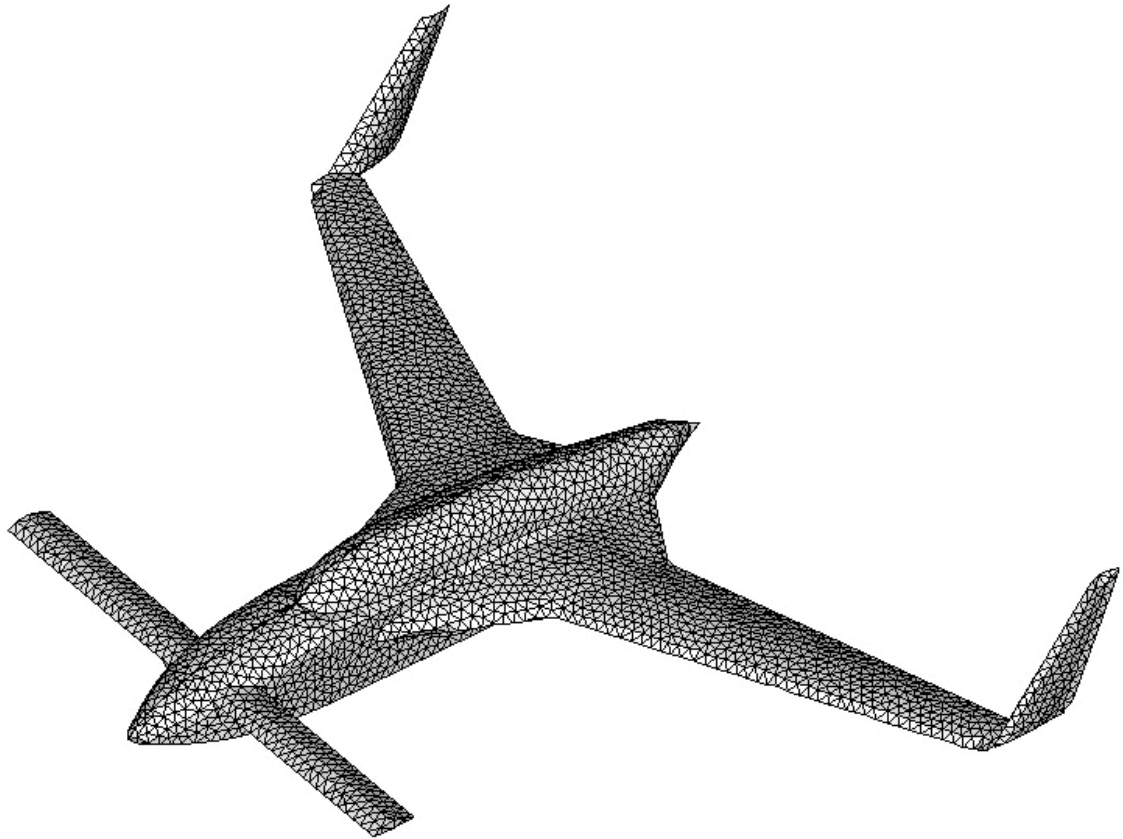
The final geometry used in this testing suite increases the stress on the relaxed wake-strand model by combining a canard-wing with a realistic fuselage and more complex surface blending. The model chosen for this test is the Rutan Varienze <sup>46</sup> which is a small civilian pusher design as shown in Figure-8.2.32. The testing goals remain the same as those for the NASA Canard-Fighter <sup>39</sup>.



**Figure 8.2.32: The Rutan VariEze**

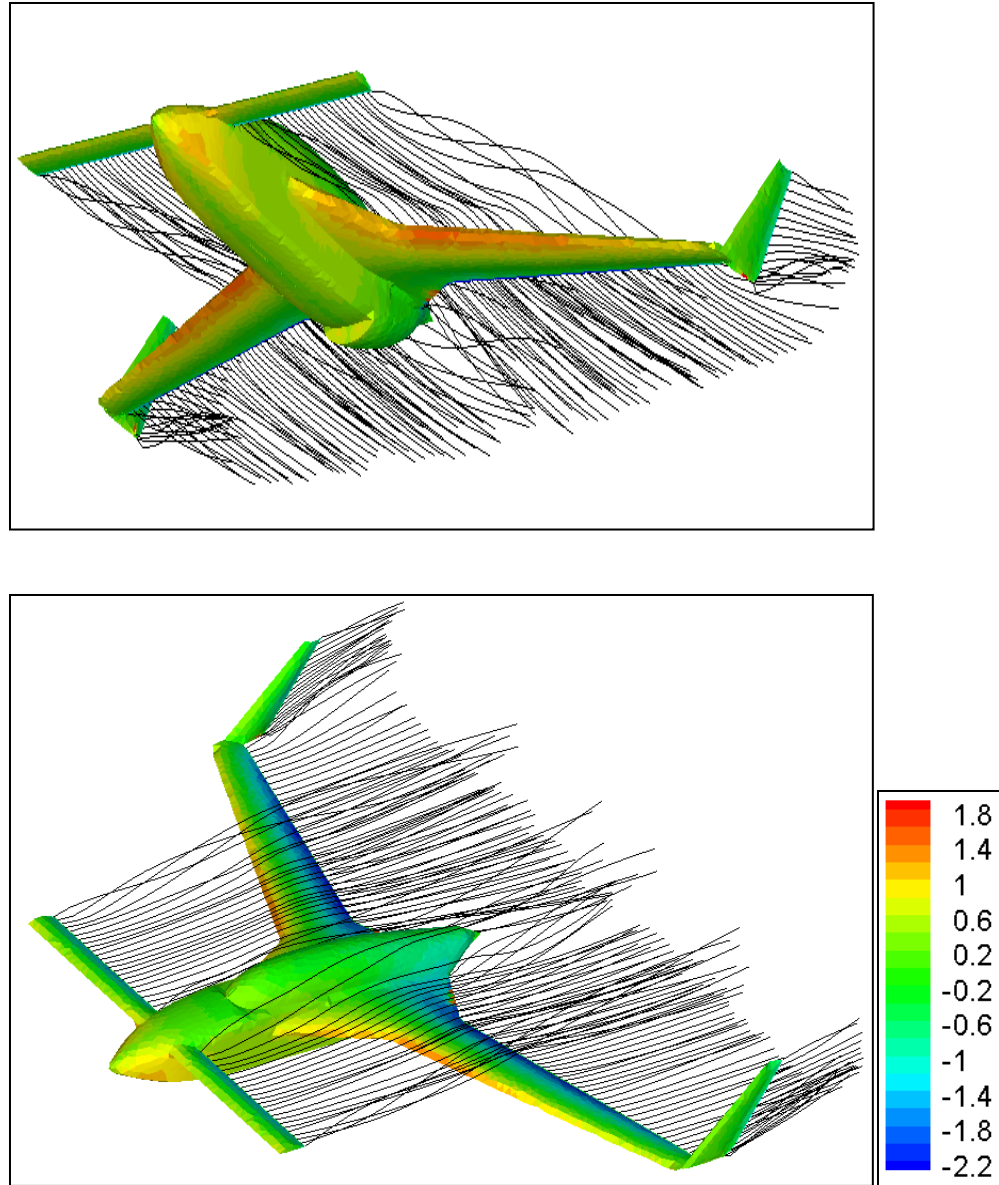
The geometry is designed using the detailed CAD information provided by the work of Long <sup>47</sup> in 1985 as part of a NASA research effort. The CAD was modified to remove the undercarriage fairings and booms as well as the propellers. The propeller nose cone was smoothed and blended into the rear fuselage. The vertical stabilizers at the wingtip were made into thin surfaces. Smaller fairings and antennae were removed as part of the standard auto-mesher de-featuring. The final resulting mesh is shown in Figure-8.2.33. The wing meshes had significant perturbations but were found to be irrelevant to the overall solver fidelity, in line with the robustness expectations following the DLR-F6 geometry testing done previously. Unlike the Canard-Fighter, the vertical stabilizers were modeled as lifting surfaces given their location on the main wing tips <sup>46</sup>. The under-fuselage engine air intake was not modeled as a velocity inlet and was closed off as a solid surface. The geometry was modeled with a symmetry plane boundary.





**Figure 8.2.33: Auto-mesher results for the Varieze post CAD design**

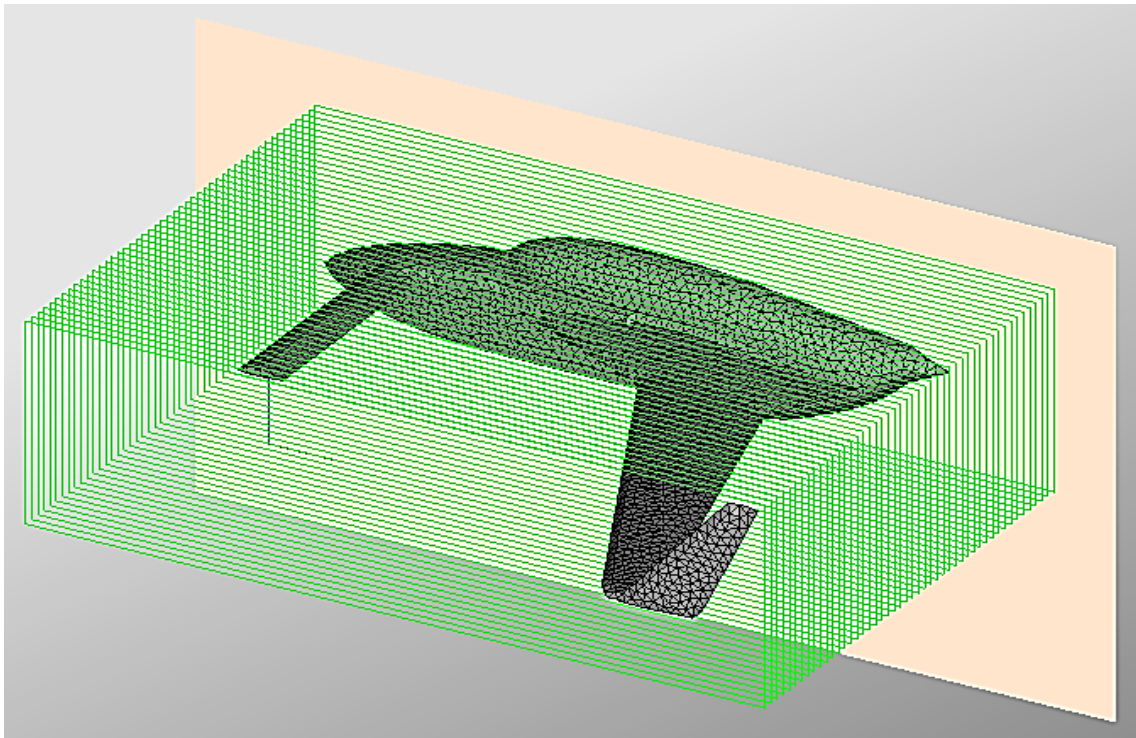
The solver was run for a  $1e-4$  convergence setting and the Trefftz plane was marked 5.0 meters behind the main wing-trailing edges. The results of the vorticity-solver for the geometry are shown in Figure-8.2.34 for an AOA of  $10^\circ$ . As discussed for the same geometry in Section-4, the wake strands are seen to show complex behavior over the wing surfaces and next to the fuselage and require more iterations (and hence more computation time) than for other relatively simpler cases such as the DLR-F6 geometry.



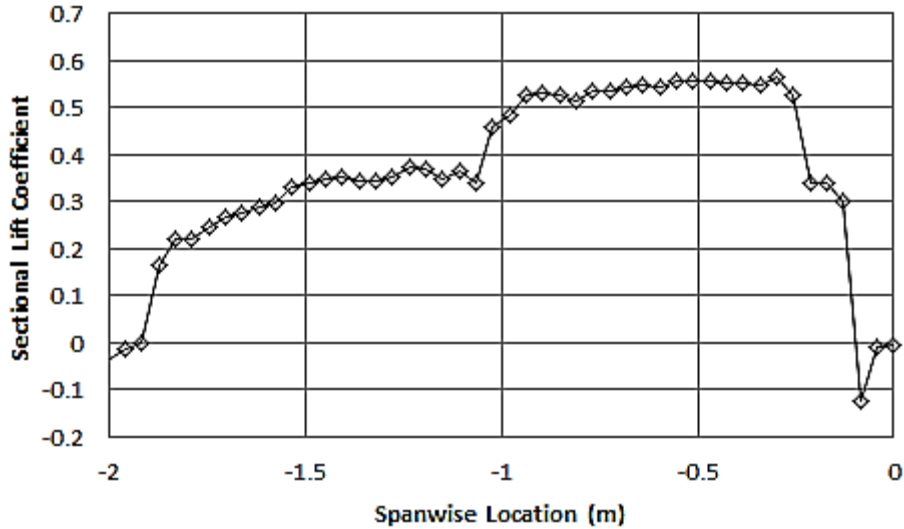
**Figure 8.2.34: Vorticity distribution on the Varieze (vorticity units = m<sup>2</sup>/sec; AOA = 10° and free-stream velocity = 1 m/sec)**

The aerodynamic loads were evaluated using 48 ICL planes (using the average mesh edge-size as discretization length) as shown in Figure-8.2.35. The sectional lift-coefficient as a function of span-wise location for an AOA of 10° is shown in Figure-

8.2.36. As for the case of the Canard-Fighter, the figure illustrates the important additive and subtractive effects between the main wing, canard and the fuselage. Unlike the Canard-Fighter, however, the Varieze has substantially more numerical noise in the ICL distribution and has convergence history behavior similar to that of the DLR-F6, which also had substantial ICL distribution scatter due to mesh perturbations.



**Figure 8.2.35: ICL distribution about the symmetry plane**



**Figure 8.2.36: Lift coefficient span-wise distribution about the symmetry plane**

The integrated aerodynamic loads were compared with the experimental results obtained from the work of Long at NASA Langley <sup>47</sup> in 1985. The flow is incompressible with only Mach 0.1 flow. The tests on the Varieze shed important light on the coupling effects of the relaxed wake flow over lifting surfaces and their corresponding effect on the overall lift generation <sup>46-47</sup>. Tests were done on the geometry with and without canards and with and without main-wing <sup>47</sup>. The results of these tests for the lift generated are presented in Figure-8.2.37.

From Figure-8.2.37, it is possible to see the coupling effects clearly. The tests done with individual components (canard or wing) are plotted and in both cases the current approach matches them accurately within the appropriate AOA range prior to flow separation. However, a plot has been made to show the decoupled sum of the wing and canard lift coefficients. This curve is parallel to the experimental data for the coupled sum of the canard and the wing. This is the coupling effect of the canard wake moving

over the wing surfaces that causes the overall lift coefficient to reduce. When run with the complete geometry setup, the current approach captures the effect clearly and the matches the measured data closely. This test underlines the importance of (and need for) relaxed wakes for complex aerodynamically-coupled geometries such as the Varieze.

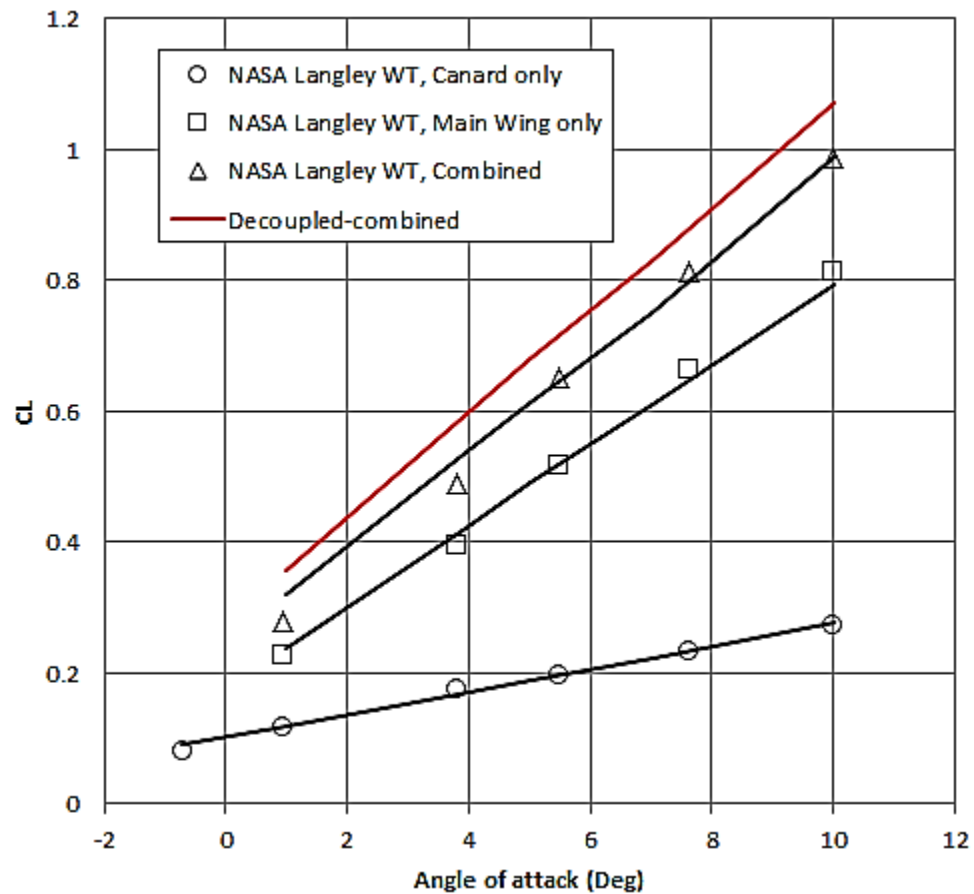


Figure 8.2.37: Lift coefficient versus angle of attack for the Varieze <sup>47</sup>

## 9. CONCLUSIONS

A method for the application of vorticity based potential-flow solvers to unstructured surface meshes has been created through the use of a novel method for aerodynamic loads. This method is designed to maintain the advantages of the vorticity based solvers while attempting to remove the limitations of geometrical inputs associated with the philosophical approach. A discussion on the necessity and advantages of this approach has been presented. It was shown for the case of simple geometric objects such as the sphere that 65% reductions in solver iterations could result from the use of an unstructured mesh (with inherent higher mesh quality) versus a converted structured mesh. In addition, a reduction of ~54% in mesh size was obtained as a result of higher mesh quality. Overall, the combination of reduced mesh size and higher quality led to a reduction of about ~88% in overall solver CPU time.

The theoretical underpinnings of the current methodology have been elaborated on in this document. A classical test case is presented in the thin elliptical wing for which an analytical solution is known from lifting line theory. Comparisons of the current numerical approach with this analytical solution show close approximation (error variation within 2.5-3%). The linearized nature of the underlying theoretical foundation for the current theory was shown using the F-18 geometry. In accordance with the stated

objectives of this effort, stability metrics were developed for the surface vorticity and the relaxed wake strands. These parameters have since been used to demonstrate the stability of the solver under stressful geometry conditions.

To further capitalize on the improved possibilities offered by the use of unstructured surface meshes, a modification to the evaluation of skin-friction coefficients using surface vorticity was developed to bypass the limitation of existing approaches. The heuristic arguments in support of the current formulations have been presented and fit well within experimentally observed behavior. A novel unstructured wake-strand model has been developed to allow handling of wakes emanating from unstructured surface meshes and test cases show the stability of the vorticity shedding as well as the applicability of its novel features such as adaptive time stepping and vorticity decay models used to eliminate mathematical singularities.

An attempt has been made to extend Potential-Flow solvers to the current industry surface meshing and numerical solver standards. This automated design pipeline is considered robust enough to allow integration with optimizers within an MDO environment no longer restricted in their geometrical design spaces. In accordance with the objective of this effort, metrics were developed for surface mesh quality and automated meshing tools were developed to convert input geometries into solver ready meshes. The effect of size based meshing and mesh quality on solution fidelity was tested. It was determined that surface meshes with uniform sizes and quality offered superior results and quicker convergence. RMS error reduction between 10-20% between higher and lower quality meshes.

Several improvements in solver methodology applicable in an unstructured mesh solver have also been developed. These included the FDMR and persistent induction matrix concepts. The use of the FDMR on the F-35 fighter geometry resulted in 80% reduction in solver iterations for the same convergence limits and an order of magnitude improvement in solver convergence ( $10^{-3}$  versus  $10^{-2}$  for the non FDMR case) for the same number of iterations. Testing using the persistent induction matrix concept allowed reductions between 65-85% in CPU time for the same number of iterations for large surface meshes (>10,000 facets).

Improvements to the boundary physics models have been made to take advantage of the unstructured meshes. Relaxation of the Neumann condition on user-specified facets or topological patches has made it possible to create velocity inlets. Additionally, the detection of the trailing edges has been automated, thereby removing a possible limitation compared to structured mesh solvers. Use of the Mirror-plane boundary condition has allowed substantial reductions in solver time for steady-state, symmetric flow geometries. Tests on the DLR-F6 geometry with and without the mirror plane boundary showed that reductions of up to ~65% could be obtained through the use of the mirror plane. Other geometries allow for varying savings in CPU time with the general trend towards increasing savings for large meshes. In general, reductions of up to 75% have been noted for some designs such as the Rutan Varieze.

A test-suite of basic shapes and bodies has been tested to evaluate the fidelity of the new approach. This suite tested the current solver over thin wing plan-forms of various types. It was observed that the current theory is accurate for rectangular wings,



sweep wings, axisymmetric bodies and delta wings with all aerodynamic load predictions within 5-10% of experimental data at low angles of attack. At higher angles of attack, expected deviations between predictions and experiments are seen and these are attributed to the non-linearity of the physics close to, and after, flow separation. The validity of the Kutta condition is shown for the axisymmetric bodies when the vorticity shedding is eliminated and no trailing edges are marked.

An advanced test-suite has also been developed to stress-test the solver as well as to test out the geometry handling pipeline required to handle these test cases. Through these tests, it was determined that the solver is able to provide high fidelity results for a wide variety of test cases. Results show a high degree of fidelity for typical fighter configuration geometries such as the F-16 and the F-18. In the case of both these geometries, the aerodynamic loads were found to be within  $\pm 10\%$  of the experimentally obtained values. For the case of the F-16 it was shown that the range of applicability of the compressibility correction factor extends to Mach Numbers between 0.8-0.9. The F-18 test case showed substantial improvement of numerical drag prediction with the current approach over contemporary solvers.

Tests were also done for commercial airliner representative test cases. The DLR-F6 geometry was used and aerodynamic load predictions were within  $\pm 5-10\%$  with slight over-predictions in drag and slight under-prediction in lift coefficient. This test was compared with CFD results using similar sized surface meshes as a starting point and the results showed substantially lower fidelity of solutions compared to experiments. In

addition, a reduction in CPU time was noticed from ~36 minutes (CFD) to about 3.5 minutes (current solver); a reduction of ~90%.

Testing also included canard equipped geometries to determine the coupling effect of the lifting components in such geometries. Two designs were tested: the NASA Canard Fighter and the Rutan Varienze. In both cases the coupling effect of the canard on the main wing was observed under differing conditions and compared with experimental data. Results showed aerodynamic load predictions within  $\pm 5-10\%$  and the coupling effect was readily visible through the use of the relaxed wake-strand model of vorticity shedding.

Finally, the stated objective of this effort to develop a solver able to handle diverse geometrical cases hitherto restricted to Navier-Stokes solver has also been reasonably demonstrated by the various models and designs tested within the basic and advanced testing suites. Realistic designs such as the F-16, F-18 and Varienze geometries as well as reference cases such as the DLR-F6 (for CFD studies) and the NASA Canard Fighter (for legacy and contemporary potential-flow studies) illustrate the spread of designs that can be readily tested for aerodynamic performance using the Method of Integrated Circulation.

## REFERENCES

1. Reneaux, J., and Thibert, J-J., "*The Use of Numerical Optimization for Airfoil Design*," AIAA 3rd Applied Aerodynamics Conference, AIAA Paper 85-5026, Colorado Springs, CO, Oct. 1985
2. Ahuja, Vivek and Hartfield, Roy, "*Optimization of Scramjet Combustor designs using Genetic Algorithms and CFD*", AIAA 2008-5925 presented at the AIAA International Multidisciplinary Design and Optimization Conference (MDO), Victoria City, British Columbia, Canada, September 2008
3. Ahuja V., Hartfield R. J., "*Optimization of UAV Geometries for Aerodynamic Performance using Advanced Paneling Aero-Propulsive Paneling Models and Genetic Algorithms*", 6th AIAA Multidisciplinary Design Optimization Specialist Conference, Orlando, Florida , April 2010
4. Ahuja, V and Hartfield, Roy, "*Optimization of Air-breathing Hypersonic Aircraft Design using Euler Codes and Genetic Algorithms*", to be presented at the 46th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit , Nashville, Tennessee, USA, in July 2010
5. Sytsma, H.S., Hewitt, B.L. and Rubbert, P.E., "*A comparison of panel methods for subsonic flow computation*," AGARD AG-241, 1979

6. Wang, H.T., "*Comprehensive Evaluation of Six Thin-Wing Lifting Surface Computer Programs*", Naval Ship Research and Development Center, Rept. 4333, June 1974
7. Langan, T.J. and Wang, H. T., "*Evaluation of Lifting Surface Programs for Computing the Pressure Distribution on Planar Foils in Steady Motion*", Naval Ship Research and Development Center, Rept. 4021, May 1973
8. Maskew, B., "*Program VSAERO Theory Document*", NASA Contractor Report 4023, NASA-11945, September 1987
9. Maskew, B., "*Program VSAERO, A computer program for calculating the nonlinear aerodynamic characteristics of arbitrary configurations, User's Manual*," NASA CR-166476, November 1982
10. Maskew, B., "*A surface vorticity method for calculating the pressure distribution over airfoils of arbitrary thickness and camber using two-dimensional potential flow*," Loughborough University of Technology report TT 6907, 1969
11. Epton, M.A. and Magnus A.E., "*PAN AIR- A computer program for predicting subsonic or supersonic linear potential flows about arbitrary configurations using a higher order panel method*," Version-I, Theory Document, NASA CR-3251, March, 1990
12. Ashby, D., Dudley, M., and Iguchi, S., "*Development and validation of an advanced low-order panel method*," NASA TM 101024, 1988
13. Dvorak, F.A., Woodward, F.A. and Maskew, B., "*A three-dimensional viscous/Potential flow interaction analysis methods for multi-element wings*," NASA CR-152012, July 1977

14. Katz, J. and Plotkin, A., "*Low-speed aerodynamics: from wing theory to panel methods*," McGraw-Hill Inc., ISB-0-07-050446-6, 1991
15. Maskew, B., "*Calculation of the three-dimensional potential flow around lifting non-planar wings and wing-bodies using a surface distribution of quadrilateral vortex rings*," Loughborough University of Technology, Dept. of Transport Tech. Report TT 7009, September 1970
16. Purcell, E.W., "*The vector method of solving simultaneous linear equations*," J. Math. Physics, Vol. 23, No. 180, 1953
17. Lamb, H., "*Hydrodynamics*," 6<sup>th</sup> ed., Cambridge University Press, 1932
18. Batchelor, G.K., "*Axial flow in trailing line vortices*," Journal of Fluid Mechanics, no. 20, pp. 645-658, 1964
19. Batchelor, G.K., "*An introduction to fluid mechanics*," Cambridge University Press, 1967
20. Filotas, L. T., "*Solution of the lifting line equation for twisted elliptical wings*," J. Aircraft, vol. 8, no. 10, pp. 835-936, 1971
21. Ahuja V., Hartfield R. J., "*Preliminary Drag Prediction Using Advanced Paneling Schemes*," 48<sup>th</sup> AIAA Aerospace Sciences Meeting, Orlando, Florida, January, 2010
22. Jobe, C.E., "*Prediction and Verification of Aerodynamic Drag, Part I: Prediction*" Progress in Astronautics and Aeronautics, edited by C. E. Eugene, Vol. 98, AIAA, New York, 1985
23. Prandtl, L., "*Applications of modern hydrodynamics to aeronautics*," Report 116, NACA, 1921

24. Purvis, J.W., "*Simplified solution of the compressible lifting surface problem,*" M.S. Thesis, Aerospace Engineering Department, Auburn University, Auburn, Alabama, August, 1975
25. Schlichting, H., and Truckenbrodt E., "*Aerodynamik des Flugzeuges,*" vol. 2, chapter 7, Springer-Verlag, Berlin, 1969
26. Wentz, W.H. and Kohlman, D.L., "*Wind tunnel investigations of vortex breakdown on slender sharp-edged wings,*" NASA Research Grant NGR-17-002-043, University of Kansas Center for Research Inc., Report FRL 68-013, November 27, 1968
27. Polhamus, E.C., "*Application of the leading-edge suction analogy of vortex lift to the drag due to lift of sharp edge delta wings,*" NASA TN D-4739, August 1968
28. Polhamus, E.C., "*A concept of the vortex lift of sharp-edge delta-wings based on a leading-edge suction analogy,*" NASA TN D-3767, 1966
29. Wentz, W.H. and McMahon, M.C., "*Further experimental investigations of delta and double-delta wing flow fields at low speeds,*" NASA CR-714, February, 1967
30. Weber J., Brebner G.G., "*Low speed tests on a 45-deg swept back wings, Part-I: pressure measurements on wings of aspect ratio 5*", ARC R&M 2882, 1958
31. Weissinger, J., "*The lift distribution of swept-back wings,*" NACA TM-1120, 1947
32. Nguyen, Luat T. et.al, "*Simulator study of stall/post-stall characteristics of a fighter airplane with relaxed longitudinal static stability,*" NASA TP-1538, December, 1979

33. Webb, T.S., Kent, D.R., “*Correlation of F-16 aerodynamics and performance predictions with early flight-test results*,” AGARD conference proceedings, n 242, October 11-13, 1977
34. Banks, D.W., “*Wind tunnel investigation of the forebody aerodynamics of a vortex lift fighter configuration at high angles of attack*,” SAW Paper 88-1419. October, 1988
35. Browne, L., Katz, J., “*Application of panel methods to wind-tunnel wall interference corrections*,” 28<sup>th</sup> Aerospace Sciences meeting, Reno, Nevada, January 8-11, 1990
36. Levy, D. W., Zickuhr, T., Vassberg, J., Agrawal, S., Wahls, R. A., Pirzadeh, S., and Hensch, M. J., “*Summary of Data from the First AIAA CFD Drag Prediction Workshop*,” 40th AIAA Aerospace Sciences Meeting and Exhibit, AIAA Paper 2002-0841, Reno, NV, Jan. 2002
37. Laflin, K., Klausmeyer, S. M., Zickuhr, T., Vassberg, J. C., Wahls, R. A., Morrison, J. H., et al., “*Data Summary from Second AIAA Computational Fluid Dynamics Drag Prediction Workshop*,” Journal of Aircraft, Vol. 42, No. 5, 2005, pp. 1165–1178
38. Vassberg, J. C., Tinoco, E. N., Mani, M., Brodersen, O. P., Einfeld, B., Wahls, R. A., Morrison, J. H., Zickuhr, T., Laflin, K. R., and Mavriplis, D. J., “*Summary of the Third AIAA CFD Drag Prediction Workshop*,” 45th AIAA Aerospace Sciences Meeting and Exhibit, AIAA Paper 2007-0260, Reno, NV, Jan. 2007
39. Stoll F., Koenig D.G., “*Large Scale Wind Tunnel Investigation of Close-Coupled Canard-Delta-Wing Fighter Model through high angles of attack*”, AIAA Aircraft

Design, Systems and Technology Meeting, Fort Worth, Texas, 17-19 October  
1983

40. Polhamus, E.C., Campbell, G.S., “*Aerodynamic characteristics of a wing with unswept quarter-chord line, aspect ratio 2, taper ratio 0.78 and NACA 65A004 airfoil section,*” NACA Research Memorandum L50A18, March 08, 1950
41. Stoll F., Koenig D.G., “*Large Speed Wind Tunnel Measurements of a Canard Controlled Fighter Model through high angles of attack*”, NASA TM-84403, 1983
42. Miller, S.G. and Youngblood, D. B., “*Application of USSAERO-3 and the PANAIR Production Code to the CDAF Model A Canard/Wing Configuration*”, AIAA Paper 83-1829, 1983
43. Minter, E. A, Yates, R. W., “*Wind Tunnel Tests of a 0.4 Scale Fighter Model at High Angles of Attack-Analysis of Pressure Data*”, NASA CR-166198, 1981
44. Levin, D., and Katz, J., “*A vortex-lattice method for the calculation of the non-steady separated flow over delta wings,*” AIAA Paper 80-1803, 1980
45. Miller, S.G. and Youngblood, D.b., “*Application of USAERO-3 and the PANAIR Production code to the CADF Model A canard/wing configuration,*”, AIAA paper 83-1829, 1983
46. Rutan, Burt, “*Development of a small high-aspect-ratio canard aircraft,*” 1976 report to the Aerospace Profession, *Tech Rev.*, vol. 13, no. 2, Soc. Exp. Test Pilots, 1976, pp-93-101
47. Long, P.Y., “*Wind tunnel investigation of a full-scale canard-configured general aviation airplane,*” NASA TP-2382, March, 1985



48. Karamcheti, K., "*Principles of Ideal-Fluid Aerodynamics*," R.E. Krieger,  
Malabar, FL, 1980

## APPENDIX-A

### DERIVATION OF THE BIO-SAVART LAW FOR VORTEX RINGS

The continuity equation for incompressible and inviscid equation can be written as:

$$\nabla \cdot v = 0 \quad (\text{A.1})$$

The velocity field can be written as the curl of a vector field  $B$ , such that:

$$v = \nabla \times B \quad (\text{A.2})$$

Since the curl of a gradient vector is zero,  $B$  is indeterminate to within the gradient of a scalar function of position and time, and the vector field  $B$  can be selected such that:

$$\nabla \cdot B = 0 \quad (\text{A.3})$$

The vorticity then becomes:

$$\gamma = \nabla \times v = \nabla \times (\nabla \times B) = \nabla(\nabla \cdot B) - \nabla^2 B \quad (\text{A.4})$$

Substituting Equation-A.3 into Equation-A.4,

$$\gamma = -\nabla^2 B \quad (\text{A.5})$$

The solution to Equation-A.5 is arrived at using Green's theorem by Karamcheti<sup>48</sup> and is written as:

$$B = \frac{1}{4\pi} \int \frac{\gamma}{|r|} dV \quad (\text{A.6})$$

Combining Equation-A.2 with Equation-A.6 gives:

$$v = \frac{1}{4\pi} \int \nabla \times \frac{\gamma}{|r|} dV \quad (\text{A.7})$$

Since the infinitesimal volume can be split into an infinitesimal area normal to the Vorticity and an incremental distance  $dl$  on the filament so that:

$$dV = ds \cdot dl \quad (\text{A.8})$$

Also, circulation on the infinitesimal area  $ds$  is given as:

$$\Gamma = \gamma ds \quad (\text{A.9})$$

Combining Equation-A.8, Equation-A.9 and Equation-A.7, we get:

$$v = \frac{1}{4\pi} \int \nabla \times \frac{\Gamma}{|r|} dl = \frac{\Gamma}{4\pi} \int \frac{dl \times r}{|r|^3} \quad (\text{A.10})$$

Equation-A.10 is the required vector formulation of the Biot-Savart law as used in the current effort.