Theory Guide

Welcome to the official theory guide for FlightStream by Research In Flight!

Learn

Back to RIF home

Last updated: Jan 4, 2024

Supersonic Extension

Unstructured surface vorticity solvers are an extension of structued thin-surface vorticity based solvers (e.g. vortex lattice methods).

$$dV_i = \frac{\Gamma}{4\pi}\frac{dl \times r}{r^3}$$

Further reading

- Read more

Last updated: Jan 4, 2024

Previous
Actuator Disc

Actuator Disc

Propeller

Conway has shown that the propeller effects on a potential flow environment can be represented by the superposition of known-strength vortex systems that satisfy both the Helmholtz Laws of vorticity as well as the propeller actuator conditions.

FlightStream utilizes the superposition of two vortex systems of the same vortex strength. The time-average of the bound and shed vorticity of the propeller is the superposition of the vorticity distributions in the plane of the propeller and the slipstream. There are two propeller vortex distributions which induce perturbations to the free-stream flow. The first is a lifting-line vortex horseshoe system simulating the lifting-line solution of a propeller disc made up of infinite numbers of blades. Such a system has the same vortex strength for all of its lifting-lines. The trailing vortices of the horseshoe system are made up of on strand shed along the axis of the disc and the other shed along tip of the disc. The system is shown in Figure-8(a). The second system of vortices are the shed vortex rings as shown in Figure-8(b). Such a system forms a vortex tube consisting of ring vortices distributed over the tube surface from the edge of the actuator disk and extending downstream to the Trefftz plane. The tube of ring vortices result from the root and tip vortices shed from the lifting lines in the actuator disc and as such also are assumed to have the same vortex strength as that of the lifting-line system

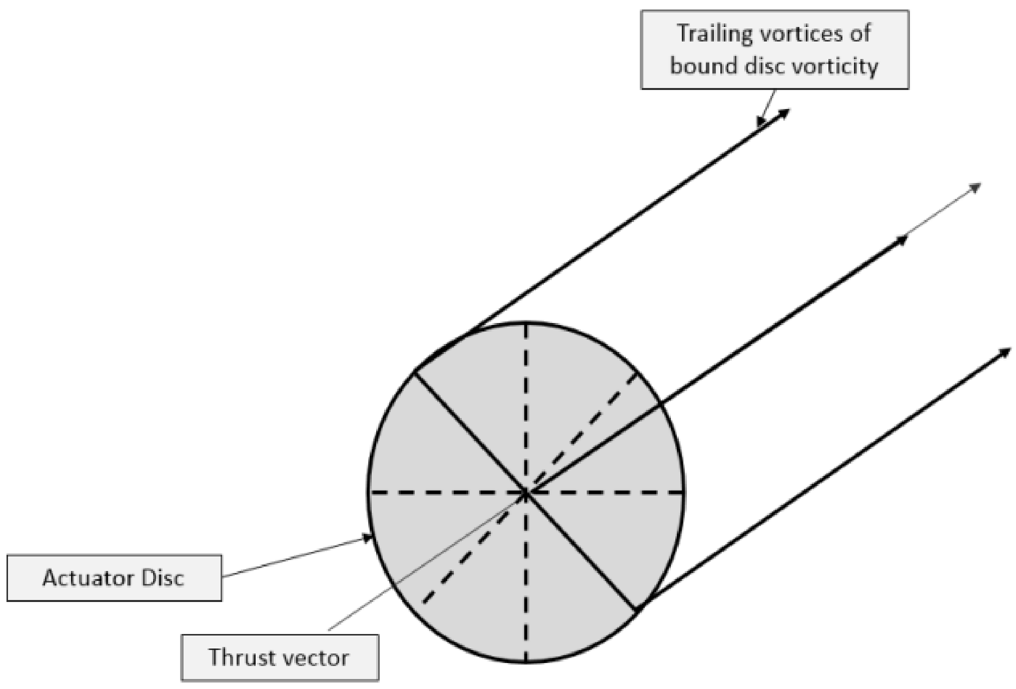

Fig. (1): Lifting-line horseshoe vortex system for the infinite blade number actuator disc

Last updated: Jan 10, 2024

Previous
Integrated Loads

Next
Supersonic Extension

Integrated Loads

Last updated: Jan 10, 2024

Previous
Unstructured Surface-Vorticity

Next
Actuator Disc

Unstructured Surface-Vorticity

Unstructured Vorticity on an Arbitrary Body

Unstructured surface vorticity solvers are an extension of structued thin-surface vorticity based solvers (e.g. vortex lattice methods). They depend on the evaluation of the circulation around a given geometry and use the Kutta–Joukowski formulation, which is applied to each panel for evaluation of the induced loads. In FlightStream, these methods are extended to work on arbitrary bodies defined by an unstructured surface mesh.

The derivation for the velocity induced by a linear segment of a face can be mathematically developed in terms of the coordinates of the two end vertices and the point of interest using the Biot–Savart Law. Consider a point P in space where a segment of a vortex ring is inducing a velocity. The velocity induced at P by an infinitely small segment of the ring edge is known from the application of the Biot–Savart law:

$$dV_i = \frac{\Gamma}{4\pi} \frac{dl \times r}{r^3} \qquad \text{Eq. (1)}$$

In Eq. (1), all variables on the right-hand side are easily evaluated from vector formulations of the planar geometry defined by the three points in space (two vertices of the vortex segment and one point of interest). Note that Eq. (1) also reduces to the two-dimensional vortex for the limiting case of infinitely long vortex. Equation (1) is the induced velocity by one segment of a facet. Therefore, to evaluate the velocity induced by the entire ring, we evaluate Eq. (1) for all three edges of the vortex ring to get the net velocity induction from that face as:

$$V_j = \Gamma_j \sum_{i=1}^{N_{edges}} V_i = \Gamma_j A_{j,p} \qquad \text{Eq. (2)}$$

The coefficient $A_{j,p}$ is the geometrical influence coefficient for the velocity induction at point P by the face j of the surface mesh.

Considering Eq. (2) the only unknown is the vorticity strength. This creates a system of equations with N unknowns that must be solved for Γ . The matrix must be evaluated in patches numerically during the inversion phase or the solver must be handed small patches of the mesh. In FlightStream, the mesh is split into solver partitions solved individually. Freezing remaining mesh sections at the previous iteration values. This approach creates an iterative solver. Given the nature of the geometry, an exact solution is not practical and instead the solver completes once the surface vorticity residual reaches some specified tolerance.

Convergence

The convergence criterion used is based on the stability of the surface vorticity 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 face vorticity. The wake instability is quantified as the rms change of the two-dimensional strand positions within the Trefftz plane.

Further Reading

On this page
Overview

Previous
Why FlightStream?

Next
Integrated Loads

Why FlightStream?

Industry Need

Current industry trends point toward an increasing use of optimization-based design principles. For complex systems, optimizers routines require hundreds or thousands of design evaluations. Since each of these designs must be tested for aerodynamic or fluid-dynamic performance, a balance between minimizing the computation time versus maintaining the fidelity of the solution and the validity of the physics must be made. This balancing represents a major limitation in the comprehensive use of multidisciplinary-optimization (MDO) design pipelines.

Navier-Stokes (e.g. RANS, LES) solvers are rapidly making inroads into the MDO environment, however this approach is restricted to simpler optimization cases. Pressure-based potential flow solvers are highly susceptible to imperfections in surface mesh quaility and refinement. Finally, vorticity based solvers are currently limited to structured thin surface geometries.

Why 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 face is bound by four vertices extracted from the rectangular structure of the underlying map (known as the U-V map). As such, any structured surface mesh is essentially quadrilateral in nature. Consequently, facets have to be “forced” to assume triangular shapes to meet geometrical demands. This is accomplished by vertex merging 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. In addition, forced U-V mappings ensure that regions of high curvature or concave warping are always populated with more facets than actually required. Furthermore, the quality of such facets is typically very poor. Finally, the typical quadrilateral facet has to be sanitized to establish its “effective” surface normal direction. With triangular faces, this is not required, given the deterministic nature of the triangle geometry in three-dimensional space. 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 face count. Despite the aforementioned shortcomings of the structured surface meshes, with potential flow solvers (and especially 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. Vorticity solvers require a face-edge pool to be split into bound and trailing vortices during the application of the Kutta–Joukowski formulation. This means that faces 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 bound and trailing vortices. The bound vortices are simply marked as those facet edges aligned along a certain numerical axis, and the other edges are marked as trailing. Note that a such marking usually only corresponds to the correct flowfield if the physical space warping and curvature are not significantly different from the alignment of the numerical axes. Furthermore, the classical induced load formulation requires knowledge of the vorticity in the downstream bound vortex. This reduces the computational burden on the solver and allows the evaluation of loads in a simple and straightforward manner.

Integrated Circulation. Journal of Aircraft, 55(6), 1719–1730. https://doi.org/10.2514/1.c035015

Last updated: Jan 4, 2024

Next
Unstructured Surface-Vorticity →

Theory Guide

Welcome to the official theory guide for FlightStream by Research In Flight!

Learn
Back to RIF home

!!! This site is still under construction and may be incomplete. Please contact support@researchinflight.com for specific questions.

Get the pdf version of this site

Last updated: Jan 15, 2024

Supersonic Extension

Unstructured surface vorticity solvers are an extension of structued thin-surface vorticity based solvers (e.g. vortex lattice methods).

$$dV_i = \frac{\Gamma}{4\pi}\frac{dl \times r}{r^3}$$

Further reading

- Read more

Last updated: Jan 4, 2024

On this page
Overview

Actuator Discs

Propeller

Conway has shown that the propeller effects on a potential flow environment can be represented by the superposition of known-strength vortex systems that satisfy both the Helmholtz Laws of vorticity as well as the propeller actuator conditions.

FlightStream utilizes the superposition of two vortex systems of the same vortex strength. The time-average of the bound and shed vorticity of the propeller is the superposition of the vorticity distributions in the plane of the propeller and the slipstream. There are two propeller vortex distributions which induce perturbations to the free-stream flow. The first is a lifting-line vortex horseshoe system simulating the lifting-line solution of a propeller disc made up of infinite numbers of blades. Such a system has the same vortex strength for all of its lifting-lines. The trailing vortices of the horseshoe system are made up of on strand shed along the axis of the disc and the other shed along tip of the disc. The system is shown in Figure-8(a). The second system of vortices are the shed vortex rings as shown in Figure-8(b). Such a system forms a vortex tube consisting of ring vortices distributed over the tube surface from the edge of the actuator disk and extending downstream to the Trefftz plane. The tube of ring vortices result from the root and tip vortices shed from the lifting lines in the actuator disc and as such also are assumed to have the same vortex strength as that of the lifting-line system

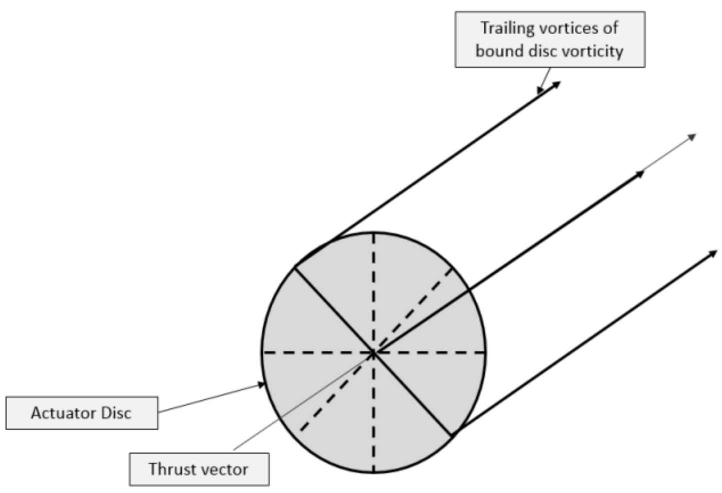

Fig. (1): Lifting-line horseshoe vortex system for the infinite blade number actuator disc

Last updated: Jan 10, 2024

Previous
Viscous Coupling
Next
Nomenclature

On this page > Overview

Integrated Loads

Vorticity induced lift, drag, and downwash can be calculated via:

$$w_y = \frac{1}{4\pi} \int_{b/2}^{b/2} \left(\frac{1}{y - y_0}\right) \frac{-d\Gamma_{y_0}}{dy} dy_0 \qquad \text{Eq. (1)}$$

$$L_i = \int_{b/2}^{b/2} \rho_\infty V_\infty \Gamma_y dy \qquad \text{Eq. (2)}$$

$$D_i = \int_{b/2}^{b/2} \rho_\infty V_\infty w_y dy \qquad \text{Eq. (3)}$$

Last updated: Jan 10, 2024

Previous
← Unstructured Surface-Vorticity

Next
Separation Modeling →

Nomenclature

Variable	Description
A	Face area
C_L	Coefficient of lift
C_l	Coefficient of lift, 2D
C_L	Coefficient of lift
C_d	Coefficient of drag, 2D
C_f	Coefficient of friction
$H\frac{\delta^*}{\theta}$	Shape factor
l	Dimensionless parameter by Thwaites or edge length
Ma	Mach number
r	Radial distance from point-to-point
Re	Reynolds number
Re_i	Reynolds number based on length, i
s	arg length coordinate
S_{ref}	Reference area
u_e	Boundary layer edge velocity
δ	Boundary layer thickness
δ^*	Displacement thickness
γ	Specific heat ratio
Γ	Circulation
μ	Fluid viscosity
ρ	Density
θ	Momentum thickness

Last updated: Jan 10, 2024

Separation Modeling

When air flows over a surface, it forms a boundary layer where the air's velocity changes from zero at the surface to the free stream velocity away from the surface. If the pressure in the direction of the flow increases (adverse pressure gradient), it can slow down the air in the boundary layer. If this adverse pressure is too severe, the boundary layer separates from the surface, leading to increased drag and possibly stall in the case of aircraft wings.

After the inviscid solution is calculated, FlightStream uses analytical models to estimate where the flow becomes separated.

Separation Criterion

FlightStream implements a modified Stratfod separation criterion to estimate boundary layer separation location. The Stratford criterion predicts a condition where this separation can be delayed or prevented, even under strong adverse pressure gradients. Stratfod derived an expression for predicting laminar boundary layer separation.

$$\bar{C}_p = 1.0 - \left(\frac{u}{U_{max}}\right)^{2.0} \quad \text{Eq. (1)}$$

$$\bar{X}^2 \bar{C}_p \frac{dC_p}{dx}^2 = 0.0104 \quad \text{Eq. (2)}$$

$$\bar{X} = x - x' \quad \text{Eq. (2)}$$

Here $(x - x')$ is the effective length of the boundary layer and x' is the origin of the layer. Typically the point of maximum velocity.

For turbulent boundary layers, Cebeci-Smith modified Stratford's original formulation to say that the boundary layer is separated where:

$$0.4(10^{-6}Re)^{0.1} = \bar{C}_p(\bar{X}\frac{d\bar{C}_p}{dx})^{\frac{1}{2}}$$

Surface Pressure on Separated Regions

Swafford velocity profile is used to estimate the pressures in the separated flow field.

Further Reading

1. [The Prediction of Separation of the Turbulent Boundary Layer]
2. [Calculation of Separation Points in Incompressible Turbulent Flows]

Last updated: Jan 10, 2024

On this page Overview

Unstructured Surface-Vorticity

Unstructured Vorticity on an Arbitrary Body

Unstructured surface vorticity solvers are an extension of structued thin-surface vorticity based solvers (e.g. vortex lattice methods). They depend on the evaluation of the circulation around a given geometry and use the Kutta–Joukowski formulation, which is applied to each panel for evaluation of the induced loads. In FlightStream, these methods are extended to work on arbitrary bodies defined by an unstructured surface mesh.

The derivation for the velocity induced by a linear segment of a face can be mathematically developed in terms of the coordinates of the two end vertices and the point of interest using the Biot–Savart Law. Consider a point P in space where a segment of a vortex ring is inducing a velocity. The velocity induced at P by an infinitely small segment of the ring edge is known from the application of the Biot–Savart law:

$$dV_i = \frac{\Gamma}{4\pi}\frac{dl \times r}{r^3} \qquad \text{Eq. (1)}$$

In Eq. (1), all variables on the right-hand side are easily evaluated from vector formulations of the planar geometry defined by the three points in space (two vertices of the vortex segment and one point of interest). Note that Eq. (1) also reduces to the two-dimensional vortex for the limiting case of infinitely long vortex. Equation (1) is the induced velocity by one segment of a facet. Therefore, to evaluate the velocity induced by the entire ring, we evaluate Eq. (1) for all three edges of the vortex ring to get the net velocity induction from that face as:

$$V_j = \Gamma_j \sum_{i=1}^{N_{edges}} V_i = \Gamma_j A_{j,p} \qquad \text{Eq. (2)}$$

The coefficient $A_{j,p}$ is the geometrical influence coefficient for the velocity induction at point P by the face j of the surface mesh.

Considering Eq. (2) the only unknown is the vorticity strength. This creates a system of equations with N unknowns that must be solved for Γ . The matrix must be evaluated in patches numerically during the inversion phase or the solver must be handed small patches of the mesh. In FlightStream, the mesh is split into solver partitions solved individually. Freezing remaining mesh sections at the previous iteration values. This approach creates an iterative solver. Given the nature of the geometry, an exact solution is not practical and instead the solver completes once the surface vorticity residual reaches some specified tolerance.

Convergence

The convergence criterion used is based on the stability of the surface vorticity 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 face vorticity. The wake instability is quantified as the rms change of the two-dimensional strand positions within the Trefftz plane.

Wake Treatment

On this page > Overview

Once the trailing edges are marked, the vertex pool corresponding to these edges is available from the global vertex map. These vertices become the starting point, or nodes, for the individual wake strands. Since no additional vorticity sources or sinks are added by the wake, the net vorticity of the geometry remains conserved and the wake propagates only the required vorticity to satisfy the Kutta condition at the trailing edges.

Fig. (1): The directional vorticity being shed into the wake strand at the trailing edge on a) a general node and b) a corner node.

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 vortex model obtained from the exact solution of the Navier–Stokes equations for a laminar 2D vortex. Knowing the initial vorticity shed into the strand at the trailing-edge node, the Lamb–Oseen model provides the vorticity decay equation as:

$$\Gamma_t = \Gamma(1 - e^{-\rho_\infty r^2/4\mu t}) \qquad \text{Eq. (3)}$$

In Eq. (3), the time used corresponds to the solver pseudotime and starts at zero at the trailing edge. It is marched alongside the strand (using values of local velocity and the discretization size of the relaxed wake, which is fixed to the average mesh edge length) until the termination of the wake downstream at the Trefftz plane. All strands are projected or terminated upon intersection with the Trefftz plane.

Further Reading

1. Aerodynamic Loads over Arbitrary Bodies by Method of Integrated Circulation

Last updated: Jan 10, 2024

← Previous
Why FlightStream?

Next →
Integrated Loads

On this page
Overview

Viscous Coupling

The core of FlightStream is a surface-vorticity potential-flow based solver. However, after the initial inviscid solution is calculated and surface streamline pressures and velocities are calculated, integral boundary layer methods can be used to estimate the boundary layer properties.

The methods used for boundary layer analysis closely follow the work in Ref. [1] by Fujiwara, Chaparro, and Nguyen. These routines have been shown to agree well with wind tunnel data and XFOIL for 2D airfoils.

Laminar Boundary Layer

The methods by Thwaites [1] are used to calculate the laminar boundary layer properties in compressible flow. Starting from von Karman's compressible integral momentum equation defined as:

$$\frac{d\theta}{ds} + \frac{1}{u_e}\frac{du_e}{ds}\theta(2 + H - Ma^2) = \frac{1}{2}C_f \qquad \text{Eq. (1)}$$

Where $H = \frac{\delta^*}{\theta}$ and u_e is the streamline edge velocity. Next, Thwaites defines the dimensionless pressure gradient:

$$\lambda = \frac{\theta^2 \rho}{\nu}\frac{du}{ds} \qquad \text{Eq. (2)}$$

And the parameter: l :

$$l = \frac{1}{2}Re_\theta C_f \qquad \text{Eq. (3)}$$

With these substitutions, Eq. 1 becomes:

$$\frac{\rho u_e \theta}{\mu}\frac{d\theta}{ds} + (2 + H - Ma^2)\lambda \qquad \text{Eq. (4)}$$

In ref. [1] it shown that the momentum thickness, θ , can be estimated numerically from:

$$\theta^2 u_e^6 = 0.45\frac{\nu}{\rho}\int_0^e u_e^5 ds \qquad \text{Eq. (5)}$$

Thwaites approximations for l and H are:

$$-0.1 < \lambda \leq 0 \begin{cases} l = 0.22 + 1.402\lambda + \frac{0.018max(\lambda,-0.1)}{\lambda+0.107} \\ H = 2.088 + \frac{0.0731}{max(\lambda,-0.1)+0.14} \end{cases} \qquad \text{Eq. (6)}$$

$$0 \leq \lambda \leq 0.1 \begin{cases} l = 0.22 + 1.57\lambda - 1.8min(\lambda, 0.1)^2 \\ H = 2.61 - 3.75\lambda + 5.24min(\lambda, 0.1)^2 \end{cases} \qquad \text{Eq. (7)}$$

Finally, δ^* and C_f can be calculated.

$$C_f = \frac{2l}{Re_\theta}\left(1 + \frac{\gamma-1}{2}Ma^2\right) \qquad \text{Eq. (8)}$$

Transition

valid for $1e5 > Re > 40e6$

Turbulent Boundary Layer

Head's method is based on the concept of entrainment. The volume rate within the boundary layer at location s is:

$$Q(s) = \int_0^{\delta(s)} u dy \qquad \text{Eq. (10)}$$

The entrainment velocity, E, is the rate at which Q increases with respect to s:

$$E = \frac{dQ}{ds} \qquad \text{Eq. (11)}$$

Combining the above equation with the definition of displacement thickness,

$$\delta^* = \delta - \frac{Q}{u_e} \qquad \text{Eq. (12)}$$

Then,

$$E = \frac{d}{ds} u_e(\delta - \delta^*) \qquad \text{Eq. (13)}$$

which can be rewritten as:

$$E = \frac{d}{ds}(u_e \theta H_1) \qquad \text{Eq. (14)}$$

Where

$$H_1 \equiv \frac{\delta - \delta^*}{\theta} \qquad \text{Eq. (15)}$$

Head assumed that the dimensionless entrainment velocity, E/u_e , is a function of H_1 only. And that H_1 in turn, is a function of $H \equiv \delta^*/\theta$ only. Cebeci and Bradshaw fit several sets of experimental data with the following equations:

$$\frac{1}{u_e}\frac{d}{ds}(u_e \theta H_1) = \frac{0.036}{(H_1 - 3.0)^0.6169} \qquad \text{Eq. (16)}$$

and

$$H_1 = \begin{cases} 3.3 + 0.8234(H - 1.1)^{-1.287}, H \leq 1.6 \\ 3.3 + 1.5501(H - 0.6778)^{3.064}, H > 1.6 \end{cases} \qquad \text{Eq. (17)}$$

Together with Von Karman integral momentum equations, the above two equations represent three equations on four unknowns θ , H , H_1 , and C_f . Two equations to march θ and H_1 downstream are now available, given initial conditions. Once θ and H_1 are advanced to the next station (a Runge-Kutta method is used), H is obtained from:

$$H = \begin{cases} 3.0, H_1 < 3.3 \\ 0.677 + 1.153(H_1 - 3.3)^{-0.326}, 3.3 < H_1 < 5.3 \\ 1.1 + 0.86(H_1 - 3.3)^{-0.777}, H_1 > 5.3 \end{cases} \qquad \text{Eq. (18)}$$

and using these values and Ludweig-Tillman equation, the skin friction coefficient C_f can be obtained, including compressibility effects:

Further Reading

1. Fujiwara, Chaparro, and Nguyen. An Integral Boundary Layer Direct Method Applied to 2D Transonic Small-Disturbance Equations
2. Thwaites, B. Approximate Calculation of the Laminar Boundary Layer
3. Etude de la Transition sur les Profiles dAile; Etablissement dun Critere de Determination de Point de Transition et Calcul de la Trainee de Profile Incompressible
4. Head, M. Entrainment in the Turbulent Boundary Layer
5. [Cebeci, Smith. Calculation of Separation Points in Incompressible Turbulent Flows] (https://doi.org/10.2514/3.59049)

Last updated: Jan 10, 2024

Previous
Separation Modeling

Next
Actuator Discs

Why FlightStream?

Industry Need

Current industry trends point toward an increasing use of optimization-based design principles in conceptual design. For complex systems, optimizer routines require hundreds or thousands of design evaluations. Since each of these designs must be tested for aerodynamic or fluid-dynamic performance, a balance between minimizing the computation time versus maintaining the fidelity of the solution and the validity of the physics must be made. This balancing represents a major limitation in the comprehensive use of multidisciplinary-optimization (MDO) design pipelines.

Navier-Stokes (e.g. RANS, LES) solvers are rapidly making inroads into the MDO environment, however this approach is restricted to simpler optimization cases. On the other hand, pressure-based potential flow solvers are highly susceptible to imperfections in surface mesh quaility and refinement. Finally, most vorticity based solvers are currently limited to structured thin surface geometries.

Why 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 face is bound by four vertices extracted from the rectangular structure of the underlying map (known as the U-V map). As such, any structured surface mesh is essentially quadrilateral in nature. Consequently, facets have to be “forced” to assume triangular shapes to meet geometrical demands. This is accomplished by vertex merging 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. In addition, forced U-V mappings ensure that regions of high curvature or concave warping are always populated with more facets than actually required. Furthermore, the quality of such facets is typically very poor. Finally, the typical quadrilateral facet has to be sanitized to establish its “effective” surface normal direction. With triangular faces, this is not required, given the deterministic nature of the triangle geometry in three-dimensional space. 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 face count. Despite the aforementioned shortcomings of the structured surface meshes, with potential flow solvers (and especially 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. Vorticity solvers require a face-edge pool to be split into bound and trailing vortices during the application of the Kutta–Joukowski formulation. This means that faces 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 bound and trailing vortices. The bound vortices are simply marked as those facet edges aligned along a certain numerical axis, and the other edges are marked as trailing. Note that a such marking usually only corresponds to the correct flowfield if the physical space warping and curvature are not significantly different from the alignment of the numerical axes. Furthermore, the classical induced load formulation requires knowledge of the vorticity in the downstream bound vortex. This reduces the computational burden on the solver and allows the evaluation of loads in a simple and straightforward manner.

Last updated: Jan 4, 2024

Next

Unstructured Surface-Vorticity →