AIAA Journal
Vol. 43, No. 3, March 2005

Computations of Wall Distances Based on Differential Equations

Paul G. Tucker*
University of Wales, Swansea, Wales SA2 8PP, United Kingdom
Chris L. Rumsey†
NASA Langley Research Center, Hampton, Virginia 23681
Philippe R. Spalart‡
Boeing Commercial Airplanes, Seattle, Washington 98124
and
Robert E. Bartels§ and Robert T. Biedron¶
NASA Langley Research Center, Hampton, Virginia 23681

The use of differential equations such as Eikonal, Hamilton–Jacobi, and Poisson for the economical calculation of the nearest (normal) wall distance *d*, which is needed by some turbulence models, is explored. Modifications that could palliate some turbulence-modeling anomalies are also discussed. Economy is of especial value for deforming/adaptive grid problems. For these, ideally, *d* is repeatedly computed. It is shown that the Eikonal and Hamilton–Jacobi equations can be easy to implement when written in implicit (or iterated) advection and advection-diffusion equation analogous forms, respectively. These, like the Poisson Laplacian term, are commonly occurring in computational-fluid-dynamics (CFD) solvers, allowing the reuse of efficient algorithms and code components. The use of the NASA CFL3D CFD program to solve the implicit Eikonal and Hamilton–Jacobi equations is explored. The reformulated *d* equations are easy to implement and are found to have robust convergence. For accurate Eikonal solutions, upwind metric differences are required. The Poisson approach is also found effective and easiest to implement. Hence this method is recommended. Modified distances are not found to affect global outputs such as lift and drag significantly, at least in common situations such as airfoil flows.

Nomenclature

C_{DES}	=	turbulence modeling constant
C_L, C_D	=	lift and drag coefficients
c	=	wing chord
d	=	nearest (normal) wall distance
$\tilde{d}$	=	distance function
d'	=	wall distance with no restriction on surface normality
i, j, k	=	array or grid point indices
K	=	dimensionless length scale function in Eq. (1)
L	=	modified Laplacian
M	=	metric term in transformed equations or Mach number
m_s, m_v	=	number of surface and internal node points
N	=	number of iterations
n	=	exponent, weighting/biasing parameter or outward facing normal
ns	=	marker in marching front procedure for seed, trial, or points to be later solved
R_u	=	residual for $\boldsymbol{U}$ equation
Re	=	Reynolds number based on wing chord
$\boldsymbol{U}$	=	front propagation velocity implied in Eikonal equation
u, v	=	Cartesian velocity components for Eikonal front propagation
x, y	=	Cartesian coordinates
y^+	=	distance to nearest surface in wall units
α	=	grid expansion or underrelaxation parameter
Γ, Γ^*	=	diffusion analogous coefficients in Eqs. (3), (6), and (7)
ε	=	constant in diffusion analogous coefficient evaluation [see Eq. (10)]
λ	=	viscosity solution coefficient
ξ, η	=	transformed coordinates
ϕ	=	dependent variable in differential wall distance or distance function equations
Ω	=	domain for wall distance or distance function computation
$\tilde{\Omega}$	=	reduced domain for normal wall distance or distance function computation
ω	=	solid angle

I. Introduction

THE normal wall distance d is still a key parameter in many turbulence modeling approaches[1,2] and also in peripheral applications incorporating additional solution physics.[3,4] Examples of the latter include explosive front, multiphase flow, and electrostatic particle force modeling. Also, their near-wall isovalues can be used in grid generation.[5,6] Importantly, far-field d contours can be used as a rapid means of evaluating computational interfaces on unstructured overset meshes having relative movements.[7]

Surprisingly, with search procedures the effort in calculating d can be significant. For example, even with Cray C90 class computers and time-invariant meshes it can take three hours just to gain d in a large three-dimensional case.[8] Because of the expense and interblock communication issues, in some codes the following dangerous approximations are made: 1) computing distances down grid lines, not allowing for grid nonorthogonality; 2) computing d as the distance between a field point and the nearest wall grid point, instead of truly the surface; and 3) in multiblock grids, determining d on a purely block wise basis, ignoring the possibility that the nearest wall distance might be associated with another block. The latter can create large inaccuracies and also nonsmooth,

Received 27 February 2004; presented as Paper 2004-2232 at the AIAA 34th Fluid Dynamics Conference, Portland, OR, 28 June–1 July 2004; revision received 27 September 2004; accepted for publication 1 October 2004. Copyright © 2004 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved. Copies of this paper may be made for personal or internal use, on condition that the copier pay the $10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 0001-1452/05 $10.00 in correspondence with the CCC.

*Professor of Engineering, School of Engineering, Civil and Computational Engineering Center, Singleton Park. Member AIAA.

†NASA Senior Researcher, Computational Modeling and Simulation Branch. Associate Fellow AIAA.

‡Senior Technical Fellow.

§NASA Senior Researcher, Aeroelasticity Branch. Senior Member AIAA.

¶NASA Senior Researcher, Aeroelasticity Branch.

unhelpful-to-convergence, d distributions. In relation to point 3, for overset grids the situation can arise where the same point in space has different equations depending on which block it is viewed from. In such a case, there is no reason why the solution should converge at all.

Clearly, inexact, nonsmooth or grid-blocking-dependent wall distances $\tilde{d}$ will mostly be unhelpful for both accuracy and convergence. However, the deliberate modification of d to some distance function $\tilde{d}$ can alleviate certain secondary turbulence model anomalies or extend modeling potential.[9] For example, $\tilde{d} \gg d$ can alleviate the excessive influence of sharp convex features in the geometry on the turbulence model.[1,4] For corners or bodies/surfaces in close proximity, the increased multiple surface turbulence damping effect (see Refs. 10 and 11) can be crudely modeled by setting $\tilde{d} < d$.

A. Requirements of an "Ultimate" Distance Function

Following Ref. 11, a preferable distance-function $\tilde{d}$ behavior is perhaps best captured in terms of the elemental solid angle $d\omega$. This is the solid angle subtended by a patch, on say a surface of radius of curvature R, a distance d from a field point. (Note that $d\omega$ equals the patch area divided by d^2.) Hence Ref. 11 proposed

$$\frac{1}{\tilde{d}} = \frac{1}{\pi} \int K \frac{d\omega}{d'} \tag{1}$$

(here we correct the factor of two error in the equation of Ref. 11) where d' is a continuous wall distance with no restriction on surface normality. Here, the dimensionless turbulence length scale function is $K = 1$, but for certain turbulence modeling requirements this might not be best.[11] For example in Ref. 11, as part of a pressure-strain correlation near-wall correction, it is a function (expressed as a turbulence length scale) of turbulence kinetic energy and the rate of dissipation of this. Turbulence models are calibrated with a single flat surface mostly in mind with $d \ll R$ or $\omega \sim 2\pi$. In that case, from Eq. (1), $\tilde{d} = d$. For several surfaces at about the same distance, the blocking effects are weakly additive, and so $\tilde{d} < d$ seems sensible. For instance, a reasonable effect, exactly given by Eq. (1), for two surfaces, is the following harmonic related mean $1/\tilde{d}^n = 1/d_1^n + 1/d_2^n$, where d_1 and d_2 are the normal distances to the two walls and the exponent $n = 1$. In Ref. 11, $n = 1$ is also suggested, but, by modifying Eq. (1), $n = 2$ or higher is also possible in the harmonic means. For a channel, helpfully, considering the harmonic mean, or a modified Eq. (1), with $n = 2$, $\omega = 4\pi$, and for a flap cove $\omega > 2\pi$, that is, extra turbulence destruction is naturally introduced.

The opposite situation is when the solid body is much smaller than its distance to the field point under consideration, or in other words the total solid angle it covers is much smaller than one. An example is a thin wire, which clearly has a much weaker damping effect than a large flat surface at the same distance. We then need to have $\tilde{d} \gg d$. This also helps remedy the excessive modeled turbulence destruction that can be found around extreme convex features.[4]

The preceding are all accuracy, or physics, considerations and rather preliminary. Numerically, d field smoothness and computing speed are always desirable. This is more the key focus of the current paper. In addition, the accurate integration of Eq. (1) with high-aspect-ratio grid cells near the wall is far from trivial, and the approach of directly solving Eq. (1) can be quite unattractive in practice for that simple reason.

B. Differential d Equations

Motivated by the expense of the d solution, Sethian[5] considered the following essentially hyperbolic Eikonal equation:

$$|\nabla\phi| = 1 + \lambda\nabla^2\phi \tag{2}$$

(note: $|\nabla\phi|^2 = \nabla\phi \cdot \nabla\phi$) seeking viscosity solutions where $\lambda \to 0$. The dependent variable in Eq. (2) ϕ models propagating front first arrival times. The right-hand side implies (away from shock-analogous features that will naturally arise when solving a hyperbolic equation) that the front has unit velocity, that is, there is some velocity field with $|\boldsymbol{U}| = 1$. This means first arrival times are equal to d. Fares and Schroder[1] essentially solve an Eikonal related equation for ϕ^{-1}. (The inverse distance avoids the $\phi = \infty$ boundary condition for an infinite domain.) To enhance modeling potential, the Laplacian has some control over $\tilde{d}$. The Eikonal equation with an explicit Laplacian, as follows, is called a Hamilton–Jacobi (HJ) equation:

$$|\nabla\phi| = 1 + \Gamma(\phi)\nabla^2\phi \tag{3}$$

Solving for ϕ^{-1} as in Ref. 1 could overcomplicate programming and requires an additional arbitrary length scale d_0 to avoid infinite values at the wall. Also, in Ref. 1 no observation of the need for upwind differenced metric terms (see later) is made. The paper does not mention that the inverse $\tilde{d}$ equation is connected to the Eikonal and HJ equations and hence is amenable to specialized solution approaches.

Spalding[4] approximately reconstructs d from solution of a more numerically benign Poisson equation ($\nabla^2\phi = -1$). The $d(\phi)$ reconstruction involves an auxiliary analytically derived equation. Because the Poisson d method is easy to implement in industrial codes, this approach is further studied. The Eikonal, unlike the Poisson approach, is challenging to code.[4,5] This is especially the case for unstructured grids. Hence its implementation in established industrial computational-fluid-dynamics (CFD) solvers represents a significant code developer time investment and hence cost. Therefore, here, extending the work in Ref. 4. and that of others, use of an Eikonal equation form, amenable to general geometry CFD code implementation, is explored. The form is reminiscent of the Euler/Navier–Stokes equations (the key equations modeled in CFD solvers). For the current work, as the base CFD solver, the NASA CFL3D[12] program is primarily used.

Operation counts show that differential approaches can be significantly faster than search procedures. For example, crude search procedures require $\mathcal{O}(m_v m_s)$ operations where m_s and m_v correspond to the number of surface and internal node points. An efficient Poisson equation solver typically scales as $\mathcal{O}(m_v \log m_v)$. With care and using a heap-sort algorithm the Eikonal equation can also be solved in $\mathcal{O}(m_v \log m_v)$ operations.[5] Hence it can be seen that differential-equation-based wall distance approaches are attractive. For fixed-mesh problems it is unusual for the search procedure to contribute significantly to the total solution cost. Nonetheless, for increasingly common nonstationary-mesh flow solutions the repeated search cost becomes highly significant. Then, differential approaches, which can make easy, safe use of previous d estimates, become especially attractive. The new d approaches presented here are best viewed in this moving-mesh context. Although the approaches to be presented have successfully been tested for distorting meshes, only stationary mesh results are given here. Less condensed work, more strongly orientated to moving meshes, detached eddy simulation (DES) and complex geometries including HJ solutions can be found in Ref. 13. Reference 4 also places emphasis on the compatibility of the Eikonal equation with DES. Stationary meshes (where the starting/initialized d field is very different from the actual) most strongly test robustness and convergence traits and so seem appropriate. The equations considered are now more fully outlined.

II. Implicit/Iterated d Equations

A. General Distance Equation Forms

In this paper, three different wall distance methods based on differential equations are used. These are implicit Eikonal, implicit HJ, and Poisson, defined next. The Eikonal [Eq. (2)] is an exact d equation. When defining the vector

$$\boldsymbol{U} = \nabla\phi \tag{4}$$

it can be rewritten in the following implicit (Eikonal$_\text{I}$) advection analogous form:

$$\boldsymbol{U} \cdot \nabla\phi = 1 \tag{5}$$

In the preceding, for convenience $\lambda\nabla^2\phi$, where $\lambda \to 0$, is ignored [prior to squaring Eq. (2)]. The vector $\boldsymbol{U}$ corresponds to the front propagation velocity implied in the Eikonal equation, where $|\boldsymbol{U}| = 1$.

With Eq. (4), an HJ-type equation [essentially the Laplacian smoothing term has a different overall power than when used in Eqs. (2) and (3)] can also be written in the following implicit (HJ$_\mathrm{I}$) form:

$$\boldsymbol{U} \cdot \nabla\phi = 1 + \Gamma(\phi)\nabla^2\phi \tag{6}$$

The positive function $\Gamma(\phi)$ is discussed later. Because $(\nabla\phi)^2 = \nabla \cdot (\phi\nabla\phi) - \phi\nabla^2\phi$ and $\boldsymbol{U} = \nabla\phi$, the following conservative form of Eq. (6) is also possible:

$$\nabla \cdot (\phi \boldsymbol{U}) = 1 + \Gamma^*\nabla^2\phi \tag{7}$$

where

$$\Gamma^* = (\Gamma + \phi) \tag{8}$$

To gain Eikonal$_\mathrm{I}$ solutions, using an initial ϕ field guess, Eqs. (4) and (5) can be solved in an iterative sequence. HJ$_\mathrm{I}$ solutions can also be made using the same approach.

Along different lines, substitution of the Poisson-based ϕ distribution arising from solving Eq. (6) with $\boldsymbol{U} = 0$ and $\Gamma = 1$ into

$$d = -\sqrt{\sum_{j=1,3}\left(\frac{\partial\phi}{\partial x_j}\right)^2} \pm \sqrt{\sum_{j=1,3}\left(\frac{\partial\phi}{\partial x_j}\right)^2 + 2\phi} \tag{9}$$

also gives distances.[4] This is called the Poisson method. The derivation of Eq. (9) assumes extensive (infinite) coordinates in the nonnormal wall directions. Hence, unlike the Eikonal, d from Eq. (9) is accurate only close to walls. However, turbulence models only need d accurate close to walls.

B. Laplacian Form and Role

Near a fine convex feature (for example, a wire) for theoretical correctness, accurate distances are needed, and so $\tilde{d} = d$. However, to prevent excessive far-field influence $\tilde{d} \gg d$ can be required further away from the feature.[1,4] Adjacent to a convex feature $\Gamma\nabla^2\phi \gg 0$. Therefore, the positive Laplacian inclusion in Eq. (6) has the desired effect of enlarging/exaggerating $d(=\phi)$. Motivated by dimensional homogeneity, the need that as $d \to 0$, $\tilde{d} = d$, but $\nabla^2\phi \to \infty$ suggests

$$\Gamma = \varepsilon d \tag{10}$$

Clearly more "aggressive" functions than Eq. (10) [e.g., $\Gamma = \varepsilon(-1 + e^d)$] are possible, but these are not explored. It should be noted that ε and the functional relationship will be problem/flow dependent.

At concave corners $\nabla^2\phi < 0$; hence, $\tilde{d} \ll d$. Therefore, with the Laplacian the damping effects of extra walls, discussed earlier, is naturally accounted for. The $\nabla^2\phi$ in Eq. (6), when diffusive, has the potential to enhance convergence. (Diffusive equations and numerical schemes are associated with robust convergence.)

III. Numerical Modeling

For brevity, Eq. (4–7) numerical modeling is described. Modeling for the much less challenging Poisson d method is discussed in Ref. 4. Equations (4–7) can be solved on curvilinear and unstructured grids. For curvilinear grids they must be transformed using the chain rule for differential calculus in, say, an ξ, η system. When solving in this system, metric terms M such as $\xi_x (= \partial\xi/\partial x)$, η_x, ξ_y, and η_y are evaluated. This aspect will be discussed later. Equations (6) and (7) diffusive terms are discretized using second-order central differences. The advection analogous derivatives in Eqs. (5–7) are typically discretized, just considering an x-coordinate direction, for example, using the following first-order upwind type of approximation:

$$\frac{\partial\phi}{\partial x} \approx n_{i-1}\Delta_{i-1,j,k}\phi + n_{i+1}\Delta_{i+1,j,k}\phi \tag{11}$$

where

$$\Delta_{i-1,j,k}\phi = \frac{\phi_i - \phi_{i-1}}{\Delta x_{i-1}}, \qquad \Delta_{i+1,j,k}\phi = \frac{\phi_{i+1} - \phi_i}{\Delta x_{i+1}} \tag{12}$$

and $n_{i-1} = 1$ for $u_{i-1} > -u_{i+1}$ and $u_{i-1} > 0$ else $n_{i-1} = 0$; and $n_{i+1} = 1$ for $-u_{i+1} > u_{i-1}$ and $-u_{i+1} > 0$ else $n_{i+1} = 0$. The propagation velocities in the preceding are evaluated from Eq. (4) where, for example,

$$u_{i-1} = \frac{\phi_i - \phi_{i-1}}{\Delta x_{i-1}}, \qquad u_{i+1} = \frac{\phi_{i+1} - \phi_i}{\Delta x_{i+1}} \tag{13}$$

Unlike the fluid velocity and turbulence variables, which have extreme near-wall variations, the distance ϕ field variation is quasilinear with relatively small gradients. Also, variations predominantly occur along the surface orthogonal grid lines. Hence, the results here and those in Refs. 3–5 and 13 show that the standard Eikonal solution practice, as used by numerous other Eikonal workers, of utilizing first-order upwinding is adequate.

Setting $\Gamma = 0$ in Eqs. (7) and (8) shows that the conservation form of the Eikonal$_\mathrm{I}$ equation still has a Laplacian. Here the view is taken that the discretized conservation form of the equation is being solved. However, to allow optimal Eikonal$_\mathrm{I}$ solution (using a hyperbolic advancing front approach described later) the extra Laplacian $\phi\nabla^2\phi$ is ignored. The approximation of neglecting $\phi\nabla^2\phi$ is equivalent to solving the nonconservation equation (5) with the alternative approximation that

$$u_i = n_{i-1}u_{i-1} + n_{i+1}u_{i+1} \tag{14}$$

The preceding equation corresponds to using offset difference-based velocity components. These correspond to those needed in the conservative equation (7) form. Either viewpoint has no great accuracy implications. Where d needs to be accurate, for turbulence models $\phi\nabla^2\phi \to 0$. Numerical tests confirm the validity of the preceding arguments.

A. Metric Discretization

The metric terms M must be carefully discretized. In CFL3D, for the i index direction M_{i-1} and M_{i+1} are first evaluated. The latter, for example, involves geometric data at the i and $i+1$ grid points. These M_{i-1} and M_{i+1} difference-based constructs are then, as with most CFD codes, averaged to give a single M_i value to be used in the discretized transport equations. However, for strongly expanding near-wall grid spacings this standard approach results in d overestimations for the Eikonal$_\mathrm{I}$ and HJ$_\mathrm{I}$. To remedy this, the following offset (or upwinded) metric formulation is used:

$$M_i = n_{i-1}M_{i-1} + n_{i+1}M_{i+1} \tag{15}$$

where n_{i-1} and n_{i+1} are as defined for Eq. (11). The standard centered CFL3D implementation corresponds to $n_{i-1} = n_{i+1} = \frac{1}{2}$ in the preceding.

B. Stabilizing Measures

To ensure stable solutions for the Eikonal$_\mathrm{I}$ and HJ$_\mathrm{I}$, velocity clipping and diagonal dominance enhancement are tried. Both use the observation that, in two-dimensions, the exact $\boldsymbol{U}$ field should satisfy

$$R_u = \left|u_{i,j}^2 + v_{i,j}^2 - 1\right| = 0 \tag{16}$$

where u and v are x and y direction velocity components, respectively. Therefore, to improve diagonal dominance R_u and $R_u\phi_i$ are added to both the discretized equation matrix diagonals and source terms, respectively. Based on Eq. (16), the following velocity clipping is used:

$$|u_{i,j}| \leq 1, \qquad |v_{i,j}| \leq 1 \tag{17}$$

The Eikonal equation does not permit a backward-front movement.[5] The implication of this is that if in Eq. (6), for example, $\Gamma\nabla^2\phi < -1$ a theoretical violation has occurred. [The sign of the Eq. (6) right-hand side gives the front propagation direction.] Hence, for HJ$_\mathrm{I}$ solutions the equation (6) Laplacian L is modified to

$$L = \max[-C, \Gamma\nabla^2\phi] \tag{18}$$

where $0 \leq C \leq 1$. Here $C = 1$ is used. This is the theoretical correctness upper limit. Because antidiffusion is associated with instability, use of Eq. (18) should be viewed as a stability measure. As a further stability measure, underrelaxation is used either through the following $\phi = (1-\alpha)\phi^{old} + \alpha\phi^{new}$ (where $\alpha \approx 0.5$ is an underrelaxation factor and the new and old superscripts indicate iterative states) or use of a pseudotime term.

When solving the Eikonal equation explicitly, for each grid point evaluation of the discretized matrix coefficients needs around 160 operations. The Eikonal$_{\mathrm{I}}$ equation needs 90 operations, but then iteration is also required [i.e., Eqs. (4) and (5) must be contained in some iterative loop]. Therefore, the key efficiency issue is the number of iterations N typically required.

C. Simultaneous Equation Solution

For Eikonal$_{\mathrm{I}}$ solutions either essentially crude global Gauss–Siedel (GGS)-type iterations or a marching-front (MF) approach are used. For HJ$_{\mathrm{I}}$ solutions the MF approach can be used to gain an initial d field and then GGS used. Alternatively, as for the Eikonal$_{\mathrm{I}}$, purely GGS-based solutions can be made. To solve the Poisson equation, essentially a GGS approach is used.

The MF approach essence is illustrated using Fig. 1. In this, a "wire" is represented on a Cartesian grid using a single node point. Points surrounding this are considered to be either seed, trial, or those to be later solved. These points are labeled through the marker variable ns with the respective value 2, 1, or 0. To start, a seed (or multiple seeds) point is defined. The Eikonal$_{\mathrm{I}}$ equations are applied N times (until convergence) to assign this seed point a d value. The seed is represented in frame (I) by the closed symbol. The Eikonal$_{\mathrm{I}}$ equations are then solved for all immediate seed neighbors (trial points). These $ns = 1$ points are shown as open symbols. As shown in frame (II), the trial point with the minimum d is taken as the next seed. Now, for this point $ns = 2$. Then, as shown in frame (III), d for all immediate neighbors to the new seed (for which $ns \neq 2$) are calculated. Hence the next seed point location is found. Frames (IV) and (V) show subsequent development stages for a clockwise moving circular front. In summary the procedure is as follows: 1) define a seed point, 2) iteratively solve for d at immediate neighbors to the seed (trial points), and 3) make the trial point with the minimum d the next seed and return to 1.

This procedure is continued until $ns = 2$ for all points in Ω or a smaller domain/extent $\tilde{\Omega}$. The latter domain is identified by the Fig. 1 dotted line. The Eikonal/MF method's compatibility with smaller domains is useful. This is because many turbulence models and DES[2] only need d to a maximum of about one-third of the boundary-layer thickness. In a practical system, all surface adjacent points can be taken as seeds and what is called an "active front" produced. For best efficiency a heap-sort procedure is required. This has not been incorporated as part of this work. Cleary, based on operation counts, $N < 2$ is needed for the Eikonal$_{\mathrm{I}}$ approach to match the Eikonal's efficiency. However, a typical Eikonal$_{\mathrm{I}}$ N value is 4 (see Ref. 13), that is, for stationary meshes the Eikonal$_{\mathrm{I}}$ equation has about twice the computational cost of the Eikonal. Hence for the Eikonal$_{\mathrm{I}}$ with stationary meshes the approximate operation count is $\mathcal{O}(2m_v \log m_v)$. For moving meshes this count greatly decreases.

D. Initial/Starting Conditions

For the Eikonal$_{\mathrm{I}}$ and HJ$_{\mathrm{I}}$ $1 < \phi < \infty$ in Ω is found adequate ($\phi > 0$ is essential). However, for MF solutions $\phi \to \infty$ is used. Note, when using the MF approach to give a DES distance field $\phi = C_{\mathrm{DES}}\Delta_i$

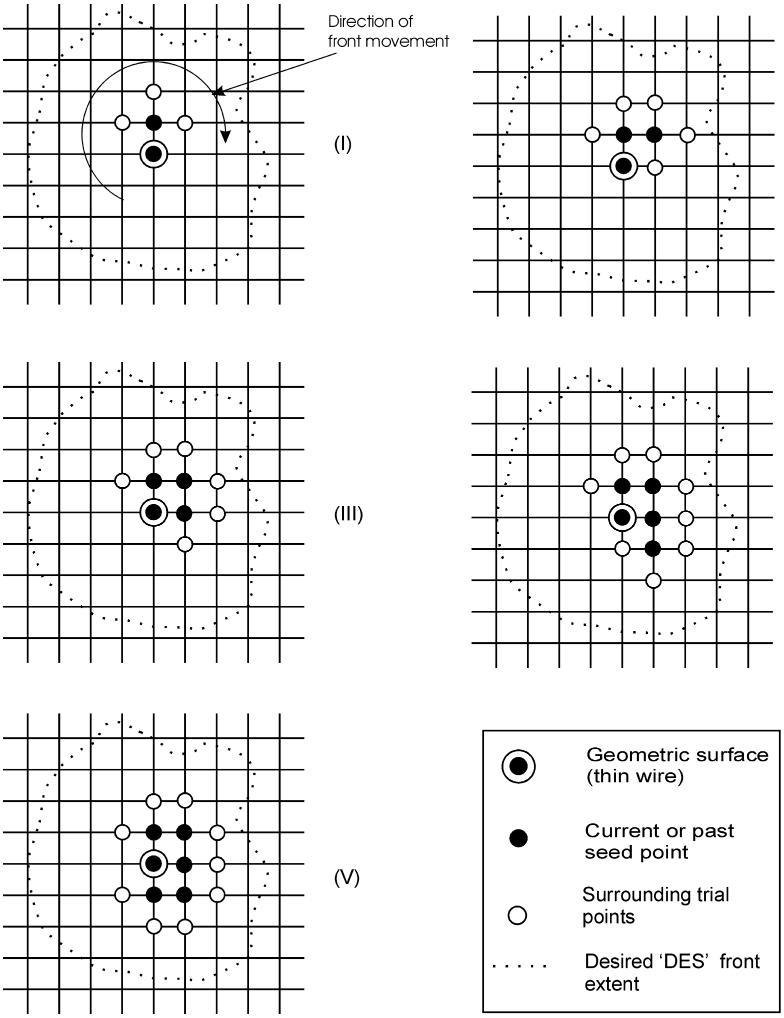
Direction of front movement
(I)
(II)
(III)
(IV)
(V)
◉ Geometric surface (thin wire)
● Current or past seed point
○ Surrounding trial points
. . . . . Desired 'DES' front extent

Fig. 1 MF solution approach with one seed.

can be used, where $\Delta_i = \max(\Delta x, \Delta y, \Delta z)$. The d computation will naturally stop when $d = C_{\rm DES}\Delta_i$. No DES d results are shown here. An example can be found in Ref. 13. For the Poisson approach $\phi \approx 0$ is adequate, and unlike the Eikonal there is no theoretical restriction on ϕ for convergence.

E. Boundary Conditions

Conditions on the domain boundaries are now described. At solid walls the following Dirichlet condition is applied:

$$\phi = 0 \tag{19}$$

At flow/far-field boundaries

$$\frac{\partial \phi}{\partial n} = 0 \tag{20}$$

can be used, where n is the boundary normal coordinate. However, if Ω is sufficiently large, Eq. (19) makes a stable, most computationally economical far-field boundary condition. It is especially preferred for the Poisson method where it gives much faster convergence.

Here either monoblock or overset grid solutions are considered. The implementation of the latter, which gives rise to interior boundaries, is based on the approach of Benek et al.[14] Trilinear interpolation is used to interpolate data between the nonmatching grid points. The interpolation is made in the computational space.

IV. Results and Discussion

The following geometries are considered: a) flat plate, b) single-element airfoil (NACA4412), c) wing body, and d) wing flap. Where d deviation (error) values are given, these are defined using the following equation:

$$\text{Error} = 100[(d_{\rm NSS} - d)/d_{\rm NSS}] \tag{21}$$

where $d_{\rm NSS}$ is coming from nearest surface search (NSS). The d without the subscript in Eq. (21) is the distance field for which the error is being evaluated. Flow solutions use the Spalart and Allmaras[15] (SA) turbulence model. This probably has the most extreme d requirements of most standard turbulence models. It needs d accurate for around one-third of the boundary-layer thickness. Hence it is a good candidate for d modeling tests. Fluid flow cases b and c are standard, sensibly grid-independent CFL3D test cases.[12] Separate studies confirm the grid independence for case d.

A. Flat Plate (Case a)

Initially, for testing just the Eikonal$_{\rm I}$ method, a flat plate is used. This is at $y = 0$, in a two-dimensional square domain Ω, with sides of unit length. First a grid (not shown) with strong grid expansion in the y direction is considered. Table 1 gives % d errors for different approximate geometric grid expansions α_y with offset and centered metrics.

Clearly (Table 1), with centered differences and larger α_y values serious errors arise. This is not surprising. Roache[16] shows for accurate flow solutions $\alpha_y < 1.3$ is needed. The front propagation nature of the Eikonal$_{\rm I}$ (and Eikonal) makes errors additive. Furthermore, centered differences are inconsistent with a hyperbolic propagating front problem. Hence, all of the remaining Eikonal$_{\rm I}$ results use offset metrics.

Next, the effect of using a badly distorted grid is explored. Figure 2 shows the badly distorted curvilinear grid used in this study. This grid is intended to severely test robustness of the Eikonal$_{\rm I}$ method.

Table 1 Eikonal$_{\rm I}$ % d errors with offset and centered metrics

	% d error	
α_y	Offset metrics	Centered metrics
1.1	0.48	1.53
1.7	0.99	8.25
2.6	3.15	15.0

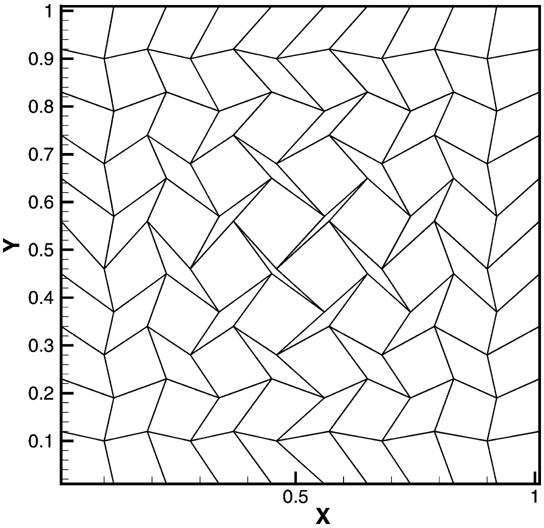

Fig. 2 Badly distorted flat-plate grid.

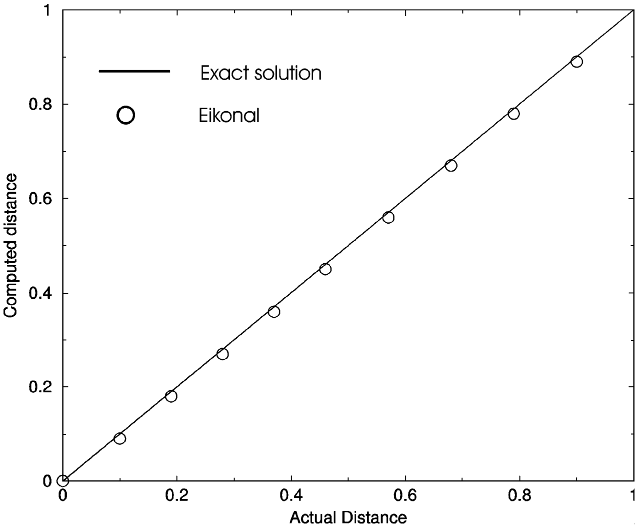

Fig. 3 Predicted against actual d at $x = 0.5$.

Figure 3 plots predicted against actual distances (y) at $x = 0.5$ for the Fig. 2 grid. The full line gives the exact solution. The symbols give the distances predicted by the Eikonal$_{\rm I}$ method. The figure shows that, even with an extremely distorted grid, reasonable distances can be obtained.

Figure 4 shows Fig. 2 grid velocity data. Frames a and b give $|\boldsymbol{U}|$ contours and $\boldsymbol{U}$ vectors, respectively. Considering the poor grid form, the velocity magnitudes and direction are surprisingly good. Figure 5 gives a Fig. 2 grid plot of log (R_u) against GGS iterations. Through its monotonic nature the plot reflects the stable convergence. The relatively high initialized Eikonal$_{\rm I}$ ϕ field results in the near solid boundary grid points, which should have the smallest ϕ values, supplying the highest residual components. However, the propagating front nature of the Eikonal solution means these high residual points are the first to be converged. Hence, the general trait of Eikonal$_{\rm I}$ solutions is for an initially high residual drop, as the high residual near-wall grid nodes are updated. Also, typically, for other convergence curves, not shown here, convergence is fairly monotonic with any nonmonotonicty occurring at low frequency. Frequency is used in an iteration number sense.

B. Single-Element Airfoil (Case b)

The Poisson, Eikonal$_{\rm I}$, and NSS procedures are now compared for a NACA4412 airfoil. The attack angle is 13.87 deg, $M = 0.2$, and $Re = 1.52 \times 10^6$ (based on the wing chord c). Figure 6 shows the

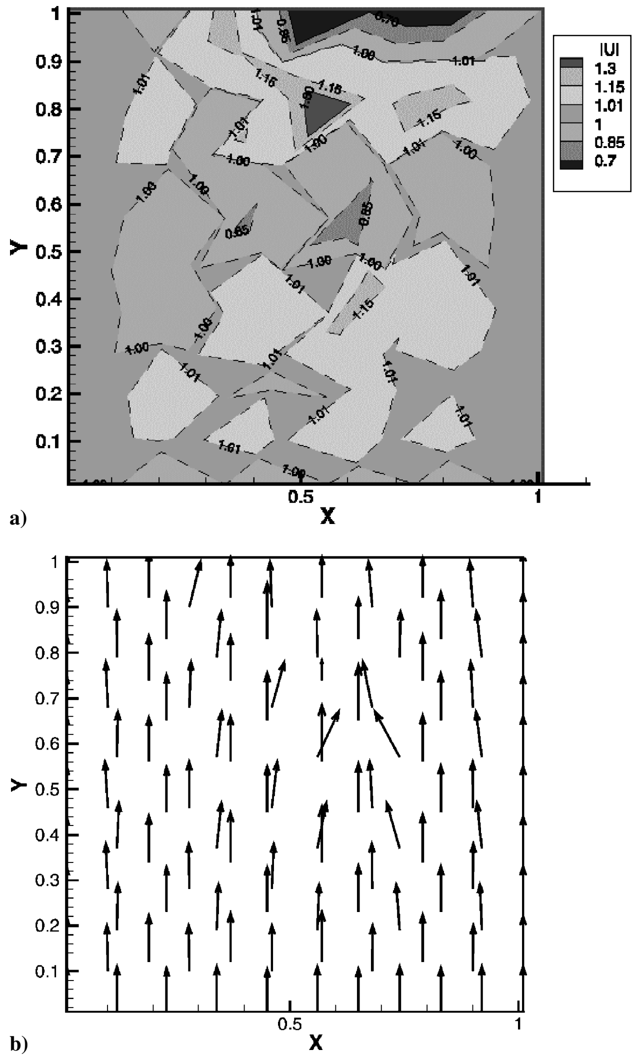

Fig. 4 Distorted grid $\nabla\phi$ data: a) $|U|$ and b) U vectors.

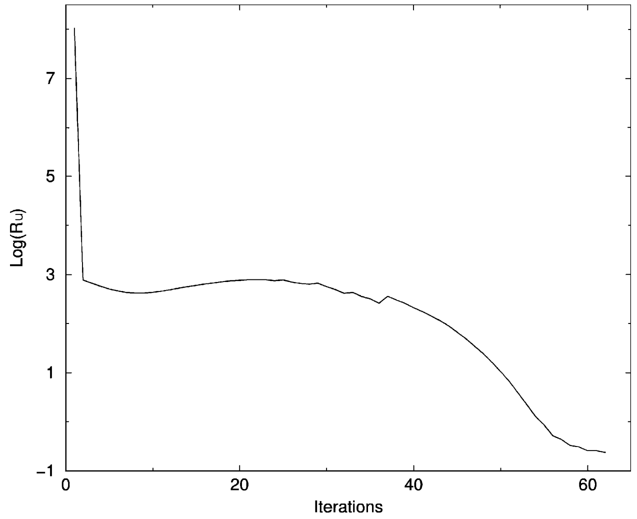

Fig. 5 Distorted grid plot of log (R_u) against GGS iterations.

three-zone, circa 25,000-cell overset grid used as part of the studies. Figure 7 gives $\text{Eikonal}_{\text{I}}$ $\boldsymbol{U}$ vectors. Correctly, the vectors are surface normal. Their magnitude is close to unity. Overset Poisson solutions are made for Ω and also just the inner zone region $\tilde{\Omega}$ labeled in Fig. 6. Obviously, the latter solution is most computationally efficient. At the $\tilde{\Omega}$ boundaries the Dirichlet condition $\phi = 0$ is used. Figure 8 shows d contours representing in the following respective frames: a) NSS, b) $\text{Eikonal}_{\text{I}}$, c) Poisson, and d) $\tilde{\Omega}$ Poisson results.

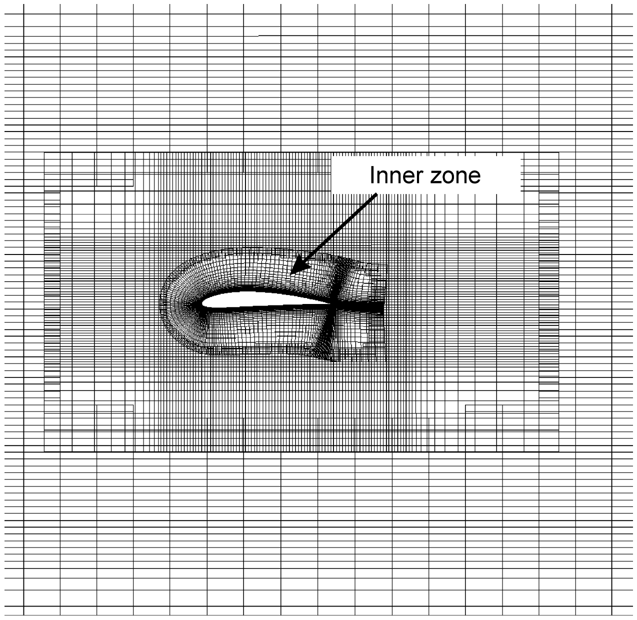

Fig. 6 NACA4412 overset solution grid.

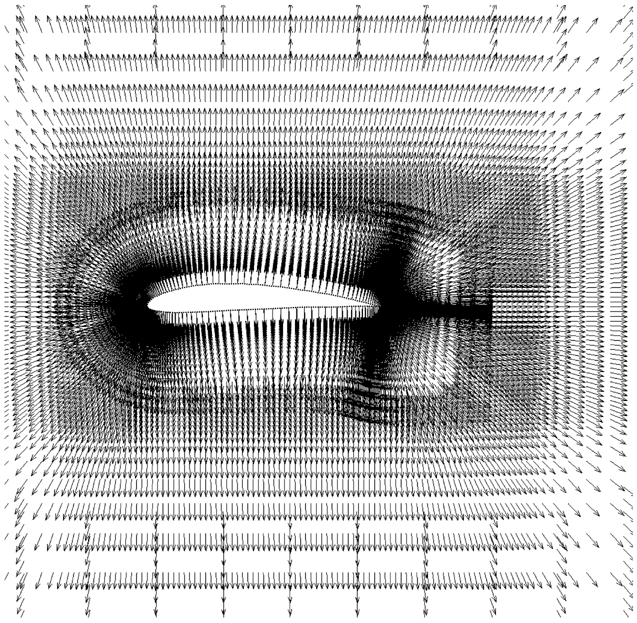

Fig. 7 $\text{Eikonal}_{\text{I}}$ U vectors.

Figure 9 gives $y^+ < 400$, d deviation histograms. Figure 9a is for the $\text{Eikonal}_{\text{I}}$ equation. The average deviation is 0.5%. Figure 9b is the overset Poisson $\tilde{\Omega}$ domain solution. Although not shown by the histograms, the average d deviation for the Poisson is significantly higher (2.5%) than that for the Eikonal equation.

Figure 10 gives a zoomed in view of trailing-edge region Poisson d deviation contours. As can be seen, a key d overestimation zone is the sharp convex trailing-edge geometry region. This overestimation zone, which will also be produced with the HJ_{I} equation, is potentially desirable.[4] Tests suggest that $\varepsilon \approx 1$ [see Eq. (10)] for the HJ equation to approximate the degree to which the Poisson equation overestimates d downstream of the trailing edge. This value is based on studies treating the trailing edge as point identified by just one node and also feeding the Poisson distance field for the full geometry into the HJ equation rearranged to solve for ε. It is, however, important to note that the d field exaggeration close to a singular point such as a wing trailing edge strongly varies with grid.

Table 2 gives lift C_L and drag C_D coefficients. Some minor differences are evident. The overset and nonoverset grid Poisson results are virtually identical and so not distinguished in the table.

Table 2 C_L and C_D for different d fields

Method	C_L	C_D
Eikonal$_\mathrm{I}$	1.704	0.03466
NSS	1.698	0.03496
Poisson	1.713	0.03543

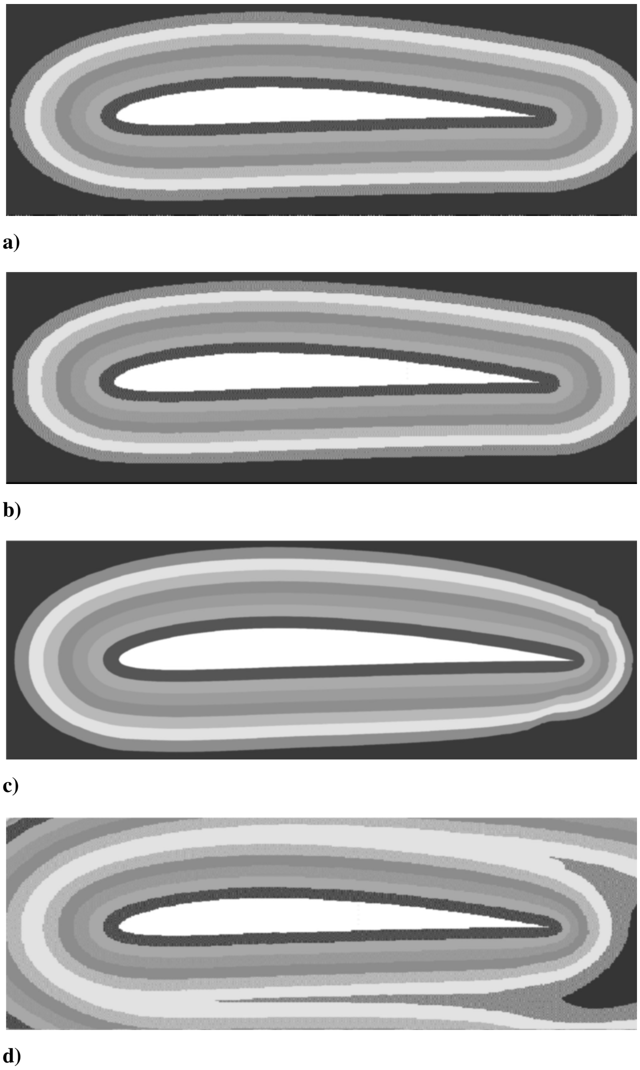

Fig. 8 Case b (NACA4412) d contours: a) NSS, b) Eikonal$_\mathrm{I}$, c) Poisson, and d) inner block Poisson.

For surface-pressure plots and boundary-layer velocity profiles, lines/results effectively overlay. Figure 11 gives SA, trailing-edge region, turbulent viscosity μ_t contours for the NSS (Fig. 11a) and Poisson (Fig. 11b) approaches. The larger $x/c > 1$ Poisson distances reduce the modeled turbulence destruction thus slightly increasing μ_t by just under 5%. Tests for flow over a thin wire suggest that turbulence levels for the exaggerated distances provided by the HJ and Poisson methods are more realistic than those provided by the NSS. A summary of these tests is given in the Appendix.

Clearly, for the current case turbulence intensities downstream of the airfoil have insignificant influence on the parameters of interest in a design context. However, for multiple-element airfoils it is not inconceivable that more significant solution differences could arise. For example, these can be caused by a peak in modeled turbulence energy convecting close to the center of the leading edge of a downstream element. Then small changes in the turbulence energy peak's position can give rise to different solutions.

The Poisson and NSS methods were also used as part of a single airfoil aeroelasticity analysis comparing with AGARD case 3 measurements.[17] The Poisson equation shows fast convergence and gives instantaneous C_L, C_D, and moment coefficient values very similar to NSS. (Stationary multiblock mesh studies, prior to complex geometry aeroelasticity studies, suggest at least 50% computing time savings. For the Eikonal$_\mathrm{I}$ and stationary meshes a modest 30% saving is found. However, these studies did not use a heap sort algorithm.[5]) These results are reported in Ref. 13.

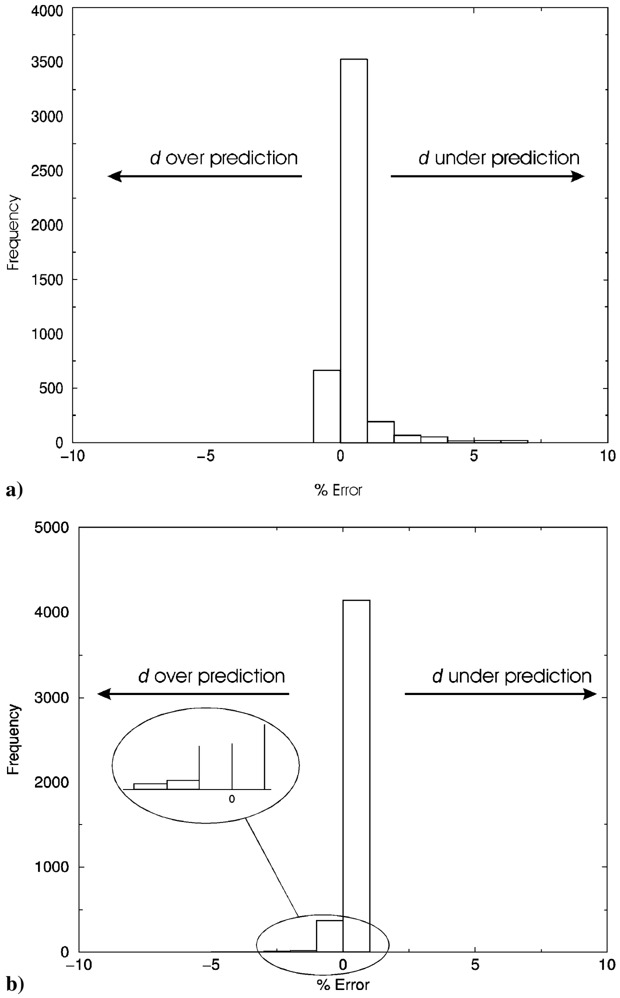

Fig. 9 Case b $y^+ < 400$ d deviation histogram: a) Eikonal$_\mathrm{I}$ and b) Poisson.

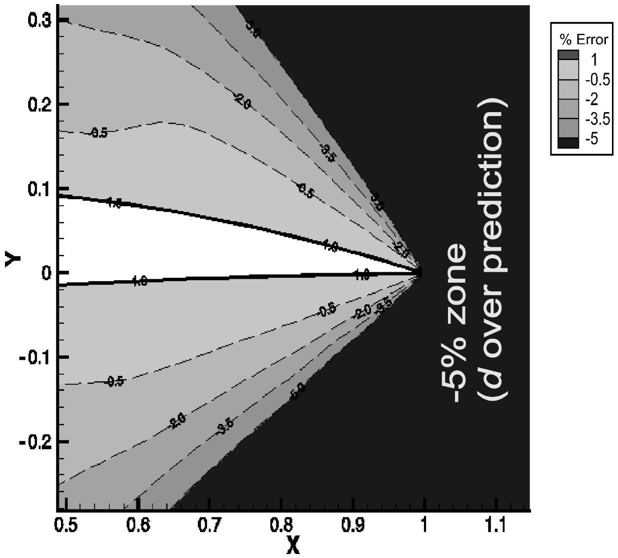

Fig. 10 Case b (NACA4412) Poisson d deviation contours.

C. Wing Body (Case c)

For this case the angle of attack is 2.87 deg, $M = 0.802$, and $Re = 13.1 \times 10^6$ (based on c). The single block grid, for which the body surface zone is shown in Fig. 12, has around 0.9 million cells. For this case the Poisson, Eikonal$_\mathrm{I}$, and NSS approaches are tested. Figure 13 gives the $y^+ < 400$-region d deviation histogram for the EikonalI$_\mathrm{I}$ and Poisson methods. The average errors are 3.13 and 3.14%, respectively. The Poisson's tendency to overpredict d is evident in the histograms. Table 3 gives C_L and C_D values for the different d fields. For this more complex geometry case, the average differences between C_L and C_D are lower than for case b.

D. Wing Flap (Case d)

For a multi-element airfoil case, the wing and flap angles are 2 and 40 deg, respectively. The wing-flap gap is 0.6% of c. The Reynolds number is 23×10^6, based on c, and $M = 0.18$. A small part of the highly stretched 10-block, approximately 0.9 million-cell grid is shown in Fig. 14a. For this case the Poisson, NSS, and Eikonal$_\mathrm{I}$ methods are tested.

Figure 14b gives Eikonal$_\mathrm{I}$ $\boldsymbol{U}$ vectors. Figure 15 gives a zoomed-in view of area A. Correctly, the vectors are surface normal. Magnitudes are close to unity. The vectors also illustrate interesting shock analogous features relating to the hyperbolic natured Eikonal equation. For example, in the wing cove region colliding vector fronts create a compression shock analogous zone. A similar zone is also created by collision of the fronts emanating from the upper wing and flap surfaces. In the immediate vicinity of the wing trailing edge, the diverging fronts from the upper and lower wing surfaces create a small expansion shock analogous zone. Figures 14c–14e give NSS, Eikonal$_\mathrm{I}$, and Poisson d contours, respectively. As can be seen, and would be expected, the NSS and Eikonal$_\mathrm{I}$ contours are virtually identical. As found in Ref. 4. the Poisson contours exhibit unsightly behavior around the lower portion of region A. In Ref. 4 this cosmetic issue (accurate d values are only needed close to surfaces) was attributed to both the significant level of block interface grid nonorthogonality and also the use of backwards differences in Eq. (9) at these. However, clearly the current results suggest the latter aspect is not an issue.

Figure 16 gives $y^+ < 400\,d$ deviation histograms. Figure 16a is for the Eikonal$_\mathrm{I}$. Figure 16b is for the Poisson. The average frame a

Table 3 Case c C_L and C_D for different d fields

Method	C_L	C_D
Eikonal$_\mathrm{I}$	0.6633	0.04839
NSS	0.6632	0.04855
Poisson	0.6635	0.04842

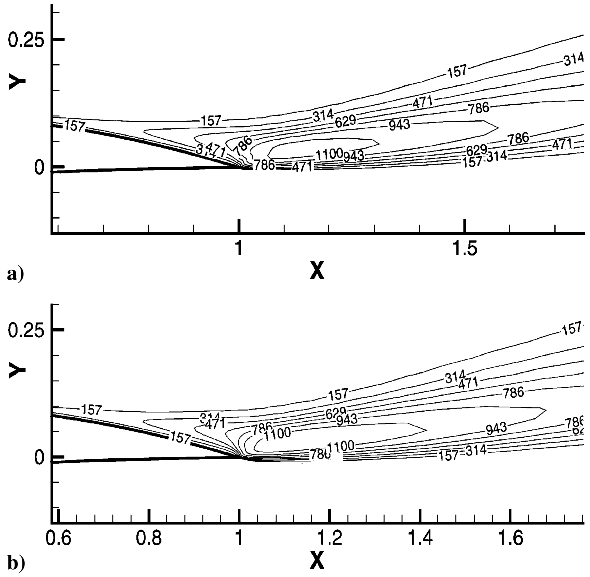

Fig. 11 Turbulent viscosity contours: a) NSS and b) Poisson.

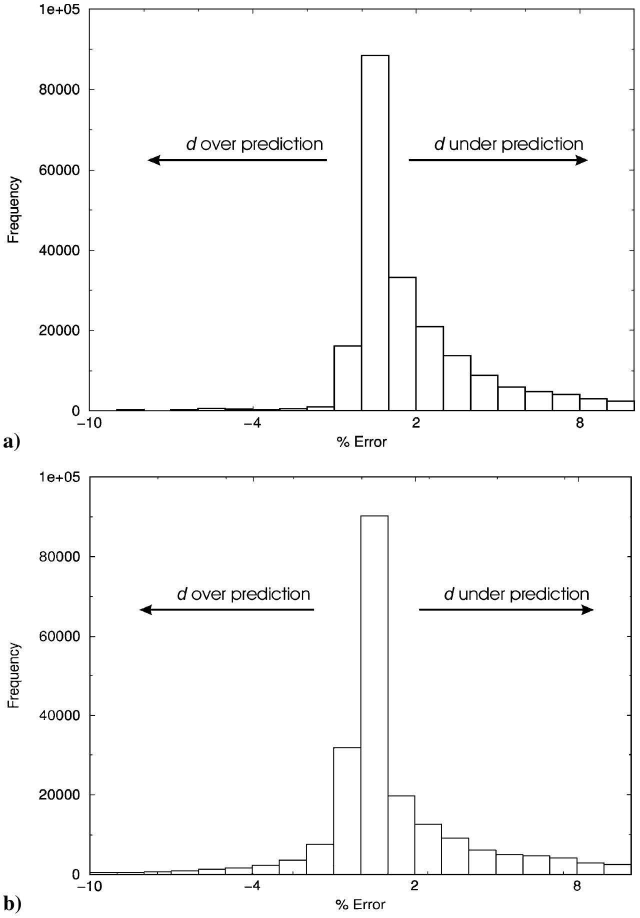

Fig. 13 Case c (wing body) $y^+ < 400\,d$ deviation histogram: a) Eikonal$_\mathrm{I}$ and b) Poisson.

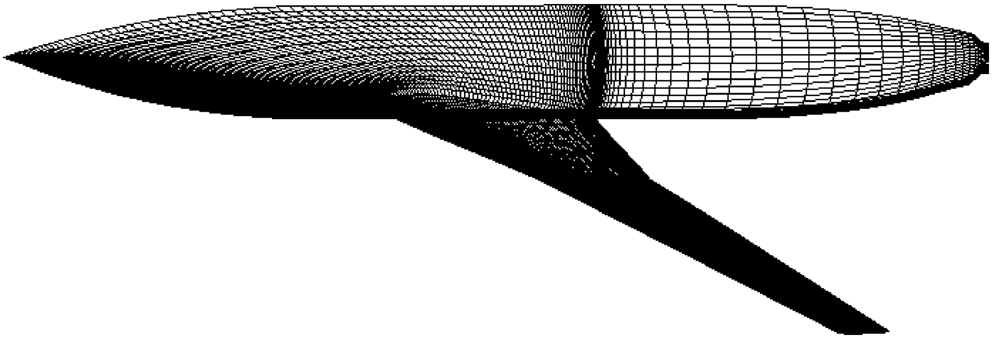

Fig. 12 Case c (wing-body) surface grid.

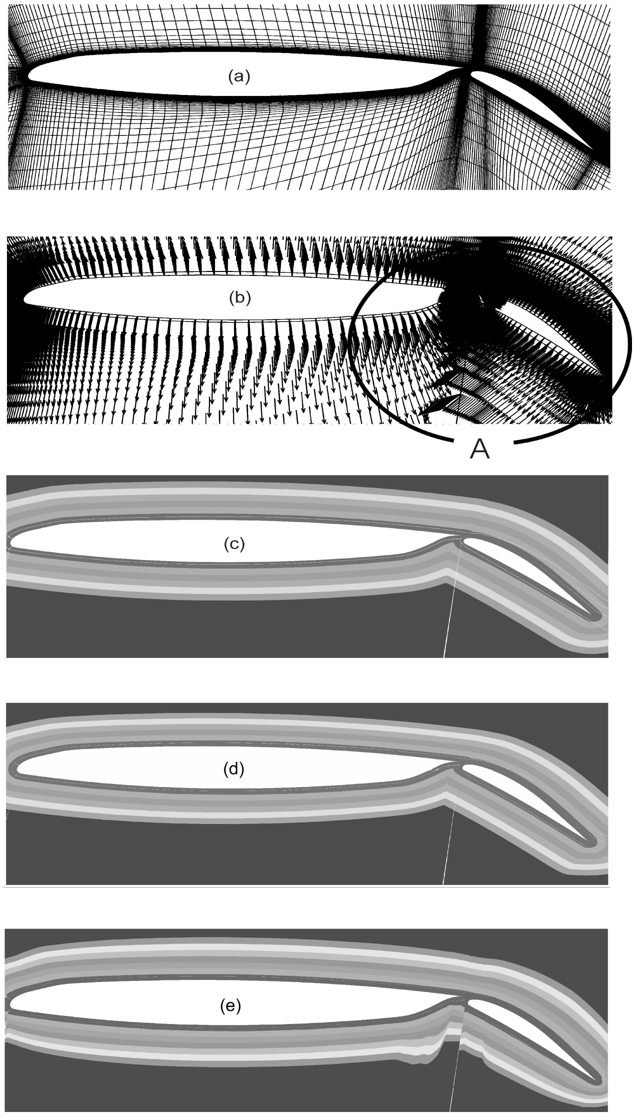

Fig. 14 Wing flap geometry near surface information: a) grid, b) *U* vectors, c) NSS *d* contours, d) Eikonal$_I$ *d* contours, and e) Poisson *d* contours.

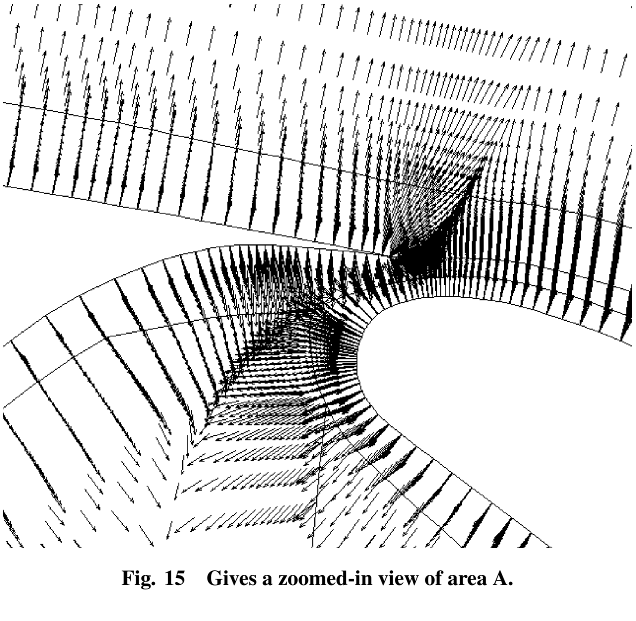

Fig. 15 Gives a zoomed-in view of area A.

Table 4 Case d C_L and C_D for different *d* fields

Method	C_L	C_D
Eikonal$_I$	2.815	0.0582
NSS	2.810	0.0584
Poisson	2.811	0.0584

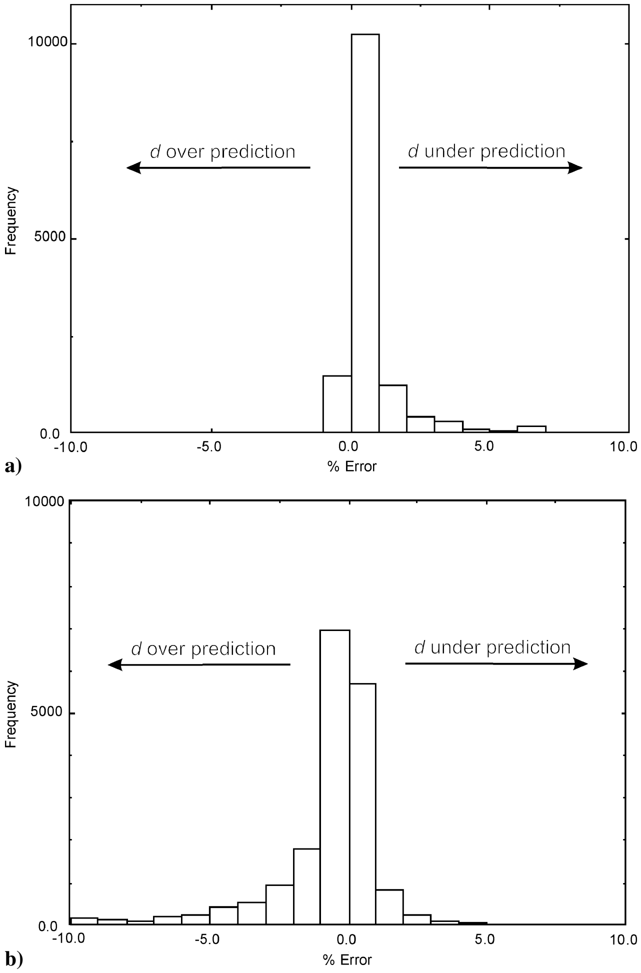

Fig. 16 Case d (wing flap) $y^+ < 400$ *d* deviation histogram: a) Eikonal$_I$ and b) Poisson.

deviation is 2.1%. The Poisson equation has a 1% deviation. This low Poisson error, relative to the Eikonal$_I$ equation, seems to contradict the histogram evidence. The contradiction is because the Eikonal equation has some large deviations at just a very few points. Clearly, the frame a Eikonal deviation distribution has the least spread. For the Poisson method (Fig. 16b), it can be seen there is a slight tendency to overpredict d. As can be inferred from Table 4, Poisson method lift and drag coefficients are within 0.05% of those for the NSS procedure. For the Eikonal$_I$ equation the deviation is a little greater but not that significant.

Pleasingly, complex geometry tests[13] show that HJ$_I$ GGS convergence rate is similar to the Eikonal$_I$. Hence the HJ$_I$ Laplacian does not seem that important for stability, its value being more in offering the capability of controlling wall distances near convex or concave surfaces.

Comparison of the results for cases d and c with those for case b show that with increasing geometric complexity C_L and C_D become less sensitive to wall distance traits. For case d this is most likely as a result of strong separation drag coming into play. However, for case c the explanation is less clear. Nonetheless this is a correct result and can also be observed for a low separation, more complex geometry, double-delta configuration result shown in Ref. 13.

V. Conclusions

Implicit forms of the Eikonal (Eikonal$_I$) and Hamilton-Jacobi (HJ$_I$) wall distance equations were presented. These are reminiscent of advection and advection-diffusion equations. Because of this, the Eikonal$_I$ and HJ$_I$ were found to be relatively easy to implement in an established industrial computational-fluid-dynamics solver. The implicit d equations were found to have robust convergence. Geometries studied included single- and two-element airfoils and a wing-body configuration. For HJ$_I$/Eikonal$_I$ accuracy, offset metric differences are required. A simple, approximate, Poisson-equation-based distance approach was also found effective. Because it did not require offset metric evaluations, it was easiest to implement. Also, because of its numerically benign nature the Poisson had comparable accuracy to the Eikonal. Hence this approach is recommended. The sensitivity of flow solutions to plausible modifications of d was explored. Especially with increasing geometrical complexity, results were not greatly affected by wall distance traits. This is partly because peculiar geometries such as thin wires were not involved. Downstream of the airfoil trailing edge, intensities were altered by just under 5%.

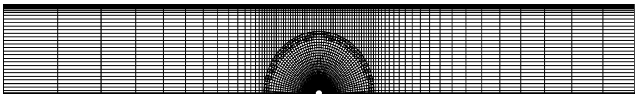

Fig. A1 Flow solution grid for thin-wire case.

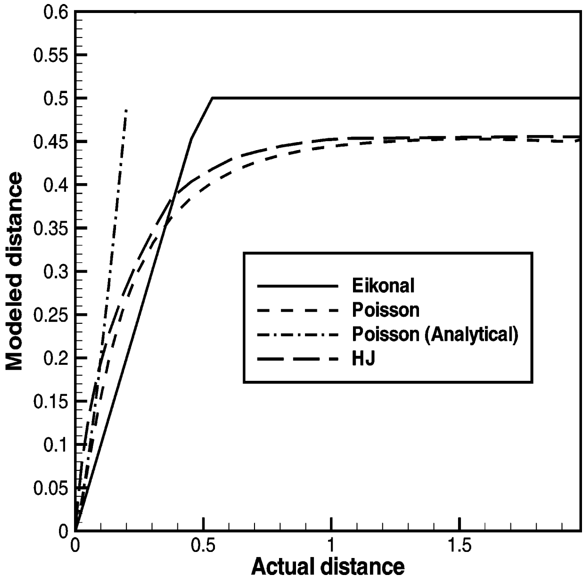

Fig. A2 Wall distance traits for thin-wire case.

Appendix: Flow over a Thin Wire

Figure A1 shows the dual-block overset solution grid used to compute the thin-wire fluid flowfield. Essentially the solution procedure and wall distance integration are as outlined in Ref. 4, which considers the HJ equation. The current case could be viewed as exploring the influence of a wire placed in a wind tunnel or channel. The wire diameter is 1/40 of the full channel width, but results are identical for a wire 1/400 of the width. At the left-hand inlet an $Re = 1 \times 10^5$ (based on the channel half-width) fully developed turbulent velocity profile is specified. Flow solutions take advantage of symmetry along the channel centerline. However, subsequent contour plots mirror solutions. In this way the results are clearer to observe.

Figure A2 plots computed wall distances against distance along the channel centerline. The x-coordinate origin is taken as the right-hand (downstream) side of the wire. The full and the long dashed lines are for Eikonal and HJ solutions, respectively. For the latter $\varepsilon = 8$ in Eq. (10). This relatively large ε value is chosen to make the HJ distances approximately match those for the Poisson equation, given by the short dashed line. The chain dashed line also gives Poisson distances, but these are analytical values for an infinite channel width. Hence, they approximate true external flow d values, which in the context of the current work seem most relevant. Clearly the Poisson equation distance overestimation trait is quite aggressive.

Figure A3 gives contours of wall distance and turbulence viscosity fields. Figure A3a(I) gives Eikonal distances and Fig. A3b(I) Poisson. Figure A3a(II) shows turbulence viscosity contours that would arise from solution of the Eikonal equation. (They in fact use NSS distances.) Figure A3b(II) gives turbulence viscosity contours for the Poisson equation. In an external flow context, near the wire

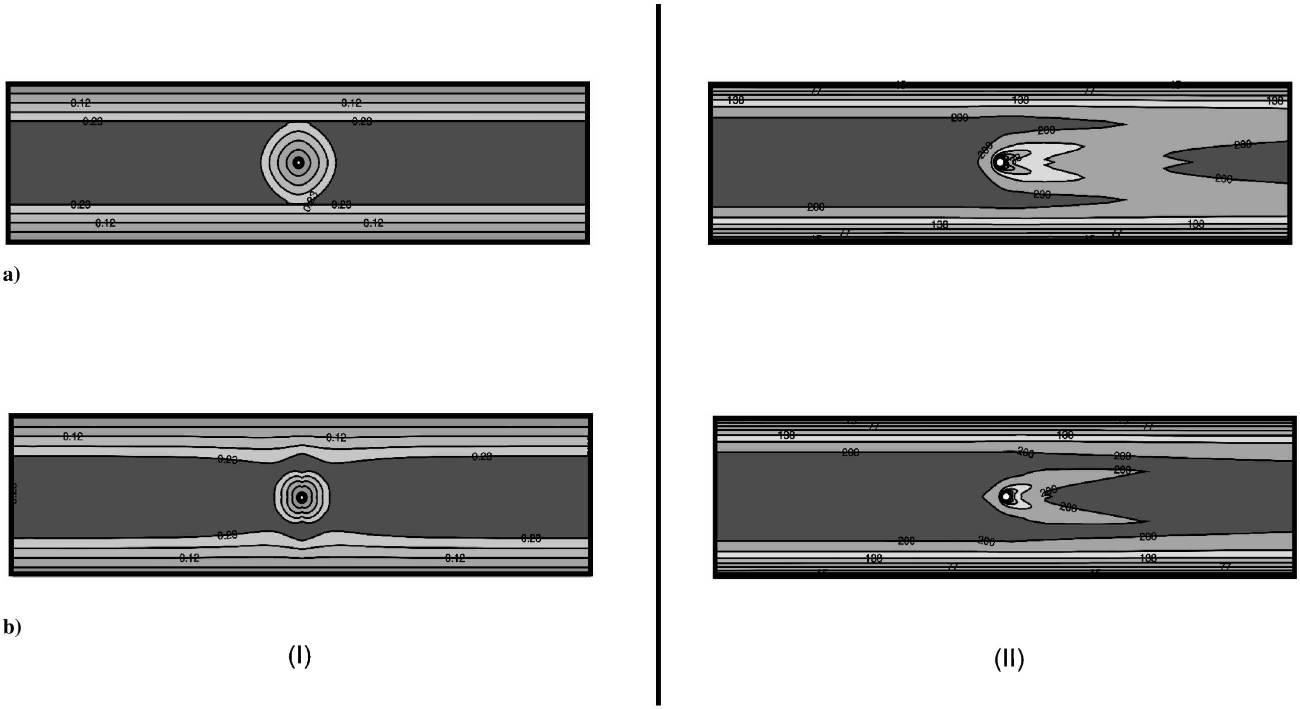

Fig. A3 Contours of variables for flow over a thin wire at $Re = 1 \times 10^5$: (I,a) Eikonal distances; (I,b) Poisson distances; (II,a) turbulence viscosity contours that would arise from use of the Eikonal equation; and (II,b) turbulence viscosity contours for the Poisson equation.

(i.e., for d less than around 17% of the channel half-width), the analytical Poisson distribution is used. However, when the numerical Poisson d field is used the turbulence viscosity contours are quite similar. As can be seen from Fig. A3a(II), in strong contradiction with physical reality the thin wire greatly disrupts the turbulence field. This field could be convected toward downstream components. The exaggerated distances arising from the Poisson (and also, as shown in Ref. 4, the HJ equation) significantly reduce the disruption caused by the wire.

Acknowledgments

This work was carried out at NASA Langley Research Center. The visit was funded by the United Kingdom Royal Academy of Engineering and Engineering Physical Sciences Research Council (Grant GR/S28402/01). We are very grateful for these awards.

References

[1] Fares, E., and Schroder, W., "A Differential Equation to Determine the Wall Distance," *International Journal for Numerical Methods in Fluids*, Vol. 39, No. 8, 2002, pp. 743–762.

[2] Strelets, M., "Detached Eddy Simulation of Massively Separated Flows," AIAA Paper 2001-0879, 2001.

[3] Osher, S., and Sethian, J. A., "Fronts Propagating with Curvature Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulation," *Journal of Computational Physics*, Vol. 79, No. 1, 1988, pp. 12–49.

[4] Tucker, P. G., "Differential Equation Based Wall Distance Computation for DES and RANS," *Journal of Computational Physics*, Vol. 190, No. 1, 1 Sept. 2003, pp. 229–248.

[5] Sethain, J. A., *Level Set Methods and Fast Marching Methods*, Cambridge Univ. Press, 1999.

[6] Barth, T. J., "Steiner Triangulation for Isotropic and Stretched Elements," AIAA Paper 95-0213, 1995.

[7] Nakahashi, K., and Togashi, F., "Unstructured Overset Grid Method for Flow Simulation of Complex Multiple Body Problems," International Council of the Aeronautical Sciences, Paper ICAS 0263, Aug.–Sept. 2000.

[8] Wigton, L. B., "Optimizing CFD Codes and Algorithms for Use on Cray Computers," *Frontiers of Computational Fluid Dynamics*, edited by D. A Caughey and M. M. Hafez, World Scientific, 1998, Chap. 16, pp. 1–15.

[9] Secundov, N., Strelets, M. Kh., and Travin, A. K., "Generalization of νt-92 Turbulence Model for Shear-Free and Stagnation Point Flows," *Journal of Fluids Engineering*, Vol. 123, March 2001, pp. 111–115.

[10] Mompean, G., Gavrilakis, S., Machiels, L., and Deville, M. O., "On Predicting the Turbulence-Induced Secondary Flows Using Non-Linear k-ε Models," *Physics of Fluids*, Vol. 8, No. 7, 1996, pp. 1856–1868.

[11] Launder, B. E., Reece, G. J., and Rodi, W., "Progress in the Development of a Reynolds-Stress Turbulence Closure," *Journal of Fluid Mechanics*, Vol. 68, Pt. 3, 1975, pp. 537–566.

[12] Krist, S. L., Biedron, R. T., and Rumsey, C. L., *CFL3D User's Manual*, Ver. 5.0, NASA TM-1998-208444, June 1998.

[13] Tucker, P. G., Rumsey, C. L., Bartels, R. E., and Biedron, R. T., "Transport Equation Based Wall Distance Computations Aimed at Flows with Time-Dependent Geometry," NASA TM-2003-212680, Dec. 2003.

[14] Benek, J. A., Steger, J. L., Dougherty, F. C., and Bunning, P. G., "Chimera: a Grid-Embedding Technique," Arnold Engineering Development Center, AEDC-TR-85-64, 1986.

[15] Spalart, P. R., and Allmaras, S. R., "A One-Equation Turbulence Model for Aerodynamic Flows," *La Recherche Aerospatiale*, Vol. 1, 1994, pp. 5–21.

[16] Roache, P. J., *Computational Fluid Dynamics*, Hermosa, Albuquerque, NM, 1976.

[17] Landon, R. H., "Compendium of Unsteady Aerodynamic Measurements, NACA 0012. Oscillatory and Transient Pitching," Data Set 3, AGARD-R-702, 1982.

P. Givi
Associate Editor