Improvements in Iterative Convergence of FUN3D Solutions

Li Wang *
NASA Langley Research Center, Hampton, VA, USA
Boris Diskin†
National Institute of Aerospace, Hampton, VA, USA
Eric J. Nielsen ‡
NASA Langley Research Center, Hampton, VA, USA
Yi Liu §
National Institute of Aerospace, Hampton, VA, USA

This paper presents a hierarchical adaptive nonlinear iteration method (HANIM) implemented in NASA computational fluid dynamics code, FUN3D, to improve robustness and computational efficiency of FUN3D solutions. In contrast to the baseline iterative solver that relies on an approximate Jacobian, a simple multicolor Gauss-Seidel point-implicit iteration scheme, and linear CFL ramping, HANIM is based upon a hierarchy of modules including preconditioner, generalized conjugate residual, realizability check, nonlinear control, and CFL adaption modules. HANIM performance is systematically compared with the performance of the baseline solver. The iterative solutions are compared for three aerodynamic benchmark cases: a subsonic separated flow around a hemisphere cylinder, a supersonic flow through a long duct, and a subsonic flow over the NASA wing-body juncture model. Two Reynolds-averaged Navier-Stokes turbulence models are used in these computations, namely, the negative variant of the linear one-equation Spalart-Allmaras model and its nonlinear extension based on quadratic constitutive relations.

I. Introduction

Computational fluid dynamics (CFD) plays a significant and growing role in analysis and design of aerospace vehicles. Advanced simulation capabilities enable reductions in wind-tunnel and in-flight tests, provide physical insight, and reduce cost and risks of development and incorporation of new aerospace technologies. The effectiveness and impact of CFD on the design and analysis of aerospace systems is largely driven by the ability to reliably conduct many accurate simulations in a relatively short time. This task is complicated by the multitude of complex flow phenomena encountered in practical aerodynamics, for example, boundary-layer separation, vortex shedding, compressibility effects, etc. Additional complications arise in the presence of increasingly irregular topology and highly stretched elements that are common in computational grids used for shock capturing or enhanced resolution in boundary layers. These challenges motivate the CFD community to focus on advancing various solver technologies that improve efficiency, robustness, and scalability of CFD solutions.[1]

This paper reports on the implementation of a strong iterative nonlinear solver in a practical CFD code, FUN3D, which is developed and supported by the NASA Langley Research Center (LaRC). FUN3D is a second-order, unstructured, node-centered, finite-volume CFD solver that is widely used for aerodynamic research and as a production

*Research Aerospace Engineer, AIAA Associate Fellow, li.wang@nasa.gov
†Sr. Research Fellow, AIAA Associate Fellow, bdiskin@nianet.org
‡Research Scientist, AIAA Associate Fellow, eric.j.nielsen@nasa.gov
§Sr. Research Engineer, AIAA member, yi.liu@nianet.org

tool within U.S. government agencies[2–10] and major aerospace companies.[11–13] The FUN3D development team has a long history of cutting-edge research in solver technology and high-performance computing for aerospace applications.[5,7,14–18] Recently, a novel hierarchical adaptive nonlinear iteration method (HANIM) has been implemented in FUN3D to improve efficiency and robustness for steady and unsteady Reynolds-averaged Navier-Stokes (RANS) simulations. In distinction from the legacy FUN3D solver that uses an approximate linearization and simple point-implicit multicolor Gauss-Seidel (GS) iterations, HANIM is based on a hierarchy of modules including preconditioner, generalized conjugate residual (GCR), realizability check, nonlinear control, and CFL adaption modules. The specific HANIM implementation reported in this paper follows Refs. 19–21. While FUN3D HANIM has been successfully applied to several unsteady problems, including rotorcraft simulations on dynamic overset grids, this paper focuses on steady RANS solutions. Two one-equation eddy-viscosity turbulence models are considered. The first turbulence model is the negative variant of the linear Spalart-Allmaras (SA-neg) turbulence model;[22,23] the other turbulence model is a nonlinear extension of the SA-neg model that uses quadratic constitutive relations, SA-neg-QCR2000.[24]

In this paper, HANIM performance is studied for three benchmark cases: (1) a subsonic separated flow around a hemisphere-cylinder configuration, (2) a supersonic flow through a long duct with a square cross-section, and (3) a subsonic flow over the NASA wing-body juncture configuration. The hemisphere-cylinder flow is simulated with the SA-neg turbulence model; the square-duct and wing-body juncture flows are simulated with the SA-neg-QCR2000 turbulence model. The hemisphere-cylinder and square-duct configurations are widely used verification cases posted in the section "Cases and Grids for Turbulence Model Numerical Analysis" of the NASA LaRC Turbulence Modeling Resource (TMR) website. [a] The square-duct and wing-body juncture cases are also SA-neg-QCR2000 verification cases selected for High-Fidelity CFD Workshop 2022. [b]

The rest of the paper is organized as follows. Section II describes key elements of FUN3D iterative solvers. Numerical results and discussions are provided in Section III. Section IV draws conclusions and discusses items that will be completed in the final paper.

II. FUN3D Iterative Solvers

A. Discretization Scheme

In the current paper, the FUN3D solver[25,26] is used to solve a nonlinear system of RANS equations discretized on unstructured mixed-element grids. The grids may contain tetrahedra, pyramids, prisms, and hexahedra. At median-dual control-volume faces, inviscid fluxes are computed by using an approximate Riemann solver. In the current study, Roe's flux difference splitting[27] is used. For second-order accuracy, density, pressure, and velocity at dual faces are obtained by a UMUSCL (Unstructured Monotonic Upstream-centered Scheme for Conservation Laws) scheme.[28,29] For mixed-element grids, the UMUSCL parameter is set as $\kappa = 0.5$ for the meanflow equations. The spatial discretization of the one-equation SA-neg turbulence model in FUN3D uses a first-order accurate convection scheme. The meanflow viscous fluxes are discretized by a finite-volume scheme, in which the Green-Gauss theorem is used to compute gradients at the dual faces. For tetrahedral meshes, this approach is equivalent to a Galerkin type approximation. For nontetrahedral meshes, cell-based Green-Gauss gradients are combined with edge-based gradients to improve stability of viscous operators. The diffusion term in the turbulence model is handled in the same fashion as the viscous terms in the meanflow equations. The vorticity-based source term for the turbulence model is computed by using velocity gradients evaluated by the Green-Gauss method on dual volumes.

The boundary conditions involved in the present study include farfield Riemann-invariant, farfield Roe-based, symmetry, and viscous-wall boundary conditions.[30] Both farfield Riemann-invariant and Roe-based boundary conditions are weakly enforced by constructing an exterior state based on the freestream conditions to close the boundary fluxes. The symmetry boundary condition strongly enforces zero normal velocity. The tangential momentum, mass, and energy conservation at the symmetry plane are enforced through the conservation equations. The no-slip viscous-wall boundary condition explicitly sets zero velocities on the wall boundary; density is computed through fluxes satisfying the continuity equation. Either isothermal or adiabatic wall boundary condition can be selected to close the energy equation. The isothermal boundary condition explicitly specifies a constant temperature at boundary nodes. The local

[a] `https://turbmodels.larc.nasa.gov`, accessed 06/01/2020

[b] `https://turbmodels.larc.nasa.gov/highfidelitycfd_workshop2022.html`, accessed 06/01/2020

pressure can be determined from the temperature and the density. The adiabatic condition is imposed either by setting an adiabatic-wall temperature derived from a flat-plate approximation or by specifying a true thermal insulation condition, $(dT/dn)_{wall} = 0$, through the numerical heat flux.

B. Legacy Iterative Solver

The discrete nonlinear equations describing conservation laws can be expressed as:

$$\mathbf{R}(\mathbf{Q}) = \mathbf{0}, \tag{1}$$

where $\mathbf{Q}$ denotes the flow solution and $\mathbf{R}$ denotes the nonlinear steady-state residual. This nonlinear system of equations is solved by a defect-correction method with a pseudotime enhancement expressed as:

$$\left[\frac{V}{\Delta\tau}\mathbf{I} + \widetilde{\frac{\partial \mathbf{R}}{\partial \mathbf{Q}}}\right]\Delta\mathbf{Q} = -\mathbf{R}(\mathbf{Q}^n), \tag{2}$$

$$\mathbf{Q}^{n+1} = \mathbf{Q}^n + \omega\,\Delta\mathbf{Q}. \tag{3}$$

underrelaxation
Here, V is the dual-cell volume; $\Delta\tau$ is the local pseudotime step; $\mathbf{I}$ is the identity matrix; the superscript n is the nonlinear iteration index, and ω is an underrelaxation parameter.

The linear iterations decouple the meanflow equations from the turbulence-model equation. For the meanflow equations, the approximate Jacobian matrix, $[\widetilde{\partial \mathbf{R}/\partial \mathbf{Q}}]$, is formed based on linearization of first-order inviscid fluxes and second-order viscous fluxes. The approximate Jacobian of the turbulence model includes linearization of the convection, diffusion, and source terms. Positivity of the diagonal term in linearization of the SA model is enforced prior to adding the pseudotime term. The Jacobian is based on linearization of the van-Leer inviscid flux. Nonlinear iterations can reuse Jacobian computed at a previous iteration; a "smart" Jacobian update is scheduled depending on the residual convergence.

At each nonlinear iteration, the linear system represented by Eq. (2) is solved by a specified number of point-implicit GS sweeps with multicolor ordering; 15 and 10 sweeps are specified for the meanflow and turbulence equations, respectively. The resulting underrelaxed solution update, $\omega\Delta\mathbf{Q}$, is added to the nonlinear solution at the n-th iteration. The updated nonlinear solution is required to satisfy the realizability constraints at each grid point. Realizability is satisfied if local values of density and pressure are greater than a small positive threshold. The solutions reconstructed at dual-cell faces for second-order accuracy should also satisfy the realizability constraints. Reconstructing a negative pressure or density causes a so-called catastrophic failure. In case of a catastrophic failure, the second-order solution reconstruction is automatically reduced to a zero-gradient (constant) reconstruction reducing accuracy to first order. The CFL number can be ramped linearly within a specified number of nonlinear iterations.

C. HANIM Algorithm

In contrast to the legacy iterative solver, HANIM is based upon a hierarchy of modules illustrated in Fig. 1. Each HANIM iteration starts from a realizable solution. HANIM modules generate a solution update and report success or failure based on the level of residual reduction and realizability of the updated solution. If any module failed, then the proposed update is discarded, CFL is reduced, and nonlinear iteration repeats from the same solution. If all modules report success, CFL is increased, solution is updated, and HANIM proceeds to the next nonlinear iteration. A similar HANIM has been previously implemented in a NASA cell-centered CFD code, USM3D.[19–21] The FUN3D implementation closely follows the USM3D HANIM formulation. Below, the key components of FUN3D HANIM are described in detail.

1. Preconditioner

Similar to the legacy iterative solver, the preconditioner uses a defect-correction scheme (Eq. (2)), a meanflow Jacobian based on a first-order approximation to inviscid fluxes, multicolor point-implicit GS iterations, and the same linearization of the turbulence-model equation. In distinction from the legacy iterative solver, the HANIM preconditioner

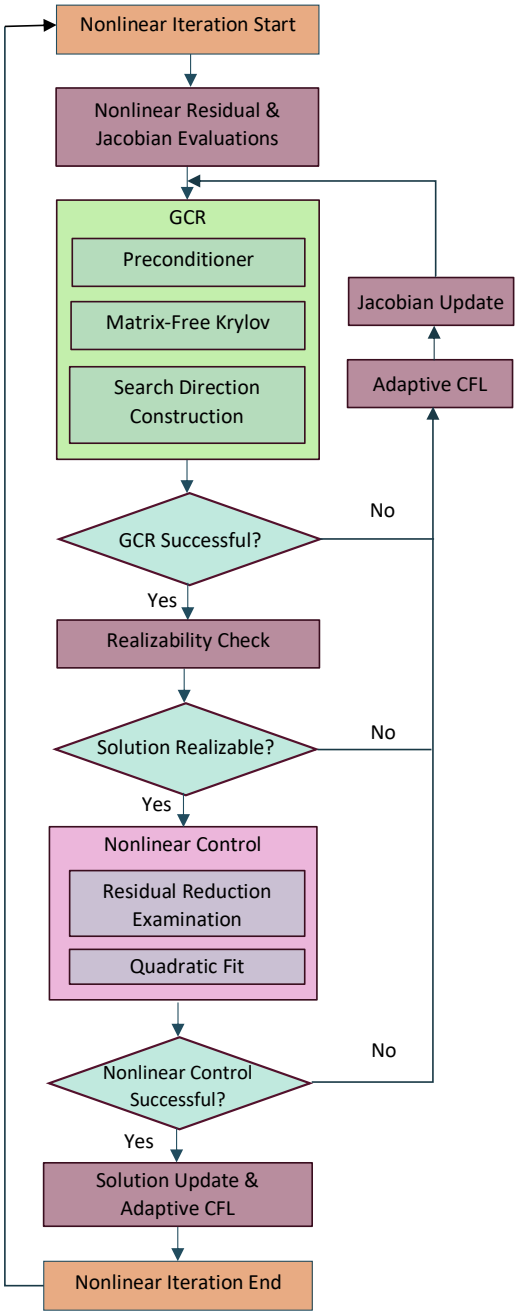


Figure 1. Flowchart illustrating key elements of HANIM algorithm.

always uses a Jacobian based on linearization of a consistent inviscid flux. That is the same approximate Riemann solver that is used in evaluating the Jacobian and nonlinear residual. The preconditioner Jacobian is updated at the beginning of each nonlinear iteration using the most recent solution. The meanflow and turbulence-model equations are decoupled and solved in consecutive order. The initial solution update, $\Delta\mathbf{Q}$, is set to zero.

The preconditioner residual is defined as:

$$\mathbf{r} = -\mathbf{R}(\mathbf{Q}^n) - \left[\frac{V}{\Delta\tau}\mathbf{I} + \widetilde{\frac{\partial \mathbf{R}}{\partial \mathbf{Q}}}\right]\Delta\mathbf{Q}. \tag{4}$$

The multicolor point-implicit GS iterations continue until either the root-mean-square (RMS) norm of the preconditioner residual has attained the prescribed reduction level or the maximum number of sweeps have been performed or preconditioner divergence has been detected. The divergence is detected when the RMS norm of the preconditioner residual has increased by more than two orders of magnitudes. In case of divergence, the preconditioner module reports failure and the solution process moves to the workflow handler that includes CFL and Jacobian updates and is discussed in Section II.C.5. If either the prescribed residual reduction or the maximum number of sweeps have been achieved, preconditioner reports success. The preconditioner solution, $\Delta\mathbf{Q}$, is then used by the GCR solver to form a search direction.

2. *GCR Solver*

The GCR method solves a linear system of equations representing exact linearization of the coupled meanflow and turbulence-model equations.

$$\left[\frac{V}{\Delta\tau}\mathbf{I} + \frac{\partial \mathbf{R}}{\partial \mathbf{Q}}\right]\Delta\mathbf{Q} = -\mathbf{R}(\mathbf{Q}^n) \tag{5}$$

Here, $[\partial\mathbf{R}/\partial\mathbf{Q}]$ is the exact linearization of the coupled nonlinear residual, Eq. (1). For second-order finite-volume discretizations, the exact linearization depends on solutions at neighbors of neighboring points. To circumvent the increased computational cost and storage requirements, the Frechet derivative approach is used to evaluate the matrix-vector product shown in Eq. (5):

$$\frac{\partial \mathbf{R}}{\partial \mathbf{Q}}\Delta\mathbf{Q} \approx \frac{\mathbf{R}\left(\mathbf{Q} + \varepsilon\frac{\Delta\mathbf{Q}}{|\Delta\mathbf{Q}|}\right) - \mathbf{R}(\mathbf{Q})}{\varepsilon}|\Delta\mathbf{Q}|, \tag{6}$$

where $|\cdot|$ denotes an l_2 norm, $\Delta\mathbf{Q}$ is preconditioner solution, and ε is a perturbation parameter, which is scaled with the solution magnitude:

$$\varepsilon = \max(1, \|\mathbf{Q}^n\|)\varepsilon_0. \tag{7}$$

Here, $\|\cdot\|$ denotes an RMS norm. Frechet derivative accuracy is sensitive to round-off errors. The perturbation factor $\varepsilon_0 \approx 10^{-7}$ is suitable for turbulent flows in moderate or high speeds. For low-speed turbulent or laminar flows, $\varepsilon_0 \approx 10^{-4}$ provides a better accuracy of Frechet derivative and benefits iterative convergence. Furthermore, to completely eliminate the round-off sensitivity associated with the Frechet derivative, the real-valued Frechet derivative can be replaced with a complex-valued Frechet derivative using an imaginary perturbation of magnitude $\varepsilon_1 = 10^{-50}$. In this case, the complex-valued Frechet derivative is evaluated as

$$\frac{\partial \mathbf{R}}{\partial \mathbf{Q}}\Delta\mathbf{Q} \approx \frac{imag\left[\mathbf{R}\left(\mathbf{Q} + i\varepsilon_1\frac{\Delta\mathbf{Q}}{|\Delta\mathbf{Q}|}\right)\right]}{\varepsilon_1}|\Delta\mathbf{Q}|. \tag{8}$$

This complex-variable derivative is practically exact, but carries a computational penalty as the complex-mode evaluation of the nonlinear residual is about three times slower than the corresponding real-mode evaluation. The real-valued Frechet derivative is used in all computations reported in this work.

The RMS and l_2 norms can be weighted, for example, to reduce the influence of large cells of a computational grid. In unstructured-grid CFD computations, large cells are often present near farfield boundaries. Small variations in a solution may lead to spurious residual fluctuations due to large grid metrics. The weighted norms are defined as

$$\mathrm{RMS}_w = \sqrt{\frac{\sum_{i=1}^{N} (w_i r_i)^2}{N}}, \qquad l_{2,w} = \sqrt{\sum_{i=1}^{N} (w_i r_i)^2}, \tag{9}$$

where r_i is the i-th component of vector $\mathbf{r}$, w_i is the weight applied to this component, and N is the total number of equations solved on a given grid. Two weighting options have been implemented in FUN3D HANIM. The first option sets the weight to be inversely proportional to the dual-cell volume with unity as the upper limit:

$$w_i = \min\left(1, \frac{1}{V_i}\right). \tag{10}$$

Here, V_i is the dual-cell volume of the grid point i. This option reduces the weight of larger cells while leaving the weight of smaller cells ($V_i < 1$) unchanged. The second option sets the weight to be inversely proportional to the characteristic cell size that is computed as the cubic root of the dual-cell volume for a three-dimensional grid:

$$w_i = \frac{1}{\sqrt[3]{V_i}}. \tag{11}$$

This option reduces the weight of large cells and increases the weight of small cells, e.g., in a viscous boundary layer.

The GCR module in the current HANIM implementation can use a maximum of K search directions to construct an approximate solution to Eq. (5); K is an input parameter. The corresponding solver is denoted as HANIM-K. Intially, the GCR correction is set to zero, $\Delta\mathbf{Q} = 0$. A new search direction is computed by a call to the preconditioner with the right side that represents the current GCR residual.

$$\left[\frac{V}{\Delta\tau}\mathbf{I} + \widetilde{\frac{\partial \mathbf{R}}{\partial \mathbf{Q}}}\right] \Delta\mathbf{Q}_k = \mathbf{r}_{k-1}, \quad k = 1, 2, \cdots, K. \tag{12}$$

The preconditioner iterations are performed for the decoupled linear system. Here, $\Delta\mathbf{Q}_k$ is the preconditioner solution to be used in forming the k-th search direction; $\mathbf{r}_{k-1}$ represents the GCR residual computed after applying the $(k-1)$-th search direction. The initial GCR residual is equal to the nonlinear residual evaluated at the beginning of the n-th iteration,

$$\mathbf{r}_0 = -\mathbf{R}(\mathbf{Q}^n). \tag{13}$$

The new search direction, $\mathbf{b}_k$, is computed and normalized as follows

$$\mathbf{b}_k = \frac{V}{\Delta\tau}\Delta\mathbf{Q}_k + \frac{\mathbf{R}\left(\mathbf{Q}^n + \varepsilon\dfrac{\Delta\mathbf{Q}_k}{|\Delta\mathbf{Q}_k|}\right) - \mathbf{R}(\mathbf{Q}^n)}{\varepsilon}\,|\Delta\mathbf{Q}_k| \tag{14}$$

$$\mathbf{b}_k = \frac{\mathbf{b}_k}{|\mathbf{b}_k|}, \quad \Delta\mathbf{Q}_k = \frac{\Delta\mathbf{Q}_k}{|\mathbf{b}_k|}. \tag{15}$$

Next, an orthonormalization procedure is performed for the search direction $\mathbf{b}_k$ against previously stored search directions $\mathbf{b}_j$ $(1 \le j < k)$; solution updates are changed accordingly.

$$\begin{gathered}\mu_k = \mathbf{b}_k^T \mathbf{b}_j, \quad \mathbf{b}_k = \mathbf{b}_k - \mu\mathbf{b}_j, \quad \Delta\mathbf{Q}_k = \Delta\mathbf{Q}_k - \mu\Delta\mathbf{Q}_j, \\ \mathbf{b}_k = \frac{\mathbf{b}_k}{|\mathbf{b}_k|}, \quad \Delta\mathbf{Q}_k = \frac{\Delta\mathbf{Q}_k}{|\mathbf{b}_k|}.\end{gathered} \tag{16}$$

The new orthonormal search direction $\mathbf{b}_k$ is stored for future search-direction orthonormalization. The residual projection on the k-th search direction is calculated, then the GCR correction and GCR residual are updated

$$\gamma_k = \mathbf{b}_k^T \mathbf{r}_{k-1}, \quad \Delta\mathbf{Q} = \Delta\mathbf{Q} + \gamma_k \Delta\mathbf{Q}_k, \quad \mathbf{r}_k = \mathbf{r}_{k-1} - \gamma_k \mathbf{b}_k. \tag{17}$$

The GCR module declares success if the RMS_w norm of GCR residual is reduced below a specified reduction target, μ_{GCR}:

$$\frac{\|\mathbf{r}_k\|}{\|\mathbf{r}_0\|} \leq \mu_{GCR}. \tag{18}$$

The norms are computed for the coupled residual vectors. In case of GCR success, HANIM proceeds to the realizability check module discussed in the next section. If the target reduction has not been met, GCR repeats the search-direction procedure, Eqs. (12)–(18). If the target GCR residual reduction is not satisfied after applying the K-th search direction, the GCR module reports failure. Failure reported either by preconditioner or by GCR module results in interruption of the current nonlinear iteration, and HANIM proceeds to the workflow handler module.

3. Realizability Check

A successful GCR offers a solution update, $\Delta\mathbf{Q}$, for the nonlinear solution. The updated solution should be realizable. Realizability check immediately follows the GCR module.

In the realizability check, the solution correction, $\Delta\mathbf{Q}$, is added to the nonlinear solution, $\mathbf{Q}^n$. If the updated density and pressure evaluated at grid points remain positive, HANIM proceeds to the nonlinear control module. If, at any grid point, the updated density or pressure become smaller than the specified tolerance, the realizability check reports failure, and HANIM proceeds to the workflow handler module.

4. Nonlinear Control

The nonlinear control module seeks a global underrelaxation parameter, $0 \leq \omega \leq 1$, to maximize the residual reduction for the updated solution, $\mathbf{Q}^n + \omega\Delta\mathbf{Q}$. The underrelaxation factor is first set to unity, and the pseudotime nonlinear residual, $\mathbf{R}_\tau$, is evaluated as follows:

$$\mathbf{Q}^{n+1} = \mathbf{Q}^n + \Delta\mathbf{Q}, \tag{19}$$

$$\mathbf{R}_\tau(\mathbf{Q}^{n+1}) = \frac{V}{\Delta\tau}\Delta\mathbf{Q} + \mathbf{R}(\mathbf{Q}^{n+1}). \tag{20}$$

The nonlinear control reports success if

$$\|\mathbf{R}_\tau(\mathbf{Q}^{n+1})\| / \|\mathbf{R}(\mathbf{Q}^n)\| \leq \mu_{nl} \tag{21}$$

where μ_{nl} is the residual reduction target set as

$$\mu_{nl} = 0.5(1 + \mu_{GCR}). \tag{22}$$

If the target residual reduction is satisfied, HANIM proceeds to the workflow handler module.

In case $\|\mathbf{R}_\tau(\mathbf{Q})\| / \|\mathbf{R}(\mathbf{Q}^n)\| > \mu_{nl}$, the the magnitude of the current correction, $\Delta\mathbf{Q}$, is compared to the magnitude of the round-off error. Round-off errors can play a noticeable role at the final stage of convergence. In the current work, corrections that satisfy $\|\Delta\mathbf{Q}\| / \|\mathbf{Q}^n\| \leq 10^{-12}$, are considered to be dominated by round-off errors. In such cases, if $\|\mathbf{R}_\tau(\mathbf{Q}^{n+1})\|$ achieves any reduction relative to $\|\mathbf{R}(\mathbf{Q}^n)\|$, the nonlinear control module reports success, but CFL is frozen.

If $\|\mathbf{R}_\tau(\mathbf{Q}^{n+1})\| / \|\mathbf{R}(\mathbf{Q}^n)\| > \mu_{nl}$ and $\|\Delta\mathbf{Q}\| / \|\mathbf{Q}^n\| > 10^{-12}$, then a quadratic search is performed to determine the underrelaxation factor in order to improve residual reduction. To determine the underrelaxation factor, two scalar functions are introduced to parametrize variation of $\mathbf{R}_\tau$ and the GCR residual,

$$f(\omega) = \left\| \frac{V}{\Delta\tau}\omega\Delta\mathbf{Q} + \mathbf{R}(\mathbf{Q}^n + \omega\Delta\mathbf{Q}) \right\|, \tag{23}$$

$$g(\omega) = \left\| \frac{V}{\Delta\tau}\omega\Delta\mathbf{Q} + \frac{\partial\mathbf{R}}{\partial\mathbf{Q}}\omega\Delta\mathbf{Q} + \mathbf{R}(\mathbf{Q}^n) \right\|. \tag{24}$$

Assuming quadratic variation for $f(\omega) = a\omega^2 + b\omega + c$ and linear variation for $g(\omega) = b\omega + c$, the following relations hold: $c = f(0)$, $b = g(1) - c$, and $a = f(1) - b - c$. Quadratic function $f(\omega)$ yields the minimum at $\omega = -b/(2a)$. Substituting the expressions of a, b, and c yields:

$$\omega = -\frac{g(1) - f(0)}{2(f(1) - g(1))}. \tag{25}$$

Here, $f(0) = \|\mathbf{R}(\mathbf{Q}^n)\|$, $f(1) = \|\mathbf{R}_\tau(\mathbf{Q}^{n+1})\|$, and $g(1) = \left\| \frac{V}{\Delta\tau}\Delta\mathbf{Q} + \frac{\partial\mathbf{R}}{\partial\mathbf{Q}}\Delta\mathbf{Q} + \mathbf{R}(\mathbf{Q}^n) \right\|$ is the RMS_w norm of the linear residual computed by the GCR module (c.f. Eq. (17)). The underrelaxation factor determined by Eq. (25) is used to update the nonlinear solution as, $\mathbf{Q}^{n+1} = \mathbf{Q}^n + \omega\Delta\mathbf{Q}$. If $\|\mathbf{R}_\tau(\mathbf{Q}^{n+1})\|$ achieves the target residual reduction, Eq. (21), the nonlinear control reports success, but CFL is frozen. If the target reduction of $\|\mathbf{R}_\tau(\mathbf{Q}^{n+1})\|$ is not attained and round-off error is not dominant, the nonlinear control reports failure, and HANIM proceeds to the workflow handler module.

5. Workflow Handler

The workflow handler module processes the success/failure and CFL-update messages generated by other HANIM modules and prepares HANIM for the next nonlinear iteration.

In case of success, the CFL number is either increased by a specified factor as $\mathrm{CFL} = \mathrm{CFL} \times \mathrm{CFL}_{incr}$ or remains frozen and the Jacobian matrix, $[\widetilde{\partial\mathbf{R}}/\partial\mathbf{Q}]$, is recomputed using the updated solution. In case of failure, the solution is restored to the state, $\mathbf{Q}^n$, at the beginning n-th nonlinear iteration, and the CFL number is reduced by a specified decrease factor as $\mathrm{CFL} = \mathrm{CFL} \times \mathrm{CFL}_{decr}$. The Jacobian matrix is updated to accommodate the reduced CFL number. A cost-effective procedure updates the diagonal terms of the Jacobian without recomputing linearized fluxes.

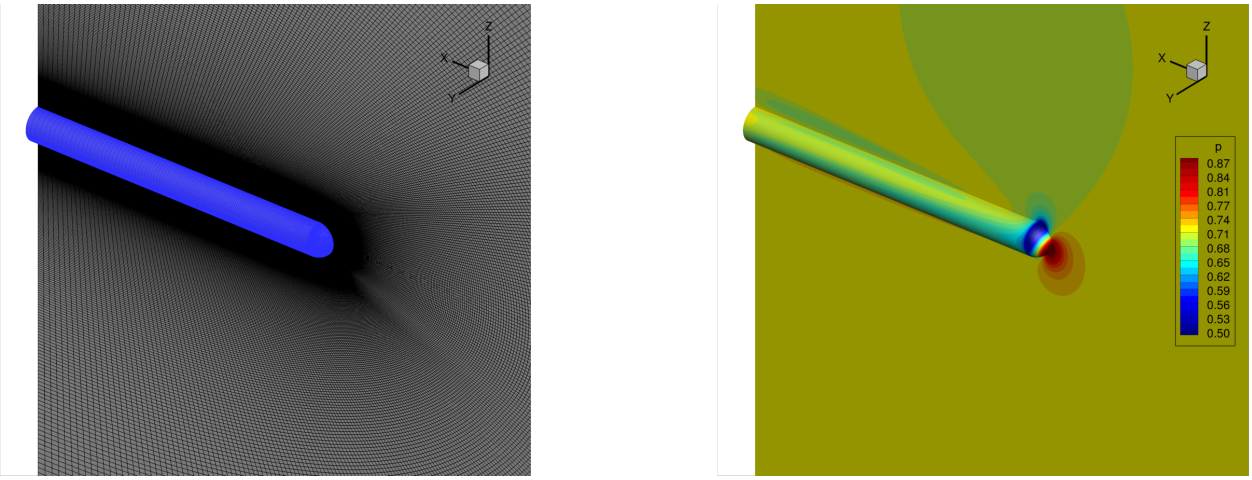
In the current work, the CFL adaptation parameters are set to $\mathrm{CFL}_{incr} = 2$ and $\mathrm{CFL}_{decr} = 0.1$. The CFL number is bounded by the maximum and minimum limits $\mathrm{CFL}_{max} = 10^{10}$ and $\mathrm{CFL}_{min} = 10^{-10}$. This range of 20 orders of magnitude is expected to be sufficient for many applications, although adjustments can be made if needed. If CFL reaches CFL_{min}, HANIM terminates nonlinear iterations, as solution updates become too small to make any significant changes in the nonlinear solution.

III. Numerical Results

In this section, the performance of HANIM is compared with the performance of a baseline preconditioner-alone (PA) solver. The PA solver closely resembles the legacy iterative solver of FUN3D with two differences. (1) The approximate Jacobian is based on Roe's flux difference splitting to provide consistency with the inviscid flux used in the nonlinear residual. (2) Similarly to HANIM, the Jacobian is updated in each nonlinear iteration. Iterative convergence is examined for three benchmark cases, namely, a subsonic separated flow around a hemisphere-cylinder configuration, a supersonic flow through a long duct with a square cross-section, and a subsonic flow over the NASA wing-body juncture configuration.

A. Subsonic Hemisphere Cylinder

The first test case is a subsonic flow around a three-dimensional hemisphere-cylinder configuration at a freestream Mach number of 0.6 and an angle of attack of 19 degrees. The geometry and flow conditions are presented in litera-



(a) L2 grid

(b) Pressure contours

Figure 2. Computational grid and pressure contours for subsonic flow around a hemisphere cylinder.

ture[15] and at the TMR website. [c] In the experiment,[31] the cylinder diameter (D) was 1 inch, the length was 10 inches, and the Reynolds number was 350,000 per inch. The reference temperature was 540°Rankine.

A computational grid containing 8,995,153 grid points and 9,830,400 hexahedral and prismatic elements is used in this study. The origin of the coordinate system is placed at the hemisphere apex. The positive x-direction coincides with the cylinder axis, the positive z-direction is up. A symmetry boundary condition is imposed at the $y = 0$ plane. A no-slip adiabatic-wall boundary condition is imposed at the hemisphere cylinder surface. Farfield freestream boundary conditions are specified at the outer quadrant of the sphere with a radius of $100D$ centered at $(x, y, z) = (10, 0, 0)$. The plane at $x = 10D$ is assigned with a back-pressure outflow boundary condition[30] and the unit ratio of the static and reference pressure. The hemisphere-cylinder flow is simulated with the SA-neg turbulence model. The farfield Spalart turbulence variable is $\hat{\nu}_{farfield} = 3.0\hat{\nu}_\infty$, where $\hat{\nu}_\infty$ is the freestream laminar kinematic viscosity. The meanflow Prandtl number is 0.72, and turbulence-model Prandtl number is 0.9. Figure 2(a) illustrates the geometry and the computational grid within the symmetry plane. Figure 2(b) displays pressure contours near the hemisphere-cylinder surface.

In HANIM, the maximum number of preconditioner GS iterations is set as 200 for both meanflow and turbulence equations. The target preconditioner residual reduction is 0.5. The GCR tolerance, μ_{GCR}, is set to 0.92 and only one search direction is used in each nonlinear iteration (HANIM-1).

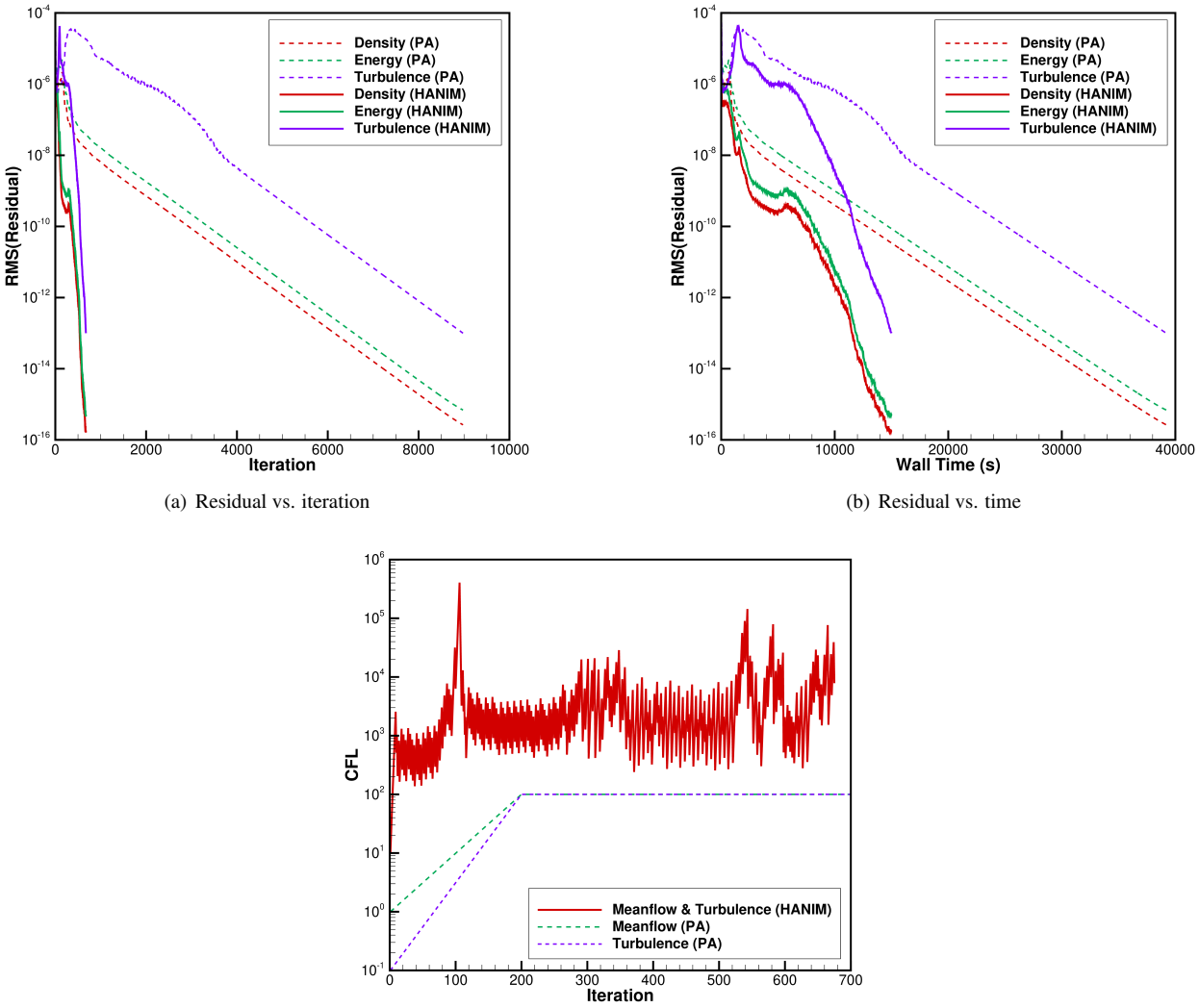
Convergence histories of the baseline PA and HANIM solvers are shown in Figs. 3(a) and 3(b), where the unweighted RMS norm of the steady-state residual is plotted versus the nonlinear iterations and computational time, respectively. To attain convergence, HANIM operates at much higher CFL and needs on an order of magnitude fewer nonlinear iterations compared to the PA solver. The speedup in terms of time to solution is more modest, about a factor of 2.6. In the final paper, the performance of HANIM will be profiled and further optimized.

B. Supersonic Duct Flow

The second test case involves a supersonic turbulent flow in a long duct with a square cross-section. The reference Mach number is 3.9, the Reynolds number is 508,000 based on the duct height or width, $D = 25.4mm$, and the reference temperature is 520°Rankine. The walls of the duct are modeled as adiabatic solid walls. Symmetry boundary conditions are applied at two boundaries; due to symmetry, only one quarter of the duct is computed.

The origin of the coordinate system is set at the lower-left corner of the duct inlet. The positive direction of the x-axis is the streamwise direction along the duct. The CFD grid is extended upstream of $x = 0$ to $x = -1.26829D$ to allow for a small run before the duct inlet using a symmetry boundary condition. Figure 4 shows the computational

[c] https://turbmodels.larc.nasa.gov/hc3dnumericspart2_val.html, accessed 06/01/2020



(a) Residual vs. iteration

(b) Residual vs. time

(c) CFL vs. iteration

Figure 3. Convergence of HANIM and PA solver for subsonic flow around hemisphere cylinder.

grid containing 3,170,833 nodes and 3,096,576 hexahedral elements that is used in this study. An enlarged scale for the y- and z-axes is used for ease of view. The boundaries $y = 0.5D$ and $z = 0.5D$ (shown in red) are locations of symmetry boundary conditions. The boundaries $y = 0$ and $z = 0$ (shown in blue) are assigned with a no-slip, adiabatic-wall boundary condition. The boundaries $x = -1.26829D$ and $x = 52D$ (shown in green) are the supersonic inflow and supersonic outflow boundary conditions, respectively. The simulations are performed with the SA-neg-QCR2000[24] turbulence model and without flux or gradient limiters.The farfield Spalart turbulence variable is $\hat{\nu}_{farfield} = 3.0\hat{\nu}_{\infty}$. The meanflow Prandtl number is 0.72, and the turbulent Prandtl number is 0.9.

The maximum number of GS multicolor iterations in the preconditioner is 100, for both meanflow and turbulence-model equations. The preconditioner residual reduction target is 0.2. The maximum number of search directions is 4 (HANIM-4) and the maximum CFL is 1,000. Weighting corresponding to Eq.(10) is used. The GCR residual reduction target is 0.92. Both PA and HANIM solvers use 240 processing cores.

Convergence of the unweighted RMS norm of the nonlinear residual is shown in Fig. 5 versus nonlinear iterations and computational time. PA and HANIM took about 9,720 and 1,280 nonlinear iterations, respectively, to converge

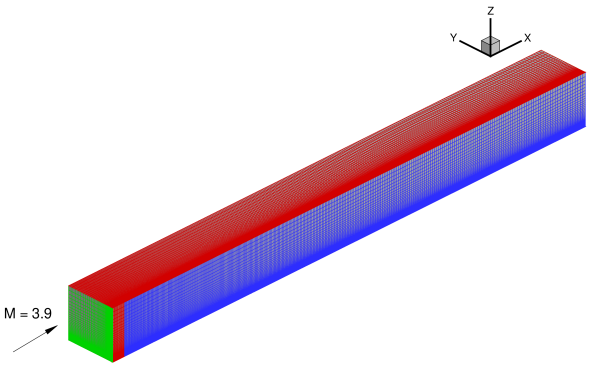
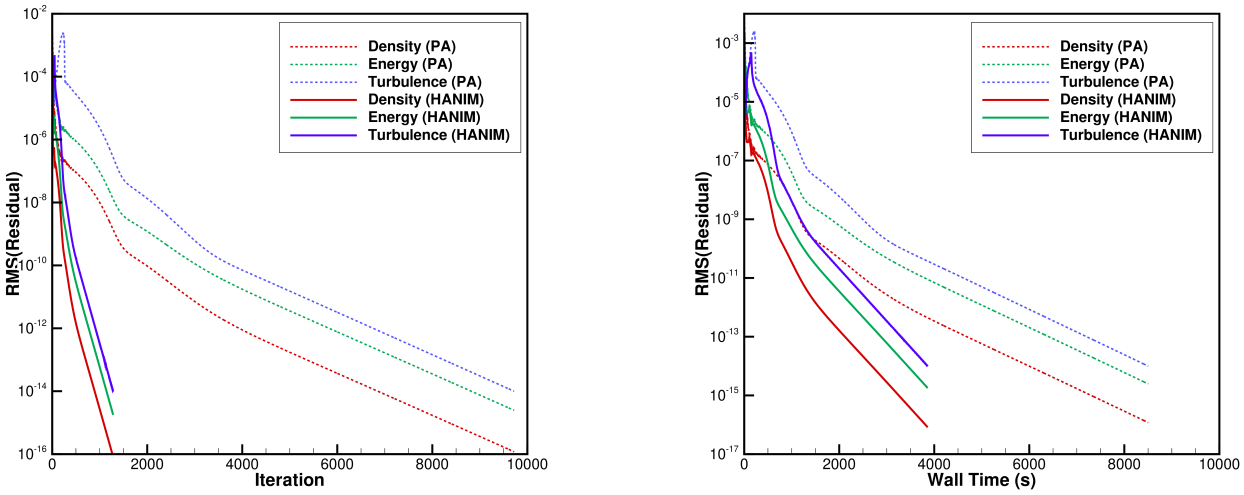


Figure 4. Computational grid for supersonic square-duct flow. Scales on y and z axes are exaggerated ($x : y : z = 1 : 10 : 10$).



(a) Residual vs. iteration

(b) Residual vs. time

Figure 5. Convergence comparisons between PA and HANIM solvers for supersonic square-duct flow.

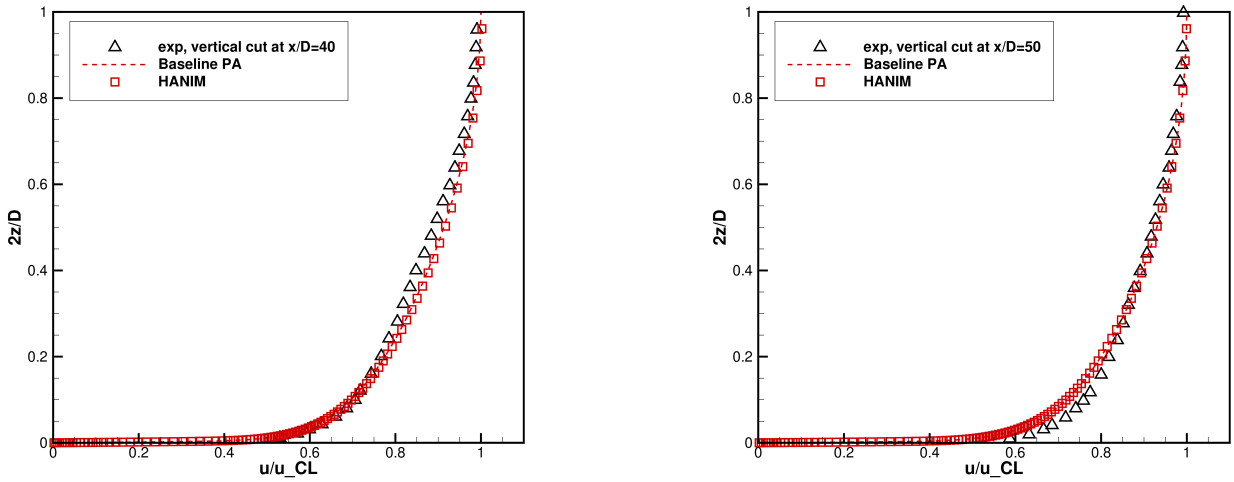
residuals to 10^{-14}. HANIM achieves a speedup factor of 2.24, as shown in Fig. 5(b). Note that at any moment after 1,000 wall-time seconds, HANIM residuals are at least two orders of magnitude lower than PA residuals.

Figure 6 shows streamwise-velocity profiles along with the vertical line of $y = 0.5D$ at two downstream locations in the duct, $x/D = 40$ and $x/D = 50$. The streamwise velocity is normalized by the centerline velocity, u_{CL}, obtained from the numerical solution at the specific x/D location. The flow velocity computed on the current grid is in good agreement with experimental measurements at both locations. As expected, PA and HANIM solutions are identical.

C. Juncture Flow Model

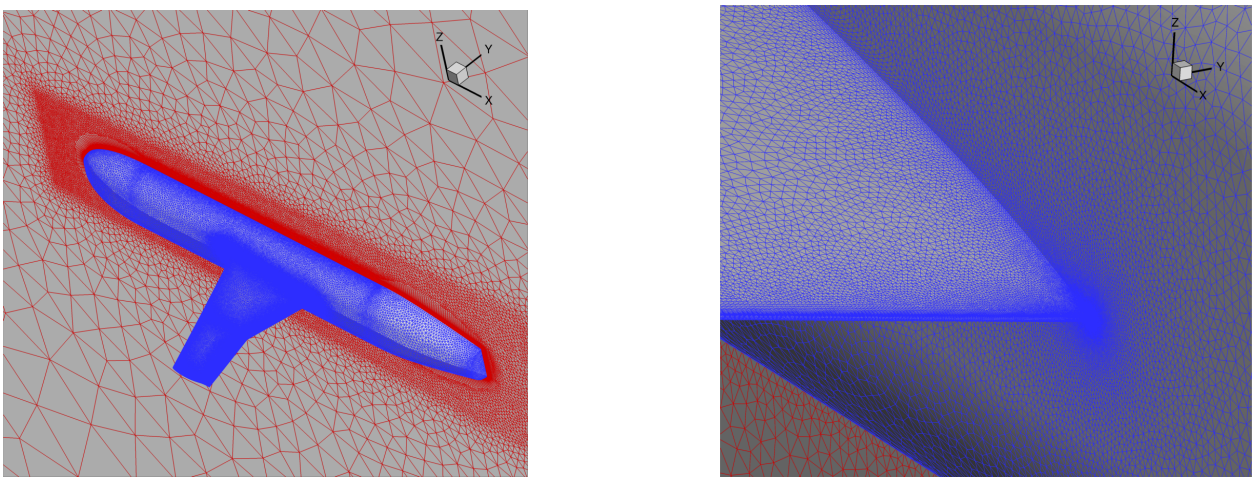
The next test case involves RANS simulations of flow around the NASA wing-body juncture configuration. This flow has been extensively studied[2,32–34] to assess the capability of various turbulence models to predict juncture-flow separation. Reference 32 showed that nonlinear RANS models using quadratic constitutive relations consistently improve separation predictions.

The flow condition in this test example is a freestream Mach number of 0.189, a Reynolds number of 2,400,000 based on the crank chord of 557.17mm, a reference temperature of 288.84 Kelvin, and an angle of attack of 5 degrees. A half model is considered with a symmetry boundary condition applied at the $y = 0$ plane as shown in Fig. 7. A



(a) $x/D = 40$ (b) $x/D = 50$

Figure 6. Comparison of streamwise velocity at locations of $x/D = 40$ and $x/D = 50$ along vertical line of $y/D = 0.5$ for supersonic square-duct flow.



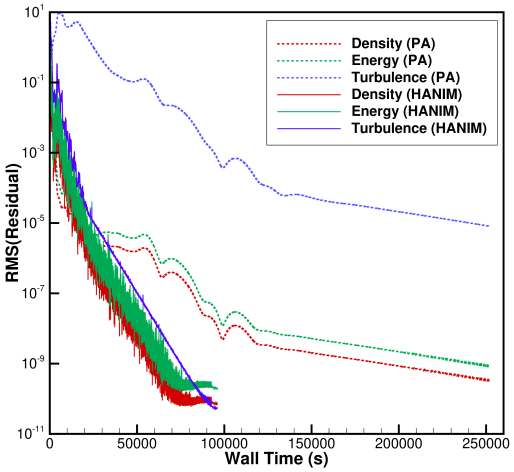
(a) Wing and fuselage (b) Closeup view of surface grid

Figure 7. Computational grid containing 13-million points for juncture-flow model simulations.

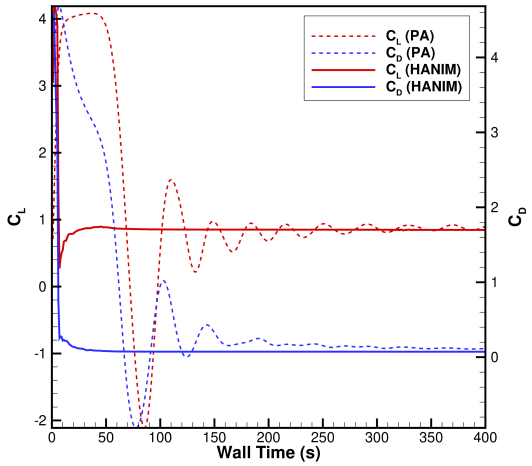
farfield freestream boundary condition is applied at the outer boundary. A no-slip adiabatic-wall boundary condition is applied at the fuselage and wing. The computational grid contains 13,036,210 points and a total of 37,126,819 mixed-typed elements including 16,645,072 tetrahedrons, 112,989 pyramids, and 20,368,758 prisms.

The turbulent stresses were computed based on SA-QCR2000 turbulence model.[24] The maximum number of GS multicolor iterations in the preconditioner was set to 100 for the meanflow and turbulence equations. The preconditioner residual convergence target is 0.5. The maximum number of search directions is set to 4 (HANIM-4). The maximum CFL is set to 1,000,000. Weighting (Eq.(11)) is selected. The target for the GCR residual reduction is set to 0.92. Both PA and HANIM use a total of 320 processing cores in the simulations.

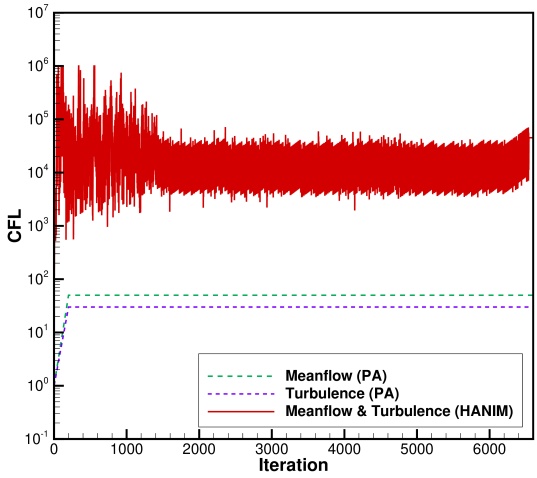
Convergence of the unweighted RMS residual norm and aerodynamic forces is shown in Fig. 8 versus computational time. In this case, the PA scheme did not achieve the target residual level within 72 wall-clock hours. In contrast, Fig. 8(a) shows that HANIM achieved more than eleven orders of residual reduction for meanflow and tur-



(a) Residual vs. time



(b) Forces vs. time



(c) CFL vs. iteration

Figure 8. Convergence comparisons between PA and HANIM solvers for juncture-flow model.

bulence model in less than 27.8 wall-clock hours (100,000 seconds). PA takes about 250,000 wall-clock seconds to converge the turbulence-model residual to the level of 10^{-5}. HANIM takes less than 25,000 seconds to converge the turbulence-model residual to the same level. Thus, at this residual level, HANIM achieves a speedup factor of 10; and the speedup factor increases for lower levels of residual.

Figure 8(b) shows convergence of lift and drag coefficients. While aerodynamic forces evaluated from the PA solution experience large oscillations, HANIM quickly converges lift and drag coefficients to their asymptotic levels. Fig. 8(c) compares the CFL number of HANIM and PA solver. The operational CFL number of HANIM is about 2-3 orders of magnitude greater than the maximum CFL number used by the PA solver.

Figure 9 shows a visualization of the converged HANIM solution. A closeup view of the streamlines near the wing-fuselage junction region showing the predicted flow separation is provided. The length of predicted flow separation in the junction region is about 164*mm* and the width is about 60*mm*. The predicted separation size is consistent with the numerical solutions reported in Ref. 32. More quantitative comparisons of meanflow velocities and Reynolds stresses with experiments as well as convergence assessments on finer grids will be provided in the final paper.

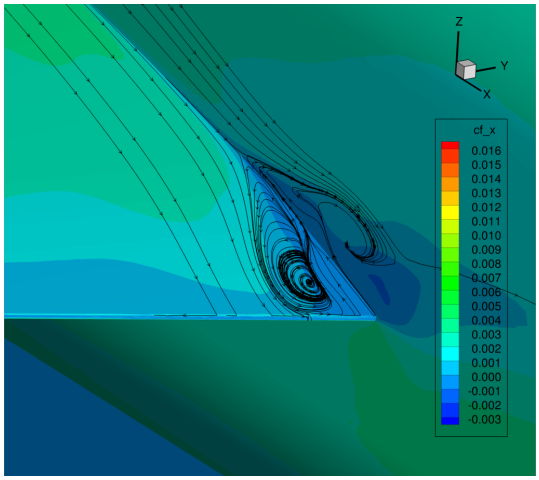


Figure 9. Streamlines near the juncture flow region colored by x-component of skin friction. Converged solution computed by HANIM.

IV. Conclusions

A hierarchical adaptive nonlinear iteration method (HANIM) has been implemented in the NASA computational fluid dynamics (CFD) solver, FUN3D. HANIM is based on a hierarchy of modules consisting of preconditioner, generalized conjugate residual, realizability check, nonlinear control, and CFL adaptation modules. This paper focuses on the performance of HANIM for steady-state Reynolds-averaged Navier-Stokes equations. Two one-equation Spalart-Allmaras (SA) turbulence models are considered: one is the negative variant of the linear model; the other is a nonlinear SA model based on quadratic constitutive relations.

Numerical experiments have been performed on three benchmarks including a subsonic separated flow around a hemisphere-cylinder configuration, a supersonic flow through a long square duct, and a subsonic flow over the NASA wing-body juncture configuration. HANIM performance has been compared with a baseline preconditioner-alone (PA) solver that closely resembles the legacy iterative solver of FUN3D.

In all the test examples, HANIM achieved improved robustness and computational efficiency of FUN3D solutions. A factor of 2.6 to 10 speedup has been achieved by HANIM over the PA solver. The final paper will provide additional details about new robustness features of HANIM. A detailed analysis of HANIM iterative convergence will be reported. More quantitative comparisons with experimental data will also be shown.

Acknowledgments

The work is supported by the NASA Revolutionary Computational Aerosciences Project, NASA Revolutionary Vertical Lift Technology Project, and NASA Langley High Performance Computing Incubator Program. The support is greatly appreciated.

References

[1]Slotnick, J., Khodadoust, A., Alonso, J., Darmofal, D., Gropp, W., Lurie, E., and Mavriplis, D., "CFD Vision 2030 Study: A Path to Revolutionary Computational Aerosciences," Tech. Rep. NASA/CR-2014-218178, NASA Langley Research Center, Hampton, Virginia, 2014.

[2]Abdol-Hamid, K. S., Ahmad, N. N., Carlson, J.-R., and Biedron, R., "Juncture Flow Computations using kL-Based Turbulence Models," *Proceedings of the AIAA SciTech Forum*, January 6–10, 2020, Orlando, Florida, AIAA Paper 2020–1305, doi.org/10.2514/6.2020-1305.

[3]Wang, L., Diskin, B., Biedron, R. T., Nielsen, E. J., Sonneville, V., and Bauchau, O. A., "High-Fidelity Multidisciplinary Design Optimization Methodology with Application to Rotor Blades," *Journal of the American Helicopter Society*, Vol. 64, No. 3, July 2019, pp. 1–11, doi.org/10.4050/JAHS.64.032002.

[4]Wang, L., Diskin, B., Lopes, L. V., Nielsen, E. J., Lee-Rausch, E., and Biedron, R. T., "High-Fidelity Multidisciplinary Design Optimization of Low-Noise Rotorcraft," *Proceedings of VFS 75th Annual Forum*, May 13–16 2019, Philadelphia, PA.

[5]Walden, A., Nielsen, E., Diskin, B., and Zubair, M., "A Mixed Precision Multicolor Point-Implicit Solver for Unstructured Grids on GPUs," *2019 IEEE/ACM 9th Workshop on Irregular Applications: Architectures and Algorithms (IA3)*, 2019, pp. 23–30.

[6]Jain, R., "Sensitivity Study of High-Fidelity Hover Predictions on the Sikorsky S-76 Rotor,," *Journal of Aircraft*, Vol. 55, No. 1, January–February 2018, pp. 78–88, doi.org/10.2514/1.C034076.

[7]Nielsen, E. J. and Diskin, B., "High-Performance Aerodynamic Computations for Aerospace Applications," *Parallel Computing*, Vol. 64, 2017, pp. 20–32, doi:10.1016/j.parco.2017.02.004.

[8]Lee-Rausch, E. M., Rumsey, C. L., and Park, M. A., "Grid-Adapted FUN3D Computations for the Second High-Lift Prediction Workshop," *Journal of Aircraft*, Vol. 52, No. 4, 2015, pp. 1098–1111.

[9]Rumsey, C., Slotnick, J., Long, M., Stuever, R., and Wayman, T., "Summary of the First AIAA CFD High-Lift Prediction Workshop," *Journal of Aircraft*, Vol. 48, No. 6, May 2011, pp. 2068–2079, doi.org/10.2514/1.C031447.

[10]Biedron, R. T. and Lee-Rausch, E. M., "Rotor Airloads Prediction Using Unstructured Meshes and Loose CFD/CSD Coupling," *Proceedings of the 26th AIAA Applied Aerodynamic Conference, AIAA Paper 2008–7341*, August 18–21, 2008, Honolulu, Hawaii.

[11]Bartels, R., Scott, R., Funk, C., Allen, T., and Sexton, B., "Computed and Experimental Flutter/LCO Onset for the Boeing Truss-Braced Wing Wind-Tunnel Model," *Proceedings of 44th AIAA Fluid Dynamics Conference*, June 16–20 2014, AIAA Paper 2014-2446, Atlanta, Georgia.

[12]Bozeman, M., Tai, J., Robertson, B., and Mavris, D., "Prediction of Dynamic Stability Derivatives for Flexible Aircraft using FUN3D," January 8–12 2018, AIAA SciTech Forum, AIAA Paper 2018-2063, Kissimmee, Florida.

[13]Edquist, K., Korzun, A., Bibb, K., Schauerhamer, G., Ma, E., McCloud, P., Palmer, G., and Monk, J., "Comparison of Navier-Stokes Flow Solvers to Falcon 9 Supersonic Retropropulsion Flight Data," *Proceedings of AIAA SPACE Forum*, September 12–14 2017, AIAA Paper 2017-5296, Orlando, Florida.

[14]Anderson, K., Newman, J., and Karman, S., "Stabilized Finite Elements in FUN3D," *Journal of Aircraft*, Vol. 55, No. 2, March–April 2017, pp. 696–714, 10.2514/1.C034482.

[15]Diskin, B., Anderson, K., Pandya, M., Rumsey, C., Thomas, J., Liu, Y., and Nishikawa, H., "Grid Convergence for Three Dimensional Benchmark Turbulent Flows," *Proceedings of AIAA SciTech Forum*, January 8–12 2018, AIAA Paper 2018-1102, Kissimmee, Florida.

[16]Diskin, B. and Nishikawa, H., "Evaluation of Multigrid Solutions for Turbulent Flows," *Proceedings of AIAA SciTech Forum*, January 13–17 2014, AIAA Paper 2014-0082, National Harbor, Maryland.

[17]Diskin, B. and Thomas, J., "Comparison of Node-Centered and Cell-Centered Unstructured Finite Volume Discretizations: Inviscid Fluxes," *AIAA Journal*, Vol. 49, No. 4, April 2011, pp. 836–854, doi.org/10.2514/1.J050897.

[18]Diskin, B., Thomas, J., Nielsen, E., Nishikawa, H., and White, J., "Comparison of Node-Centered and Cell-Centered Unstructured Finite-Volume Discretizations Part I: Viscous Fluxes," *AIAA Journal*, Vol. 48, No. 7, July 2010, doi.org/10.2514/1.44940.

[19]Pandya, M., Jespersen, D., Diskin, B., Thomas, J., and Frink, N., "Accuracy, Scalability, and Efficiency of Mixed-Element USM3D for Benchmark Three-Dimensional Flows," *Proceedings of AIAA SciTech Forum*, January 7–11 2019, AIAA Paper 2019–2333, San Diego, California.

[20]Pandya, M., Diskin, B., Thomas, J., and Frink, N., "Assessment of USM3D Hierarchical Adaptive Nonlinear Method Preconditioners for Three-Dimensional Cases," *AIAA Journal*, Vol. 55, 08 2017, pp. 1–16, 10.2514/1.J055823.

[21]Pandya, M. J., Diskin, B., Thomas, J. L., and Frink, N. T., "Improved Convergence and Robustness of USM3D Solutions on Mix-Element Grids," *AIAA Journal*, Vol. 54, No. 9, 2016, pp. 2589–2596, doi.org/10.2514/1.J054545.

[22]Spalart, P. and Allmaras, S., "A one-equation Turbulence Model for Aerodynamic Flows," *Le Recherche Aerospatiale*, Vol. 1, 1994, pp. 5–21, doi:10.2514/6.1992-439.

[23]Allmaras, S. R., Johnson, F. T., and Spalart, P. R., "Modifications and Clarifications for the Implementation of the Spalart-Allmaras Turbulence Model," *Proceedings of 7th International Conference on Computational Fluid Dynamics*, July 9–13 2012, ICCFD7–1902, BigIsland, Hawaii.

[24]Spalart, P., "Strategies for Turbulence Modelling and Simulation," *International Journal of Heat and Fluid Flow*, Vol. 21, No. 3, 2000, pp. 252–263, doi.org/10.1016/S0142-727X(00)00007-2.

[25]Anderson, W. K. and Bonhaus, D. L., "An Implicit Upwind Algorithm for Computing Turbulent Flows on Unstructured Grids," *Computers & Fluids*, Vol. 23, No. 1, 1994, pp. 1–21, doi:10.1016/0045-7930(94)90023-X.

[26]Biedron, R. T., Carlson, J.-R., Derlaga, J. M., Gnoffo, P. A., Hammond, D. P., Jones, W. T., Kleb, B., Lee-Rausch, E. M., Nielsen, E. J., Park, M. A., Rumsey, C. L., Thomas, J. L., and Wood, W. A., "FUN3D Manual: 13.6," NASA/TM-2019–220416, NASA Langley Research Center, Hampton, Virginia, 2019.

[27]Roe, P. L., "Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes," *Journal of Computational Physics*, Vol. 43, No. 2, 1981, pp. 357–372, doi:10.1016/0021-9991(81)90128-5.

[28]Burg, C. O. E., "Higher Order Variable Extrapolation For Unstructured Finite Volume RANS Flow Solvers," *Proceedings of 17th AIAA Computational Fluid Dynamics Conference*, June 6–9 2005, AIAA Paper 2005–4999, Toronto, Ontario Canada.

[29]van Leer, B., "Towards the Ultimate Conservative Difference Scheme, V. A Second Order Sequel to Godunov's Method," *Journal of Computational Physics*, Vol. 32, No. 1, 1979, pp. 101–136, doi:10.1016/0021-9991(79)90145-1.

[30]Carlson, J.-R., "Inflow/Outflow Boundary Conditions with Application to FUN3D," October 2011, NASA/TM–2011-217181, NASA Langley Research Center, Hampton, Virginia.

[31]Tsieh, T., "An Investigation of Separated Flow about A Hemisphere-Cylinder at 0- to 19-Deg Incidence In the Mach Number Range from 0.6 to 1.5," AEDC-TR-76-112, Propulsion Wind Tunnel Facility, Arnold Engineering Development Center, Air force Systems Command, Arnold Air Force Station, Tennessee, 1976.

[32]Rumsey, C. L., Carlson, J.-R., and Ahmad, N. N., "FUN3D Juncture Flow Computations Compared with Experimental Data," *Proceedings of the AIAA SciTech Forum*, January 7–11 2019.

[33]Rumsey, C. L., Lee, H. C., and Pulliam, T. H., "Reynolds-Averaged Navier-Stokes Computations of the NASA Juncture Flow Model Using FUN3D and OVERFLOW," *Proceedings of the AIAA SciTech Forum*, January 6–10, 2020, AIAA Paper 2020–1304, Orlando, Florida.

[34]Iyer, P. S. and Malik, M. R., "Wall-modeled LES of the NASA Juncture Flow Experiment," *Proceedings of the AIAA SciTech Forum*, January 6–10, 2020, AIAA Paper 2020–1307, Orlando, Florida.