AIAA Aviation
16-20 June 2014, Atlanta, GA
7th AIAA Theoretical Fluid Mechanics Conference

First, Second, and Third Order Finite-Volume Schemes for Navier-Stokes Equations

Hiroaki Nishikawa*
National Institute of Aerospace, Hampton, VA 23666, USA

In this paper, we present first-, second-, and third-order implicit finite-volume schemes for solving the Navier-Stokes equations on unstructured grids based on a hyperbolic formulation of the viscous terms. These schemes are first-, second-, and third-order accurate on irregular grids for both the inviscid and viscous terms and for all Reynolds numbers, not only in the primitive variables but also in the first-derivative quantities such as the velocity gradients for the incompressible Navier-Stokes equations and the viscous stresses and heat fluxes for the compressible Navier-Stokes equations. Numerical results show that the developed schemes are capable of producing highly accurate derivatives on highly-skewed grids whereas a conventional scheme suffers from severe oscillations on such grids. Moreover, these first-, second-, and third-order schemes enable the construction of implicit solvers that converge intrinsically faster in CPU time than a conventional second-order implicit Navier-Stokes solver with the acceleration factor growing in the grid refinement.

I. Introduction

In this paper, we present first-, second-, and third-order implicit finite-volume schemes for solving the Navier-Stokes (NS) equations on unstructured grids. These schemes are intrinsically more accurate and efficient than conventional schemes, having the following features:

1. Equal order of accuracy achieved on *irregular* grids for the primitive variables and derivative quantities such as the viscous stresses, the heat fluxes, and the vorticity.

2. $O(1/h)$ acceleration over conventional solvers in iterative convergence, where h is a typical mesh spacing.

This paper shows that the developed implicit solver can converge faster in CPU time on a given grid than a conventional second-order implicit NS solver, not only for the first- and second-order schemes but also for the third-order schemes. Moreover, the developed schemes are capable of producing smooth and accurate derivatives on highly-skewed grids whereas a conventional scheme suffers from severe oscillations on such grids, allowing highly arbitrary grid adaptation for viscous flows over complex geometries.

The first-, second-, and third-order NS schemes are constructed based on the first-order hyperbolic system method, or the hyperbolic method in short, introduced in Ref.1. In the hyperbolic method, the diffusion equation is written as a hyperbolic system such that it reduces to the diffusion equation in the steady state. The hyperbolic system is then discretized by methods for hyperbolic systems, and solved for the steady state. This method is different from stiff relaxation methods[2–5] in that the choice of the relaxation time is arbitrary and thus can be determined not by a physical constraint but solely by numerical considerations such as fast iterative convergence. Also, it is different from the mixed finite-element method[6] and other first-order system methods[7,8] in that our system is hyperbolic in (pseudo) time while their systems have no such characterization. It is also different from the method in Ref.9 in that our relaxation time is arbitrary and $O(1)$ whereas their relaxation time depends on the order of accuracy, e.g., $T_r = O(h^2)$ for second-order schemes, to make the hyperbolic formulation time accurate[a]. As a result, the $O(1/h)$ convergence acceleration cannot be obtained in their method. We emphasize that the method considered in this paper is unique in that the hyperbolic system is designed such that it recovers the original equation in the steady state, and is thus targeted at solving

*Associate Research Fellow (hiro@nianet.org), National Institute of Aerospace, 100 Exploration Way, Hampton, VA 23666 USA

[a]In Ref.9, it is stated that the hyperbolic method was first introduced in 2004 in Ref.10, but it is not true. The hyperbolic method was first introduced in 2007 in Ref.1.

the original diffusion and Navier-Stokes equations, not their asymptotic approximations. Consequently, time-accurate schemes cannot be constructed by explicit time-stepping schemes, but can be constructed by implicit time-stepping schemes or space-time methods as already demonstrated in Ref.11. As a matter of distinction, therefore, if time-accurate computations can be performed by an explicit time-stepping scheme, then the scheme is different from the hyperbolic method.

The hyperbolic method was first introduced in Ref.1 for diffusion, demonstrating the equal order of accuracy in the solution and the gradients and accelerated convergence by a second-order residual-distribution (RD) scheme on regular triangular grids. It was then extended to the advection-diffusion equation in Ref.12 where the advective and diffusive terms are integrated into a single hyperbolic system and discretized by a multi-dimensional upwinding RD scheme, demonstrating the superior accuracy and convergence on irregular triangular grids. The method was first extended to the compressible Navier-Stokes equations in Ref.13, where a second-order finite-volume scheme was constructed based on a separate treatment of the inviscid and hyperbolized viscous terms. The development of first-, second-, and third-order finite-volume schemes based on the hyperbolic method was initiated in Ref.14. Introducing the divergence formulation of source terms in Ref.15, we have reformulated the hyperbolic system for diffusion in order to simplify the construction of the third-order scheme. These schemes have been extended to the advection-diffusion equation in Ref.16, where an implicit solver is developed for efficient computations on stretched grids. In this paper, the first-, second-, and third-order finite-volume schemes are extended to the Navier-Stokes equations. For the compressible NS equations, the hyperbolic NS system proposed in Ref.13 is employed, which has the viscous stresses and the heat fluxes as additional variables. For the incompressible NS equations, a hyperbolic viscous system is constructed by following the previous work as described in Refs.14, 16. As is done in many conventional schemes, NS schemes are constructed as a sum of an inviscid scheme and a viscous scheme - the viscous scheme is an upwind scheme because the viscous terms are now hyperbolic. To enable efficient computations for practical problems, we develop an implicit solver based on the Jacobian that is exact for the first-order scheme: Newton's method for the first-order scheme, and a defect-correction method for the second- and third-order schemes.

II. Hyperbolic Compressible NS System

Consider the compressible NS equations:

$$\partial_t \rho + \mathrm{div}(\rho \mathbf{v}) = 0, \tag{II.1}$$

$$\partial_t (\rho \mathbf{v}) + \mathrm{div}(\rho \mathbf{v} \otimes \mathbf{v}) = -\mathrm{grad}\, p + \mathrm{div}\, \boldsymbol{\tau}, \tag{II.2}$$

$$\partial_t (\rho E) + \mathrm{div}(\rho \mathbf{v} H) = \mathrm{div}\,(\boldsymbol{\tau}\mathbf{v}) - \mathrm{div}\, \mathbf{q}, \tag{II.3}$$

where $\otimes$ denotes the dyadic product, ρ is the density, $\mathbf{v}$ is the velocity vector, p is the pressure, E is the specific total energy, and $H = E + p/\rho$ is the specific total enthalpy. The viscous stress tensor, $\boldsymbol{\tau}$, and the heat flux, $\mathbf{q}$, are given by

$$\boldsymbol{\tau} = -\frac{2}{3}\mu(\mathrm{div}\,\mathbf{v})\mathbf{I} + \mu\left(\mathrm{grad}\,\mathbf{v} + (\mathrm{grad}\,\mathbf{v})^t\right), \quad \mathbf{q} = -\frac{\mu}{Pr(\gamma-1)}\mathrm{grad}\,T, \tag{II.4}$$

where T is the temperature, γ is the ratio of specific heats, Pr is the Prandtl number, and μ is the viscosity defined by Sutherland's law. Stokes' hypothesis has been assumed. All the quantities are assumed to have been nondimensionalized by their free-stream values except that the velocity and the pressure are scaled by the free-stream speed of sound and the free-stream dynamic pressure, respectively (see Ref.17). Thus, the viscosity is given by the following form of Sutherland's law:

$$\mu = \frac{M_\infty}{Re_\infty}\frac{1 + C/\tilde{T}_\infty}{T + C/\tilde{T}_\infty}T^{\frac{3}{2}}, \tag{II.5}$$

where $\tilde{T}_\infty$ is the dimensional free stream temperature, and $C = 110.5$ [K] is the Sutherland constant. The ratio of the free stream Mach number, M_∞, to the free stream Reynolds number, Re_∞, arise from the nondimensionalization. The system is closed by the nondimensionalized equation of state for ideal gases: $\gamma p = \rho T$.

We develop a steady solver for the compressible NS equations by discretizing the hyperbolic NS system

proposed in Ref.13:

$$\partial_t \rho + \mathrm{div}(\rho \mathbf{v}) = 0, \tag{II.6}$$

$$\partial_t (\rho \mathbf{v}) + \mathrm{div}(\rho \mathbf{v} \otimes \mathbf{v}) = -\mathrm{grad}\, p + \mathrm{div}\, \boldsymbol{\tau}, \tag{II.7}$$

$$\partial_t (\rho E) + \mathrm{div}(\rho \mathbf{v} H) = \mathrm{div}\,(\boldsymbol{\tau} \mathbf{v}) - \mathrm{div}\, \mathbf{q}, \tag{II.8}$$

$$\frac{T_v}{\mu_v}\, \partial_t \boldsymbol{\tau} = -\frac{1}{2}(\mathrm{div}\, \mathbf{v})\mathbf{I} + \frac{3}{4}\left(\mathrm{grad}\, \mathbf{v} + (\mathrm{grad}\, \mathbf{v})^t\right) - \frac{\boldsymbol{\tau}}{\mu_v}, \tag{II.9}$$

$$\frac{T_h}{\mu_h}\, \partial_t \mathbf{q} = -\frac{1}{\gamma(\gamma-1)} \mathrm{grad}\, T - \frac{\mathbf{q}}{\mu_h}, \tag{II.10}$$

where $\mathbf{I}$ is the identity matrix, μ_v and μ_h are the scaled viscosities,

$$\mu_v = \frac{4}{3}\mu, \quad \mu_h = \frac{\gamma \mu}{Pr}, \tag{II.11}$$

T_v and T_h are relaxation times associated with the viscous stress and the heat flux. Note that we have simply added evolution equations for the viscous stresses and the heat fluxes to the NS equations; the continuity, momentum, and energy equations remain intact. Note also that the system is equivalent to the NS equations in the steady state for any nonzero relaxation times; the time t is thus considered as the pseudo time throughout the paper. Therefore, T_v and T_h are free parameters, which can be determined solely for fast steady convergence. Here, we define them as

$$T_v = \frac{L^2}{\nu_v}, \quad T_h = \frac{L^2}{\nu_h}, \tag{II.12}$$

where L is a length scale defined as $L = 1/(2\pi)$ in the previous work, and ν_v and ν_h are the kinematic viscosities,

$$\nu_v = \frac{\mu_v}{\rho}, \quad \nu_h = \frac{\mu_h}{\rho}. \tag{II.13}$$

The eigen-structure of the whole first-order system remains unknown, but it is not necessary to proceed[b]. The inviscid part is well known, and it is hyperbolic. The viscous part has also been shown to be hyperbolic in the pseudo time.[13] In this sense, we call the above first-order system the hyperbolic NS system. Note that the number of equations for the viscous stresses is actually 3 in two dimensions and 6 in three dimensions by symmetry. The total number of equations in the first-order system is, therefore, 9 in two dimensions and 14 in three dimensions.

In two dimensions, the hyperbolic NS system can be written as

$$\mathbf{P}^{-1} \partial_t \mathbf{U} + \partial_x \mathbf{F} + \partial_y \mathbf{G} = \mathbf{S}, \tag{II.14}$$

where

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \\ \tau_{xx} \\ \tau_{xy} \\ \tau_{yy} \\ q_x \\ q_y \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p - \tau_{xx} \\ \rho u v - \tau_{xy} \\ \rho u H - \tau_{xx} u - \tau_{xy} v + q_x \\ -u \\ -3v/4 \\ u/2 \\ \dfrac{T}{\gamma(\gamma-1)} \\ 0 \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} \rho v \\ \rho u v - \tau_{xy} \\ \rho v^2 + p - \tau_{yy} \\ \rho v H - \tau_{xy} u - \tau_{yy} v + q_y \\ v/2 \\ -3u/4 \\ -v \\ 0 \\ \dfrac{T}{\gamma(\gamma-1)} \end{bmatrix}, \tag{II.15}$$

[b]If the eigen-structure of the whole system were discovered, it would enable a unified construction of a numerical scheme for the inviscid and viscous terms and also allow us to attempt to derive an optimal relaxation time as demonstrated for the advection-diffusion equation in Ref.12.

$$\mathbf{S} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \tau_{xx}/\mu_v \\ \tau_{xy}/\mu_v \\ \tau_{yy}/\mu_v \\ q_x/\mu_h \\ q_y/\mu_h \end{bmatrix}, \quad \mathbf{P}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & T_v/\mu_v & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & T_v/\mu_v & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & T_v/\mu_v & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & T_h/\mu_h & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & T_h/\mu_h \end{bmatrix}, \tag{II.16}$$

where $\mathbf{v} = (u, v)$, $\mathbf{q} = (q_x, q_y)$, and three independent viscous stresses are denoted by τ_{xx}, τ_{xy}, and τ_{yy}. The preconditioned conservative form (II.14) was proposed in Ref.13, mainly for the purpose of simplifying the construction of numerical schemes; it may be extended to further accelerate the steady convergence by the local-preconditioning technique.[18,19] The wave structure of the system can be analyzed by the Jacobian matrix projected along an arbitrary vector, (n_x, n_y):

$$\mathbf{PA}_n \equiv \mathbf{P}\left(\frac{\partial \mathbf{F}}{\partial \mathbf{U}} n_x + \frac{\partial \mathbf{G}}{\partial \mathbf{U}} n_y\right), \tag{II.17}$$

which is split into the inviscid and viscous parts:

$$\mathbf{PA}_n = \mathbf{PA}_n^i + \mathbf{PA}_n^v, \tag{II.18}$$

or, equivalently, since $\mathbf{P}$ has no effect on the inviscid Jacobian,

$$\mathbf{PA}_n = \mathbf{A}_n^i + \mathbf{PA}_n^v, \tag{II.19}$$

where $\mathbf{A}_n^i$ and $\mathbf{A}_n^v$ are the projected Jacobians derived from the inviscid and viscous fluxes, respectively. The viscous Jacobian has real eigenvalues and linearly independent eigenvectors, and therefore hyperbolic as already shown in Ref.13, and can be discretized by methods for hyperbolic systems. In this work, we construct an upwind flux for the viscous part and simply add it to an inviscid numerical flux, following the simplified approach first introduced in Ref.13 for the hyperbolic NS system, and then studied in Ref.16 for the model advection-diffusion equation. Note that the hyperbolic viscous system has six nonzero eigenvalues associated with three sets of symmetric waves, and three zero eigenvalues: one originates from the fact that the density is not affected by the viscous terms, and the other two associated with the constraints:

$$\partial_{xx}\left(\frac{\tau_{yy}}{\mu_v}\right) + \frac{1}{2}\partial_{xx}\left(\frac{\tau_{xx}}{\mu_v}\right) + \partial_{yy}\left(\frac{\tau_{xx}}{\mu_v}\right) + \frac{1}{2}\partial_{yy}\left(\frac{\tau_{yy}}{\mu_v}\right) - \partial_{xy}\left(\frac{\tau_{xy}}{\mu_v}\right) = 0 \tag{II.20}$$

$$\partial_x\left(\frac{q_y}{\mu_h}\right) - \partial_y\left(\frac{q_x}{\mu_h}\right) = 0, \tag{II.21}$$

which must be satisfied in the steady state. These modes associated with a zero eigenvalue are called the inconsistency damping modes because errors associated with these constraints are purely damped.[1] However, as will be discussed later, these zero eigenvalues can be made nonzero by the divergence formulation of source terms.[15]

III. Hyperbolic Incompressible NS System

Consider the incompressible NS equations in the pseudo-compressible form:[20]

$$\partial_t P + \operatorname{div}(a^{*2}\mathbf{v}) = 0, \tag{III.1}$$

$$\partial_t \mathbf{v} + \operatorname{div}(\mathbf{v}\otimes\mathbf{v} + P\mathbf{I}) = \operatorname{div}(\nu \operatorname{grad} \mathbf{v}), \tag{III.2}$$

where P is the kinematic pressure defined by $P = p/\rho$, and a^* is the artificial speed of sound,[20–24] which is taken to be 10 for all test cases in this study. We assume that the kinematic viscosity is constant and that the velocity and the kinematic pressure have been nondimensionalized by the free stream speed and its square, respectively, thus resulting in $\nu = 1/Re_\infty$. Note that, as is well known, the pseudo-compressible form is equivalent to the incompressible NS equations only in the steady state. While preserving the equivalence in the steady state, we can hyperbolize the viscous terms as follows:

$$\partial_t P + \operatorname{div}(a^{*2}\mathbf{v}) = 0, \tag{III.3}$$

$$\partial_t \mathbf{v} + \operatorname{div}(\mathbf{v}\otimes\mathbf{v} + P\mathbf{I}) = \operatorname{div}(\nu\mathbf{g}), \tag{III.4}$$

$$T_r \partial_t \mathbf{g} = \operatorname{grad}\mathbf{v} - \mathbf{g}, \tag{III.5}$$

where T_r is a relaxation time defined by

$$T_r = \frac{L^2}{\nu}. \tag{III.6}$$

Observe that $\mathbf{g} = \operatorname{grad}\mathbf{v}$ in the steady state, and thus the system will reduce to the incompressible NS equations. In two dimensions, the system can be written as

$$\mathbf{P}^{-1}\partial_t \mathbf{U} + \partial_x \mathbf{F} + \partial_y \mathbf{G} = \mathbf{S}, \tag{III.7}$$

where

$$\mathbf{U} = \begin{bmatrix} P \\ u \\ v \\ g_{xx} \\ g_{xy} \\ g_{yx} \\ g_{yy} \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} a^* u \\ 2u^2 + P \\ vu \\ -u \\ 0 \\ -v \\ 0 \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} a^* v \\ uv \\ 2v^2 + P \\ 0 \\ -u \\ 0 \\ -v \end{bmatrix}, \tag{III.8}$$

$$\mathbf{S} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -g_{xx} \\ -g_{xy} \\ -g_{yx} \\ -g_{yy} \end{bmatrix}, \quad \mathbf{P}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & T_r & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & T_r & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & T_r & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & T_r \end{bmatrix}, \tag{III.9}$$

where g_{xx}, g_{xy}, g_{yx}, and g_{yy} are the components of $\mathbf{g}$,

$$\mathbf{g} = \begin{bmatrix} g_{xx} & g_{xy} \\ g_{yx} & g_{yy} \end{bmatrix}. \tag{III.10}$$

Note that we have in the steady state:

$$g_{xx} = \partial_x u, \quad g_{xy} = \partial_y u, \quad g_{yx} = \partial_x v, \quad g_{yy} = \partial_y v. \tag{III.11}$$

The normal flux Jacobian,

$$\mathbf{PA}_n \equiv \mathbf{P}\left(\frac{\partial \mathbf{F}}{\partial \mathbf{U}} n_x + \frac{\partial \mathbf{G}}{\partial \mathbf{U}} n_y\right), \tag{III.12}$$

has the following eigenvalues:

$$\lambda = u_n \pm \sqrt{u_n^2 + a^{*2} + a_v^2}, \quad \frac{1}{2}\left(u_n \pm \sqrt{u_n^2 + a_v^2}\right), \quad 0, \quad 0, \quad 0, \tag{III.13}$$

where $u_n = un_x + vn_y$ and $a_v = \sqrt{\nu/T_r}$. Zero eigenvalues represent the inconsistency damping modes associated with the constraints on the extra variables,

$$\partial_x\left(g_{xy}\right) - \partial_y\left(g_{xx}\right) = 0, \quad \partial_x\left(g_{yy}\right) - \partial_y\left(g_{yx}\right) = 0, \tag{III.14}$$

and the fact that the continuity equation does not have a viscous term. Zero eigenvalues of the constraints can be made nonzero by the divergence formulation of source terms.[15] The corresponding eigenvectors can be shown to be linearly independent. Therefore, the hyperbolic incompressible NS system is hyperbolic. We remark that the system has been written as a preconditioned form in order to allow the length scale L to vary in space.

In contrast to the compressible version, the full eigen-structure is available for the incompressible version, and thus a unified construction of numerical schemes as in Ref.12 is possible. However, in this paper, we consider the simplified approach of treating the inviscid and viscous parts separately. For the inviscid flux, we employ a widely-used upwind flux.[20–24] We then add to it an upwind viscous flux constructed based on the eigen-structure of the hyperbolic viscous terms only, which is essentially identical to that constructed for diffusion in the previous work[14,16] because the viscous terms are Laplacians. Moreover, all the techniques developed in the previous work, such as the fully hyperbolic formulation in Ref.16, directly apply to the hyperbolic viscous terms in the incompressible case.

IV. Node-Centered Edge-Based Finite-Volume Scheme

IV.A. Discretization

The node-centered edge-based finite-volume scheme for the hyperbolic Navier-Stokes system is given by

$$V_j \frac{d\mathbf{U}_j}{dt} = -\mathbf{P}_j\left(\sum_{k\in\{k_j\}} \mathbf{\Phi}_{jk} A_{jk} - \mathbf{S}_j V_j\right), \tag{IV.1}$$

where $\mathbf{P}_j$ is the preconditioning matrix evaluated at node j, V_j is the measure of the dual control volume around the node j, $\{k_j\}$ is a set of neighbors of j, $\mathbf{\Phi}_{jk}$ is a numerical flux, and A_{jk} is the magnitude of the directed area vector, i.e., $A_{jk} = |\mathbf{n}_{jk}| = |\mathbf{n}_{jk}^{\ell} + \mathbf{n}_{jk}^{r}|$ (see Figure 1). This formulation is valid for triangular, quadrilateral, or mixed grids, and all schemes developed below can be directly applied to any grid except the third-order scheme, which is third-order accurate only on triangular grids. For the third-order scheme, a straightforward point-source integration in Equation (IV.1) leads to accuracy deterioration; the source terms need to be discretized carefully to preserve the accuracy as will be discussed later. Note also that an appropriate boundary flux must be supplied at the boundary node. For first-order schemes, a point evaluation is sufficiently accurate, but for second-order schemes, a special quadrature is required for the linear exactness in the flux integration. See Appendix E in Ref.25 for a comprehensive list of linearity preserving boundary quadrature formulas in two and three dimensions. For the third-order scheme, we will employ a general formula recently derived in Ref.26 that preserves quadratic fluxes to achieve third-order accuracy through boundary nodes.

IV.B. Numerical Flux

We define the numerical flux as an upwind flux in the form:

$$\mathbf{\Phi}_{jk} = \frac{1}{2}\left[\mathbf{H}_R + \mathbf{H}_L\right] - \frac{1}{2}\mathbf{P}^{-1}\left|\mathbf{PA}_n\right|\left(\mathbf{U}_R - \mathbf{U}_L\right), \tag{IV.2}$$

where the subscript L and R indicate the values at the left and right of the edge midpoint, and $\mathbf{H}$ is the physical flux projected along the directed area vector,

$$\mathbf{H} = [\mathbf{F}, \mathbf{G}] \cdot \hat{\mathbf{n}}_{jk}, \quad \hat{\mathbf{n}}_{jk} = \frac{\mathbf{n}_{jk}}{|\mathbf{n}_{jk}|}, \tag{IV.3}$$

and $\mathbf{A}_n = \partial\mathbf{H}/\partial\mathbf{U}$. Note that the dissipation term has been constructed following the well-known procedure in the local preconditioning technique [27]. The evaluation of the absolute Jacobian, $|\mathbf{PA}|$, requires the complete eigen-structure of the full Jacobian that is still not known for the hyperbolic compressible NS system. But it can be approximated by

$$|\mathbf{PA}_n| \approx |\mathbf{PA}_n^i| + |\mathbf{PA}_n^v| = \sum_{k=1}^{m^i} |\lambda_k^i| \mathbf{\Pi}_k^i + \sum_{k=1}^{m^v} |\lambda_k^v| \mathbf{\Pi}_k^v, \tag{IV.4}$$

where λ_k^i and $\mathbf{\Pi}_k^i$ denote the k-th eigenvalue and projection matrix of the inviscid Jacobian, respectively, and similarly λ_k^v and $\mathbf{\Pi}_k^v$ for the viscous part (see Ref.13). The number of nonzero wave components for the inviscid and viscous parts, m^i and m^v, are respectively 4 and 6 for the hyperbolic compressible NS equations, and 3 and 4 for the hyperbolic incompressible NS equations. Note that the approximate construction of the dissipation term is a standard practice in many discretization methods; the numerical flux constructed as above is nothing but the sum of the upwind inviscid flux and the upwind viscous flux:

$$\mathbf{\Phi}_{jk} = \mathbf{\Phi}_{jk}^i + \mathbf{\Phi}_{jk}^v, \tag{IV.5}$$

where $\mathbf{\Phi}_{jk}^i$ and $\mathbf{\Phi}_{jk}^v$ denote the upwind inviscid and viscous fluxes,

$$\mathbf{\Phi}_{jk}^i = \frac{1}{2}\left[\mathbf{H}_R^i + \mathbf{H}_L^i\right] - \frac{1}{2}\left|\mathbf{A}_n^i\right|\left(\mathbf{U}_R - \mathbf{U}_L\right), \tag{IV.6}$$

$$\mathbf{\Phi}_{jk}^v = \frac{1}{2}\left[\mathbf{H}_R^v + \mathbf{H}_L^v\right] - \frac{1}{2}\mathbf{P}^{-1}\left|\mathbf{PA}_n^v\right|\left(\mathbf{U}_R - \mathbf{U}_L\right), \tag{IV.7}$$

and $\mathbf{H}^i$ and $\mathbf{H}^v$ are the inviscid and viscous parts of the physical flux, respectively. Hence, if one wishes, the inviscid flux can be replaced by any other flux (e.g., Rotated-RHLL,[28] HLLC,[29] etc). It also implies that the hyperbolic scheme can be implemented into an existing NS solver simply by replacing the existing viscous flux by the upwind viscous flux (IV.7).

The interface quantities needed to evaluate the dissipation matrix are computed by the Roe-averages[30] in the inviscid part for the compressible case, and otherwise by the arithmetic averages. We remark that the viscous part of the dissipation term can be simplified and thus can be implemented as a single vector, and also that face-tangent vectors arising in the projection matrices can be eliminated completely (see Ref.17). The left and right fluxes and solutions are defined at the edge midpoint and evaluated by the nodal values for first-order accuracy and by the linear extrapolation from the nodes for second/third-order accuracy.

It is emphasized that the dissipation term is of critical importance in the hyperbolic method. Without the dissipation term, the discrete equations may be decoupled and the variables such as the viscous stresses and heat fluxes would be determined explicitly from the primitive variables. Consequently, the order of accuracy would be one order lower than that of the primitive variables on irregular grids, and furthermore no acceleration in convergence will be achieved. This problem is discussed for the diffusion equation in Ref.14.

IV.C. Implicit Solver

Steady-state solutions can be obtained by marching in time towards the steady state as demonstrated in the previous papers.[13,14] In this paper, following the work in Ref.14, we drop the time derivative term,

$$0 = -\mathbf{P}_j \left(\sum_{k\in\{k_j\}} \mathbf{\Phi}_{jk} A_{jk} - \mathbf{S}_j V_j \right), \tag{IV.8}$$

and construct an implicit solver for the global system of steady residual equations. The advantage of $O(1/h)$ acceleration in the steady convergence over traditional methods, which has been observed for explicit pseudo-time-marching schemes,[1,12–14] now comes in the iterative convergence of the linear system arising from the implicit formulation as demonstrated in Ref.16. Note also that the residual (IV.8) is consistent with the original Navier-Stokes equations, and therefore we will be solving the Navier-Stokes equations consistently, not any approximated equations.

Consider the global system of residual equations, which consists of rows of the nodal residual (IV.8):

$$0 = \mathbf{Res}(\mathbf{U}_h), \tag{IV.9}$$

where $\mathbf{U}_h$ denotes the global solution vector for which the system is to be solved. We consider the iterative method in the form:

$$\mathbf{U}_h^{n+1} = \mathbf{U}_h^n + \Delta\mathbf{U}_h, \tag{IV.10}$$

where the correction $\Delta\mathbf{U}_h$ is the solution to the following linear system:

$$\frac{\partial \mathbf{Res}}{\partial \mathbf{U}_h}\Delta\mathbf{U}_h = -\mathbf{Res}(\mathbf{U}_h^n). \tag{IV.11}$$

The Jacobian matrix is constructed by analytically differentiating the residual of the first-order scheme for all schemes. Therefore, the method is Newton's method for the first-order scheme, and a defect correction method for the second- and third-order schemes, provided the linear system is fully solved. In practice, we do not fully solve but only relax the linear system. In this work, for simplicity, we employ the sequential Gauss-Seidel (GS) relaxation to relax the linear system to a specified tolerance. It is possible to add a pseudo-time term to the left hand side, but it is not used in this work except for the initial first-order iterations as described later. Through numerical experiments, we encountered a significant slow-down in the linear relaxation for high-aspect ratio grids. An analysis shows that the problem is associated with the source term, and suggests that the following modification in the length scale improves the convergence:

$$L = \frac{1}{2\pi\mathcal{AR}}, \tag{IV.12}$$

where $\mathcal{AR}$ is the local cell-aspect-ratio. The aspect ratio is defined in a triangle as the ratio of the longest side to the height. At a node, it is defined as the average of the cell-aspect-ratio sharing the node, and at an edge is defined as the maximum of the two stored at the two nodes of the edge. A detailed study on the effects of high aspect ratio grids will be reported elsewhere. Notice that the length scale now varies in space and thus T_r is not a global constant; this is the reason that the preconditioned form is used in the hyperbolic incompressible NS system.

Note that the condition number of the Jacobian is $O(1/h)$ for the hyperbolic NS system because there exist no second derivatives, i.e., no $O(1/h^2)$ terms, even in the viscous limit, implying $O(1/h)$ acceleration in iterative convergence over traditional schemes for viscous dominated problems. The $O(1/h)$ convergence acceleration has been observed for many problems in the previous papers.[1,12–14,16]

IV.D. First-Order Scheme

We construct the first-order scheme by evaluating the left and right states by the nodal solutions:

$$\mathbf{U}_L = \mathbf{U}_j, \qquad \mathbf{U}_R = \mathbf{U}_k, \tag{IV.13}$$

and the numerical flux by the upwind flux (IV.2). The source term is discretized by a point integration of Equation (IV.1). The resulting scheme corresponds to Scheme I in Ref.14. The Jacobian is constructed exactly based on this scheme.

IV.E. Second-Order Scheme

For second-order accuracy, we compute the nodal gradient by a linear least-squares (LSQ) method, and evaluate the left and right states by the linear extrapolation from the nodes:

$$\mathbf{U}_L = \mathbf{U}_j + \frac{1}{2}\nabla\mathbf{U}_j \cdot \Delta\mathbf{l}_{jk}, \qquad \mathbf{U}_R = \mathbf{U}_k - \frac{1}{2}\nabla\mathbf{U}_k \cdot \Delta\mathbf{l}_{jk}, \tag{IV.14}$$

where $\Delta\mathbf{l}_{jk} = (x_k - x_j, y_k - y_j)$, $\nabla\mathbf{U}_j$ is the gradient of $\mathbf{U}$ computed by the LSQ method at j, and similarly for $\nabla\mathbf{U}_k$. The numerical flux is computed by the upwind flux (IV.2). The source term is, again, discretized by a point integration of Equation (IV.1). As in the first-order scheme, the resulting scheme belongs to Scheme I in Ref.14.

IV.F. Third-Order Scheme

As in the previous work, we consider the third-order edge-based finite-volume scheme discovered by Katz and Sankaran.[31] It is a special and economical node-centered scheme for hyperbolic conservation laws: the second-order node-centered edge-based finite-volume scheme achieves third-order accuracy on triangular grids if the nodal gradients are exact for quadratic functions, and in the case of nonlinear fluxes, if the flux is linearly extrapolated to the edge-midpoint. The third-order accuracy has been demonstrated for the advection equation and the Euler equations on regular as well as irregular triangular grids in Refs.31–33. The extension to the NS equations is straightforward if the viscous terms are written as a hyperbolic system. We simply discretize the viscous terms just like the inviscid terms. The required nodal gradients are computed by a quadratic LSQ fit. In this study, we employ the two-step implementation as described in Ref.16. For the source term, special discretization formulas are required to achieve third-order accuracy. In this work, we employ the divergence formulation of source terms as proposed in Ref.15:

$$\mathbf{S} = \partial_x \mathbf{F}^s + \partial_y \mathbf{G}^s, \tag{IV.15}$$

where

$$\mathbf{F}^s = (x - x_j)\mathbf{S} - \frac{1}{2}(x - x_j)^2 \partial_x \mathbf{S} + \frac{1}{6}(x - x_j)^3 \partial_{xx}\mathbf{S}, \tag{IV.16}$$

$$\mathbf{G}^s = 0, \tag{IV.17}$$

where x_j is the x-coordinate of the node at which the residual is defined. Hence, the hyperbolic NS system can be written as a hyperbolic conservation law without source terms:

$$\mathbf{P}^{-1}\partial_t \mathbf{U} + \partial_x \left(\mathbf{F} - \mathbf{F}^s\right) + \partial_y \left(\mathbf{G} - \mathbf{G}^s\right) = 0. \tag{IV.18}$$

In the hyperbolic incompressible NS system, the derivatives of the source terms are already available as computed in the reconstruction step, but they need to be computed for the hyperbolic compressible NS system by a quadratic LSQ fit. Note that the resulting source divergence form is a hyperbolic system, and it is discretized in the same way as the inviscid and viscous terms. As a result, the residual at node j is now given by

$$0 = -\mathbf{P}_j \left(\sum_{k \in \{k_j\}} \mathbf{\Phi}_{jk} A_{jk} \right), \tag{IV.19}$$

where the numerical flux is now defined as a sum of the inviscid, viscous, and source-divergence fluxes:

$$\mathbf{\Phi}_{jk} = \mathbf{\Phi}^i_{jk} + \mathbf{\Phi}^v_{jk} - \mathbf{\Phi}^s_{jk}. \tag{IV.20}$$

Here, $\mathbf{\Phi}^s_{jk}$ denotes the upwind source-divergence flux given by

$$\mathbf{\Phi}^s_{jk} = \frac{1}{2}\left[\mathbf{H}^s_R + \mathbf{H}^s_L\right] - \frac{1}{2}\mathbf{P}^{-1}\left|\mathbf{P}\mathbf{A}^s_n\right|\left(\mathbf{U}_R - \mathbf{U}_L\right), \tag{IV.21}$$

where

$$\mathbf{H}^s = \left[\mathbf{F}^s, \mathbf{G}^s\right] \cdot \hat{\mathbf{n}}_{jk}, \quad \mathbf{A}^s_n = \frac{\partial \mathbf{F}^s}{\partial \mathbf{U}} n_x + \frac{\partial \mathbf{G}^s}{\partial \mathbf{U}} n_y. \tag{IV.22}$$

The flux Jacobian $\mathbf{A}^s_n$ is constructed based only on the leading terms of the source fluxes and the viscosity is frozen for simplicity, thus making the Jacobian diagonal and consequently making the construction of the upwind flux very simple. It is very important to note that the above source flux is valid only for the node j, not valid for the neighbor node k. It must be re-computed for the node k with the divergence form defined at k. Hence, the source discretization is not conservative, and it is, in fact, consistent with the Galerkin discretization of the source term. See Ref.15 for details. For the third-order scheme, the Jacobian matrix in the implicit solver is constructed exactly (except for the simplification mentioned above for the source terms) based on the above residual with the nodal gradients ignored.

V. Boundary Conditions

Boundary conditions are implemented weakly as described in Ref.34 through the numerical flux (IV.2) evaluated at a boundary node, j, for the two states:

$$\mathbf{U}_L = \mathbf{U}_j, \quad \mathbf{U}_R = \mathbf{U}_b, \tag{V.1}$$

where $\mathbf{U}_j$ is the solution at the boundary node j, and $\mathbf{U}_b$ is the state determined by boundary conditions. For example, at an outflow boundary far enough from a body, $\mathbf{U}_b$ is determined from the primitive variables at j with the pressure replaced by the free stream pressure (the so-called back pressure condition). The flux at a boundary node across a boundary edge is then determined by the upwind flux, which recognizes the local characteristic directions and takes appropriate states for the flux computation. The boundary fluxes are incorporated into a suitable quadrature formula as in Refs.25, 26 to close the residual at the boundary node. In this manner, the residual is computed for all equations at all boundary nodes. Strong boundary conditions are applied after the residual is computed.

At a solid wall, we impose the no-slip condition strongly at a boundary node j by replacing the momentum residuals by the algebraic equations:

$$u_j = 0, \quad v_j = 0. \tag{V.2}$$

For the hyperbolic incompressible NS schemes, we also strongly impose the zero velocity gradients along the solid wall: $(g_{xx}, g_{xy})_j \cdot \mathbf{t}_j = 0$ and $(g_{yx}, g_{yy})_j \cdot \mathbf{t}_j = 0$, where $\mathbf{t}_j$ denotes a tangent vector defined at boundary node j. The normal velocity gradients, $(g_{xx}, g_{xy})_j \cdot \mathbf{n}_j$ and $(g_{yx}, g_{yy})_j \cdot \mathbf{n}_j$, where $\mathbf{n}_j$ denotes the wall normal vector at boundary node j, can then be computed by solving their residual equations projected along the wall normal direction. For the hyperbolic compressible NS schemes, the viscous stresses are determined by solving their residual equations at boundary nodes, but the adiabatic condition is imposed strongly by $(q_x, q_y)_j \cdot \mathbf{n}_j = 0$, and the tangential component is determined by solving its residual equation. Therefore, the Nuemann boundary condition can be imposed as the Dirichlet condition through the extra variables in the hyperbolic method, which is a great benefit especially on unstructured grids. If the wall temperature is specified and it is constant, then we impose zero tangential heat flux strongly, $(q_x, q_y)_j \cdot \mathbf{t}_j = 0$, and determine the normal heat flux by solving its residual equation.

VI. Computation of Velocity Gradients from Viscous Stresses

Accurate vorticity prediction is of critical importance for various applications: e.g., unsteady vortical flow simulations; accurate evaluation of source terms in turbulence model equations. For the incompressible NS equations, the hyperbolic method yields high-order accurate velocity gradients, which can be directly used to obtain the vorticity as well as the viscous stresses to the same order of accuracy. For the compressible NS equations, on the other hand, the hyperbolic schemes considered in this study yield high-order accurate viscous stresses; it is not straightforward to convert them into the velocity gradients. In two dimensions, we have

$$\frac{\tau_{xx}}{\mu_v} = \partial_x u - \frac{1}{2}\partial_y v, \quad \frac{\tau_{xy}}{\mu_v} = \frac{3}{4}(\partial_y u + \partial_x v), \quad \frac{\tau_{yy}}{\mu_v} = \partial_y v - \frac{1}{2}\partial_x u, \tag{VI.1}$$

which are three equations and thus cannot be solved for the four unknowns, $(\partial_x u, \partial_y u, \partial_x v, \partial_y v)$. Note, however, that the first and third equations can be solved for $\partial_x u$ and $\partial_y v$.

$$\partial_x u = \frac{4}{3}\left(\frac{\tau_{xx}}{\mu_v}\right) + \frac{2}{3}\left(\frac{\tau_{yy}}{\mu_v}\right), \quad \partial_y v = \frac{4}{3}\left(\frac{\tau_{yy}}{\mu_v}\right) + \frac{2}{3}\left(\frac{\tau_{xx}}{\mu_v}\right). \tag{VI.2}$$

Therefore, these velocity gradient components can be obtained to the same order of accuracy as the viscous stresses, and thus as the primitive variables. The remaining components, $\partial_y u$ and $\partial_x v$, cannot be obtained

directly from the viscous stress, $\frac{\tau_{xy}}{\mu_v}$. However, by differentiating Equations (VI.1), we obtain

$$\partial_x\left(\frac{\tau_{xx}}{\mu_v}\right) = \partial_{xx}u - \frac{1}{2}\partial_{xy}v, \quad \partial_y\left(\frac{\tau_{xx}}{\mu_v}\right) = \partial_{xy}u - \frac{1}{2}\partial_{yy}v, \tag{VI.3}$$

$$\partial_x\left(\frac{\tau_{xy}}{\mu_v}\right) = \frac{3}{4}(\partial_{xy}u + \partial_{xx}v), \quad \partial_y\left(\frac{\tau_{xy}}{\mu_v}\right) = \frac{3}{4}(\partial_{yy}u + \partial_{xy}v), \tag{VI.4}$$

$$\partial_x\left(\frac{\tau_{yy}}{\mu_v}\right) = \partial_{xy}v - \frac{1}{2}\partial_{xx}u, \quad \partial_y\left(\frac{\tau_{yy}}{\mu_v}\right) = \partial_{yy}v - \frac{1}{2}\partial_{xy}u, \tag{VI.5}$$

which are now six equations and can actually be solved for the six components of the second derivatives of u and v:

$$\partial_{xx}u = \frac{4}{3}\partial_x\left(\frac{\tau_{xx}}{\mu_v}\right) + \frac{2}{3}\partial_x\left(\frac{\tau_{yy}}{\mu_v}\right), \quad \partial_{xy}u = \frac{4}{3}\partial_y\left(\frac{\tau_{xx}}{\mu_v}\right) + \frac{2}{3}\partial_y\left(\frac{\tau_{yy}}{\mu_v}\right), \tag{VI.6}$$

$$\partial_{yy}u = -\frac{2}{3}\partial_x\left(\frac{\tau_{xx}}{\mu_v}\right) + \frac{4}{3}\partial_y\left(\frac{\tau_{xy}}{\mu_v}\right) - \frac{4}{3}\partial_x\left(\frac{\tau_{yy}}{\mu_v}\right), \quad \partial_{xx}v = -\frac{2}{3}\partial_y\left(\frac{\tau_{xx}}{\mu_v}\right) + \frac{4}{3}\partial_x\left(\frac{\tau_{xy}}{\mu_v}\right) - \frac{4}{3}\partial_y\left(\frac{\tau_{yy}}{\mu_v}\right), \tag{VI.7}$$

$$\partial_{xy}v = \frac{2}{3}\partial_x\left(\frac{\tau_{xx}}{\mu_v}\right) + \frac{4}{3}\partial_x\left(\frac{\tau_{yy}}{\mu_v}\right), \quad \partial_{yy}v = \frac{2}{3}\partial_y\left(\frac{\tau_{xx}}{\mu_v}\right) + \frac{4}{3}\partial_y\left(\frac{\tau_{yy}}{\mu_v}\right). \tag{VI.8}$$

Therefore, the second derivatives can be directly computed from the gradients of the viscous stresses, which are available as a part of the finite-volume discretization: first-order accurate for the second-order scheme, and second-order accurate for the third-order scheme. For the second-order scheme, it is then possible to perform a quadratic gradient reconstruction in a linear LSQ stencil with the curvature term evaluated by these second derivatives. For the third-order scheme, similarly, we can compute the third derivatives (first-order accurate) from these second-derivatives by a linear or quadratic LSQ fit, and then perform a cubic gradient reconstruction in a quadratic LSQ stencil. In this work, therefore, we employ this method for computing the remaining components $\partial_y u$ and $\partial_x v$. As will be shown later, the velocity gradients computed as described are very accurate, but they are not second- and third-order but first- and second-order accurate for the second- and third-order schemes, respectively. If not impossible, it is generally very difficult to compute the velocity and the velocity gradients to the same order of accuracy on irregular grids by such a reconstruction algorithm: compute derivatives from a numerical solution. For the first-order scheme, the derivatives of the viscous stresses are not available, and therefore it is not possible to obtain the second derivatives of the velocity. Note that the LSQ gradient reconstruction will not work for first-order accurate solutions on irregular grids: the LSQ gradients will be inconsistent unless some special algorithm is available that recovers the gradients with first-order accuracy from first-order accurate solutions. Note also that the inconsistent gradients are common to all first-order schemes (not just the hyperbolic schemes) on irregular grids for any equation. Thus, it does not make sense in general to develop a first-order scheme for the NS equations in the sense that the quantities of interest in viscous simulations, i.e., the viscous stresses and heat fluxes, are derivatives and thus cannot be obtained consistently. On the other hand, the first-order hyperbolic NS scheme is capable of producing first-order accurate solutions and viscous and heat fluxes simultaneously.

We remark that the procedure described above is not applicable to the Navier-Stokes equations in three dimensions. In fact, in three dimensions, neither velocity gradient components nor the second derivatives can be obtained from the viscous stresses as described above under Stokes' hypothesis. In order to compute the velocity gradients to the same order of accuracy or compute them with upgraded LSQ fits, other means need to be developed for the compressible NS equations. One possible approach is to construct a hyperbolic NS system by using the velocity gradients instead of the viscous stresses. We have actually tried it, but found interestingly that the order of accuracy is still one order lower for $\partial_y u$ and $\partial_x v$ although the viscous stress $\tau_{xy} = \mu(\partial_y u + \partial_x v)$, which is computed directly from the low-order gradients, is second- and third-order accurate for the second- and third-order schemes.

VII. Numerical Results

The first-, second-, and third-order hyperbolic NS schemes have been tested for a set of steady laminar test cases, and compared with the results obtained by a conventional scheme. The conventional scheme is a second-

order node-centered edge-based finite-volume scheme based on the same upwind flux for the inviscid terms and the alpha-damping flux for the viscous terms.[25,35] The parameter α in the alpha-damping scheme has been set to be 4/3; it is known to yield fourth-order accuracy for diffusion on regular quadrilateral grids and also yields very accurate solutions even on highly-skewed grids (see Refs.25, 35). An implicit solver is constructed also for the conventional scheme. The inviscid Jacobian is again exact for the first-order inviscid scheme, but the viscous Jacobian is based on the inconsistent edge-terms-only scheme[36] because a consistent and compact version is not available for the alpha-damping scheme. The conventional scheme is referred to as Alpha4/3 in the rest of the section. In all computations, the Jacobian is re-computed at every iteration until the residuals are reduced by two orders of magnitude, and thereafter once at every order of magnitude reduction.

VII.A. Accuracy Verification

Accuracy of the developed numerical schemes has been verified by method of manufactured solutions. A smooth sine function is set as the exact solution by introducing the source term into the Navier-Stokes equations, with $M_\infty = 0.3$, $Pr = 3/4$, $\gamma = 1.4$, $Re_\infty = 50$, and $T_\infty = 300$ [k]. The discretization error is measured on a series of irregular triangular grids (1024, 2304, 4096, 6400, 9216, 12544, 16384, 20736, 25600 nodes) in a square domain with the Dirichlet condition. For all cases, the discrete problem is fully solved in the sense that the residuals have been reduced by ten orders of magnitude (or to a machine zero). Results for the compressible NS equations are shown in Figures 2-14. As clearly shown in these figures, first-, second-, and third-order accuracy of the hyperbolic Navier-Stokes schemes have been confirmed for all variables, $(\rho, u, v, p, \tau_{xx}, \tau_{xy}, \tau_{yy}, q_x, q_y)$. As expected, Alpha4/3 gives second-order accuracy for (ρ, u, v, p), but only first-order accuracy in the derivative quantities: $(\tau_{xx}, \tau_{xy}, \tau_{yy}, q_x, q_y)$. Shown in Figures 11-14 are the results for the velocity gradients. For Alpha4/3 and the first-order hyperbolic Navier-Stokes scheme, the velocity gradients were computed by a linear LSQ fit. As expected, they are first-order accurate for Alpha4/3. For the hyperbolic Navier-Stokes scheme, $\partial_x u$ and $\partial_y v$ have been computed with first-, second-, and third-order accuracy for the first-, second-, and third-order schemes as expected. However, the remaining components, $\partial_y u$ and $\partial_x v$, are obtained with one order lower accuracy although the error level is significantly low compared with that of Alpha4/3. Results for the incompressible NS equations are shown in Figures 15-21. As shown, first-, second-, and third-order accuracy has been confirmed for the hyperbolic Navier-Stokes schemes for all variables, $(p, u, v, \partial_x u = g_{xx}, \partial_y u = g_{xy}, \partial_x v = g_{yx}, \partial_y v = g_{yy})$. On the other hand, Alpha4/3 gives second-order accuracy in (p, u, v), but only first-order accuracy in the velocity gradients. Finally, we notice that the hyperbolic schemes yield higher errors in the velocity components compared with the conventional scheme in both the incompressible and compressible cases. This tendency has been observed in the previous work for Scheme I, and it is known that Scheme II gives lower errors (see Ref.14). Future work should include the implementation of Scheme II for the hyperbolic NS schemes.

VII.B. Bump

We consider a subsonic laminar flow over a bump with $M_\infty = 0.2$, $Pr = 3/4$, $\gamma = 1.4$, $Re_\infty = 100$, and $T_\infty = 300$ [k]. The geometry and boundary conditions are similar to those described in Ref.[37] Three levels of highly-skewed irregular triangular grids are used: Grid1, Grid2, and Grid3, with the number of nodes, $N = 6400$, 25600, 102400, respectively. See Figures 22 and 23. For this problem, the GS linear relaxation is performed until the linear residual is reduced by one order of magnitude. The initial solution is set by the free stream condition, and the first-order scheme is used for the first 15 iterations or until the first-order residual is reduced by two orders of magnitude in order to provide a reasonable initial soltuion for the second- and third-order schemes. The first-order scheme converges very rapidly since the solver is Newton's method. Similarly for the conventional scheme, the first-order inviscid and inconsistent viscous schemes are used to generate an initial solution for the second-order scheme. During the initial low-order iteration, the pseudo-time term is incorporated with the CFL number starting from 10 and increased by a factor of 10 at every iteration. The second- and third-order solvers are taken to be converged when the residuals are reduced by six orders of magnitude in L_1 norm. The vorticity contours are shown in Figures 24-26. Observe that the smoothness improves consistently from Alpha4/3 to the second- and then third-order hyperbolic NS schemes. Note that the order of accuracy in the velocity gradients, and thus in the vorticity, is first-order for both Alpha4/3 and the second-order hyperbolic NS scheme. The skin friction coefficient distribution over the bump is compared in Figures 27 and 28. Alpha4/3 generates severe oscillations, which is expected for conventional schemes on highly-skewed grids, but the hyperbolic schemes yield highly smooth and accurate skin friction coefficients over the bump. Moreover, these high-quality results have been obtained at a *reduced* cost. As shown in Figures 29 and 30, the second- and third-order hyperbolic schemes converge faster in CPU time than the Alpha4/3 scheme for the finest grid. As repeatedly demonstrated through

the development of the hyperbolic method, the convergence acceleration comes from the elimination of second derivatives and a strong coupling among the extended set of variables. For implicit solvers, the acceleration is achieved mainly in the linear relaxation: $O(1/h)$ or equivalently $O(\sqrt{N})$ faster convergence. This feature is demonstrated in Figure 42, which results in $O(\sqrt{N})$ faster convergence in the total CPU time as shown in Figure 43. Note that the hyperbolic schemes are intrinsically faster than conventional schemes: even if slower on a given grid, it will overwhelm the conventional solver as the grid gets refined. Similar results have been obtained for the incompressible NS equations. See Figures 34-43. Note that the per-iteration cost is larger in the hyperbolic solvers than the conventional solver due to the increase in the number of equations: 9 instead of 4 in the compressible case, and 7 instead of 3 in the incompressible case. However, the convergence acceleration factor grows in the grid refinement, and therefore it will overwhelm the constant factor in the per-iteration cost, which has been demonstrated in the results presented.

VII.C. Cylinder

We consider a laminar flow over a cylinder of unit diameter with $M_\infty = 0.2$, $Pr = 3/4$, $\gamma = 1.4$, $Re_\infty = 40$, and $T_\infty = 300$ [k]. The outer boundary is located at the distance of 500 times the diameter. Free stream boundary condition is applied at inflow, and the back-pressure condition is applied at outflow. On the cylinder surface, the no-slip and adiabatic conditions are imposed. For this problem, a steady solution exists; a pair of wake vortices are created and they extend in the x direction by the distance approximately 2.2 times the diameter.[38] Three levels of irregular triangular grids are used: Grid1, Grid2, and Grid3, with the number of nodes, $N = 12800$, 51200, 204800, respectively. Grid1 is shown in Figure 44. For this problem, we fix the number of GS relaxations: 100 for all cases (or less if the tolerance is met before reaching 100). In practical CFD runs, the number of linear relaxations is often calibrated and fixed for a given problem to avoid an excessively large number of relaxations that might be required to meet a tolerance. This test will show that the hyperbolic solver still converges faster than the conventional solver in such a situation.

Pressure contours obtained by the conventional scheme and the second- and third-order hyperbolic NS schemes are shown in Figures 45, 46, and 47, respectively. As can be seen, the second-order solutions are comparable, and the third-order solution is superior in smoothness to the second-order results. Skin friction coefficient distribution over the cylinder is shown in Figures 48 and 49. Again, the conventional scheme gives severe oscillations while the hyperbolic schemes yield highly smooth distributions. The convergence results in Figure 50 demonstrate that the hyperbolic solvers take less number of iterations than the conventional solver for Grid 2 and Grid3, and they converge faster in CPU time as well as shown in Figure 51. Similar results have been obtained for the incompressible NS equations. See Figures 52 and 53 for the skin friction coefficient distributions, and Figures 54 and 55 for convergence results.

VIII. Remarks on Cost: Three Dimensional Case

The cost per iteration is higher in the hyperbolic solver than in the conventional solver because of the additional equations added to form a hyperbolic system. In two dimensions, the factor is about 9/4 in the compressible case, and 7/3 in the incompressible case. Note, however, that the factor is constant, and the growing acceleration factor of $O(1/h)$ given by the hyperbolic method will overwhelm eventually. The same is true in three dimensions: the factor in the cost per iteration is about 14/5 in the compressible case and 13/4 in the incompressible case, which is again constant, and will be beaten by the $O(1/h)$ convergence acceleration. Certainly, it will require more memory to run the hyperbolic solver than conventional solvers. In a parallel environment, it may require the number of cores to be reduced to secure enough memory. But then again the factor of speed-down due to the less number of cores will be constant even in a perfectly scaled code, and thus the hyperbolic solver may turn out to converge still faster by the overwhelming $O(1/h)$ convergence acceleration; even if not, it will be faster eventually for finer grids.

The developed hyperbolic finite-volume schemes are highly efficient in comparison with other high-order methods. For example, in the Discontinuous Galerkin (DG) methods, 3 additional degrees of freedom and equations are needed for each variable for second-order accuracy (P1 DG) in three dimensions. For the three-dimensional Navier-Stokes equations, the total number of equations will be 20 (50 for P2), which already exceeds the number of equations, 14, required for the second- and third-order hyperbolic schemes developed in this paper. In the Reconstructed DG (RDG) method,[39] a third-order P2 DG scheme can be constructed from the P1 DG scheme to achieve third-order accuracy. But again it exceeds the number of equations required for the third-order hyperbolic scheme, i.e., 20 versus 14. In continuous Galerkin methods or residual-distribution

methods, such as the SUPG scheme, third-order accuracy is obtained with a continuous P2 element, where additional degrees of freedom are stored at edge-midpoints. Additional degrees of freedom at the edge-midpoint is roughly five times the number of nodes on a tetrahedral grid. Solving five equations per degree of freedom is, therefore, virtually equivalent to $5 \times (5 + 1) = 30$ equations per node on P1 elements, which is about twice the number of equations required for the third-order hyperbolic scheme. Moreover, and in particular, the third-order hyperbolic scheme developed here offers additional advantages of $O(1/h)$ convergence acceleration and high-quality third-order derivatives over a conventional second-order edge-based finite-volume solver widely used in practical simulations. It remains to be demonstrated that other third-order schemes can provide such advantages.

IX. Concluding Remarks

In this paper, we presented first-, second-, and third-order finite-volume schemes for the incompressible and compressible Navier-Stokes equations, based on a hyperbolic formulation of the viscous terms. An implicit solver is constructed based on the exact linearization of the first-order scheme. The solver is Newton's method for the first-order scheme, and a defect-correction method for the second- and third-order schemes. Accuracy has been verified by the method of manufactured solutions on fully irregular triangular grids. The developed schemes have been applied to a laminar flow over a bump and another over a cylinder, and compared with a conventional second-order solver. Numerical results show that the hyperbolic solvers yield highly accurate derivatives, the vorticity, and the skin friction coefficient, on highly skewed irregular triangular grids, while the conventional scheme suffers from severe oscillations on such grids. It thus enables highly arbitrary viscous grid adaptation, where the current practice is to freeze the initial boundary layer grid to avoid introducing irregularity or excessive uniform refinement.[40] Moreover, it has been demonstrated that the hyperbolic solver not only yields highly accurate derivatives, but also converges faster in CPU time than the conventional second-order solver, even for the third-order scheme. This paper thus demonstrated that third-order accurate solutions can be obtained faster in CPU time than second-order accurate solutions on a given grid. Also, the paper demonstrated that a first-order scheme can be constructed for the Navier-Stokes equations with physically relevant quantities such as the viscous stresses and the heat fluxes obtained to the same order of accuracy on irregular grids. Such derivative quantities are typically inconsistent on irregular grids if reconstructed from first-order accurate solutions. The first-order scheme provides a consistent Jacobian for implicit solvers, which is more effective than inconsistent Jacobians typically emptied for viscous terms (because first-order viscous schemes are not available).

Future work will focus on further numerical tests and improvements. For the test problems considered, effects of Reynolds number may be studied, the grid convergence of the drag coefficient may be investigated, the unstructured results may be compared with structured grid results, and so on. Also, a flow over a flat plate should also be investigated where the exact solution is available for the incompressible flow. The developed schemes should also be tested to see if it still offers the advantages demonstrated in this paper for high-Reynolds-number turbulent flows. Furthermore, high-speed flows involving shock waves and heat transfer should also be investigated. For algorithmic improvements, Scheme II, which re-uses the extra variables in the solution reconstruction and found very beneficial for the model equations,[14] should be implemented, especially for improving the accuracy in the velocity components. Also, the fully hyperbolic formulation, which was successfully applied to the model advection-diffusion equation, should be considered for the viscous terms to avoid the computation of second-derivatives for the source terms. The development of these techniques, including a method for obtaining the velocity gradients from the viscous stresses, remains a challenge and may require the construction of a new hyperbolic system for the compressible NS equations. A new approach is currently being explored to enable high-order velocity gradients for the compressible NS equations. For turbulence computations, high-order velocity gradients for source terms are promising. Improvements in the solver are also expected, for example, by a Newton-Krylov method where the implicit solver developed in this study will serve as an effective preconditioner. Ultimately, the developed single-grid solver can be employed as a smoother for a more powerful multigrid solver. Also, a p-multigrid solver may be developed based on the three levels of first-, second-, and third-order solvers developed here. Once a robust and efficient steady solver is in place, it will be readily extended to time-dependent problems based on implicit time-stepping schemes and the divergence formulation of the physical time derivative following the work in Ref.11. Eventually, the developed schemes will be implemented into practical three-dimensional codes, which appears straightforward but remains to be demonstrated.

Acknowledgments

This work has been funded by the U.S. Army Research Office under the contract/grant number W911NF-12-1-0154. Support by Software CRADLE is greatly acknowledged.

References

[1]Nishikawa, H., "A First-Order System Approach for Diffusion Equation. I: Second Order Residual Distribution Schemes," *J. Comput. Phys.*, Vol. 227, 2007, pp. 315–352.

[2]Cattaneo, C., "A Form of Heat-Conduction Equations which Eliminates the Paradox of Instantaneous Propagation," *Ct. R. Acad. Sci., Paris*, Vol. 247, 1958, pp. 431–433.

[3]Lowrie, R. B. and Morel, J. E., "Methods for Hyperbolic Systems with Stiff Relaxation," *Int. J. Numer. Meth. Fluids*, Vol. 40, 2002, pp. 413–423.

[4]Gómez, H., Colominas, I., Navarrina, F., and Casteleiro, M., "A Discontinuous Galerkin Method for a Hyperbolic Model for Convection-Diffusion Problems in CFD," *Int. J. Numer. Meth. Eng.*, Vol. 71, 2007, pp. 1342–1364.

[5]H. Gómez, I. Colominas, F. N. and Casteleiro, M., "A hyperbolic model for convection-diffusion transport problems in CFD: Numerical analysis and applications," *Rev. R. Acad. Cien. Serie A. Mat.*, Vol. 102, No. 2, 2008, pp. 319–334.

[6]Girault, V. and Raviart, P. A., "An analysis of a mixed finite element method for the Navier-Stokes equations," *Numer. Math.*, Vol. 33, 1979, pp. 235–271.

[7]Heys, J., Lee, E., Manteuffel, T., and McCormick, S., "An alternative least-squares formulation of the Navier-Stokes equations with improved mass conservation," *J. Comput. Phys.*, Vol. 226, 2008, pp. 944–1006.

[8]Jiang, B.-N., *The Least-Squares Finite Element Method*, Springer, 1998.

[9]Montecinos, G. I., Mller, L. O., and Toro, E. F., "Hyperbolic Reformulation of a 1D Viscoelastic Blood flow model and ADER finite volume schemes," *J. Comput. Phys.*, Vol. 266, 2014, pp. 101–123.

[10]Nishikawa, H. and Roe, P. L., "On High-Order Fluctuation-Splitting Schemes for Navier-Stokes Equations," *Computational Fluid Dynamics 2004*, edited by C. Groth and D. W. Zingg, Springer-Verlag, 2004, pp. 799–804.

[11]Mazaheri, A. and Nishikawa, H., "First-Order Hyperbolic System Method for Time-Dependent Advection-Diffusion Problems," *NASA-TM–2014–218175*, March 2014.

[12]Nishikawa, H., "A First-Order System Approach for Diffusion Equation. II: Unification of Advection and Diffusion," *J. Comput. Phys.*, Vol. 229, 2010, pp. 3989–4016.

[13]Nishikawa, H., "New-Generation Hyperbolic Navier-Stokes Schemes: $O(1/h)$ Speed-Up and Accurate Viscous/Heat Fluxes," *Proc. of 20th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2011-3043, Honolulu, Hawaii, 2011.

[14]Nishikawa, H., "First-, Second-, and Third-Order Finite-Volume Schemes for Diffusion," *J. Comput. Phys.*, Vol. 256, 2014, pp. 791–805.

[15]Nishikawa, H., "Divergence Formulation of Source Term," *J. Comput. Phys.*, Vol. 231, 2012, pp. 6393–6400.

[16]Nishikawa, H., "First, Second, and Third Order Finite-Volume Schemes for Advection-Diffusion," *J. Comput. Phys.*, Vol. 273, 2014, pp. 287–309.

[17]Masatsuka, K., "I do like CFD, VOL.1, Second Edition," `http://www.cfdbooks.com`, 2013.

[18]Nishikawa, H., Roe, P., Suzuki, Y., and van Leer, B., "A General Theory of Local Preconditioning and Its Application to the 2D Ideal MHD Equations," *Proc. of 16th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2003-3704, Orlando, Florida, 2003.

[19]Turkel, E., "Preconditioning Techniques in Computational Fluid Dynamics," *Annu. Rev. Fluid Mech.*, Vol. 31, 1999, pp. 385–416.

[20]Chorin, A. J., "A Numerical Method for Solving Incompressible Viscous Flow Problems," *J. Comput. Phys.*, Vol. 2, 1967, pp. 12–26.

[21]Kwak, D., Chang, J. L. C., Shanks, S. P., and Chakravarthy, S. K., "A Three-Dimensional Incompressible Navier-Stokes Solver Using Primitive Variables," *AIAA J.*, Vol. 24, 1986, pp. 390–396.

[22]Neel, R. E., Godfrey, A. G., and McGrory, W. D., "Low-Speed, Time-Accurate Validation of GASP Version 4," *Proc. of 43rd AIAA Aerospace Sciences Meeting and Exhibit*, AIAA Paper 2005-686, Reno, Nevada, 2005.

[23]Rogers, S. E. and Kwak, D., "Upwind Differencing Scheme for the Time-Accurate Incompressible Navier-Stokes Equations," *AIAA J.*, Vol. 28, No. 2, 1990, pp. 253–262.

[24]Rogers, S. E., Kwak, D., and Kiris, C., "Steady and Unsteady Solutions of the Incompressible Navier-Stokes Equations," *AIAA J.*, Vol. 29, No. 4, 1991, pp. 603–610.

[25]Nishikawa, H., "Beyond Interface Gradient: A General Principle for Constructing Diffusion Schemes," *Proc. of 40th AIAA Fluid Dynamics Conference and Exhibit*, AIAA Paper 2010-5093, Chicago, 2010.

[26]Nishikawa, H., "Accuracy-Preserving Boundary Flux Quadrature for Finite-Volume Discretization on Unstructured Grids," 2014, in review.

[27]van Leer, B., Lee, W.-T., and Roe, P. L., "Characteristic Time-Stepping or Local Preconditioning of the Euler Equations," *Proc. of 10th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 91-1552, Hawaii, 1991.

[28]Nishikawa, H. and Kitamura, K., "Very Simple, Carbuncle-Free, Boundary-Layer Resolving, Rotated-Hybrid Riemann Solvers," *J. Comput. Phys.*, Vol. 227, 2007, pp. 2560–2581.

[29]Toro, E. F., Spruce, M., and Speares, W., "Restoration of the Contact Surface in the HLL-Riemann Solver," *Shock Waves*, Vol. 4, 1994, pp. 25–34.

[30]Roe, P. L., "Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes," *J. Comput. Phys.*, Vol. 43, 1981, pp. 357–372.

[31]Katz, A. and Sankaran, V., "Mesh quality effects on the accuracy of CFD solutions on unstructured meshes," *J. Comput. Phys.*, Vol. 230, 2011, pp. 7670–7686.

[32]Katz, A. and Sankaran, V., "An Efficient Correction Method to Obtain a Formally Third-Order Accurate Flow Solver for Node-Centered Unstructured Grids," *J. Sci.Comput.*, Vol. 51, 2012, pp. 375–393.

[33]Diskin, B. and Thomas, J. L., "Effects of mesh regularity on accuracy of finite-volume schemes," *Proc. of 50th AIAA Aerospace Sciences Meeting*, AIAA Paper 2012-0609, Nashville, Tennessee, 2012.

[34]Carlson, J.-R., "Inflow/Outflow Boundary Conditions with Application to FUN3D," *NASA-TM–2011-217181*, October 2011.

[35]Nishikawa, H., "Robust and Accurate Viscous Discretization via Upwind Scheme - I: Basic Principle," *Comput. Fluids*, Vol. 49, No. 1, October 2011, pp. 62–86.

[36]Thomas, J. L., Diskin, B., and Nishikawa, H., "A Critical Study Agglomerated Multigrid Methods for Diffusion on Highly-Stretched Grids," *Comput. Fluids*, Vol. 40, No. 1, 2010, pp. 82–93.

[37]"Turbulence Modeling Resource Website," `http://turbmodels.larc.nasa.gov`.

[38]Coutanceau, M. and Bouard, R., "Experimental determination of the main features of the viscous flow in the wake of a circular cylinder in uniform translation. Part 1. Steady flow," *J. Fluid Mech.*, Vol. 79, 1977, pp. 231–256.

[39]Luo, H., Luo, L., Nourgaliev, R., and Mousseau, V. A., "A Reconstructed Discontinuous Galerkin Method for the Compressible Navier-Stokes Equations on Arbitrary Grids," *Proc. of 47th AIAA Aerospace Sciences Meeting*, AIAA Paper 2010-364, Orlando, Florida, 2010.

[40]Veer N. Vatsa, Mehdi R. Khorram, M. A. P. and Lockard, D. P., "Aeroacoustic Simulation of Nose Landing Gear on Adaptive Unstructured Grids with FUN3D," *Proc. of 19th AIAA/CEAS Aeroacoustics Conference*, AIAA Paper 2013-2071, Berlin, Germany, 2011.

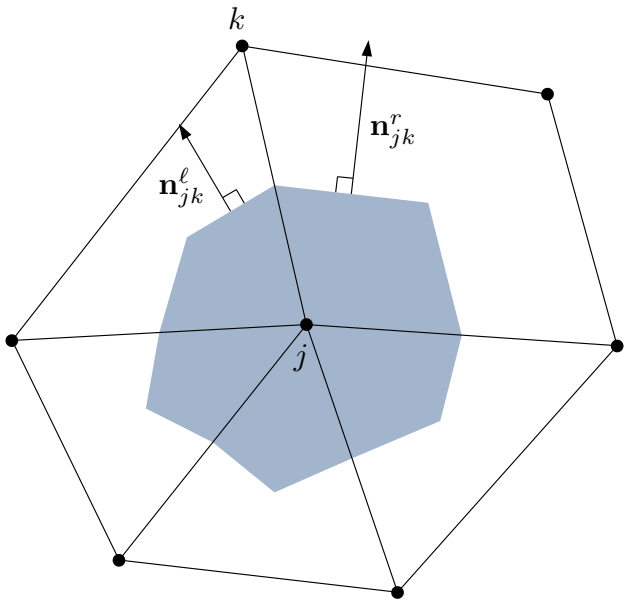

Figure 1. Dual control volume for the node-centered finite-volume method with scaled outward normals associated with an edge, $\{j, k\}$.

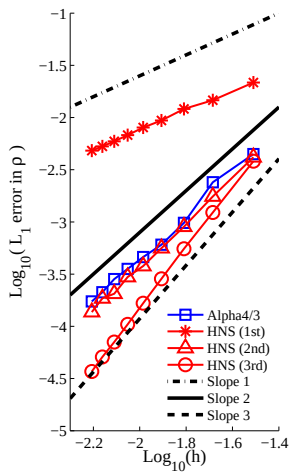

Figure 2. Error convergence of ρ for the compressible NS equations.

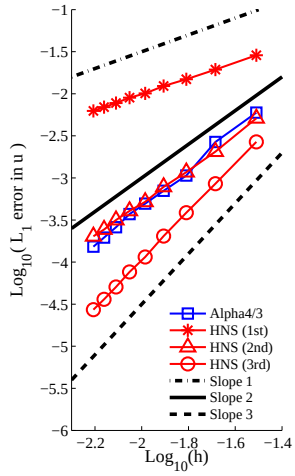

Figure 3. Error convergence of u for the compressible NS equations.

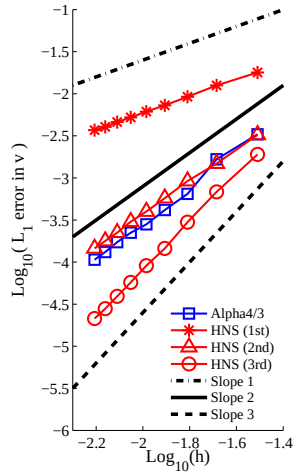

Figure 4. Error convergence of v for the compressible NS equations.

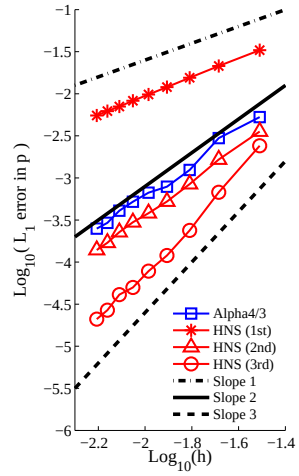

Figure 5. Error convergence of p for the compressible NS equations.

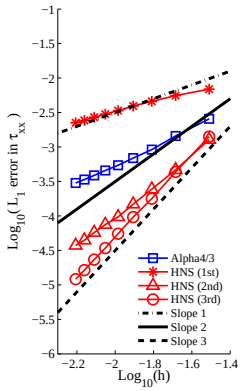

Figure 6. Error convergence of τ_{xx} for the compressible NS equations.

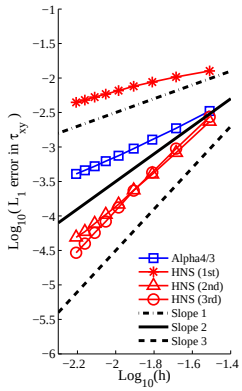

Figure 7. Error convergence of τ_{xy} for the compressible NS equations.

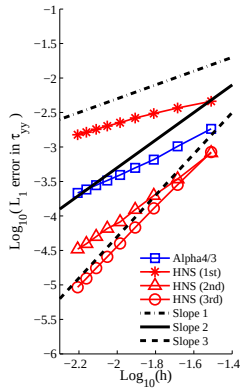

Figure 8. Error convergence of τ_{yy} for the compressible NS equations.

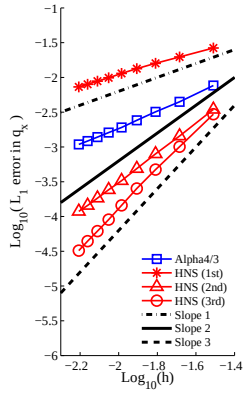

Figure 9. Error convergence of q_x for the compressible NS equations.

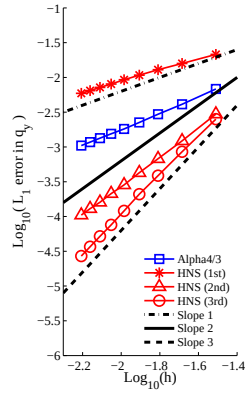

Figure 10. Error convergence of q_y for the compressible NS equations.

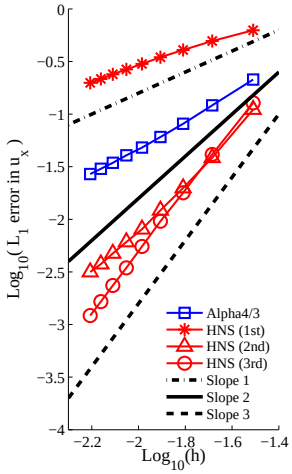

Figure 11. Error convergence of $\partial_x u$ for the compressible NS equations.

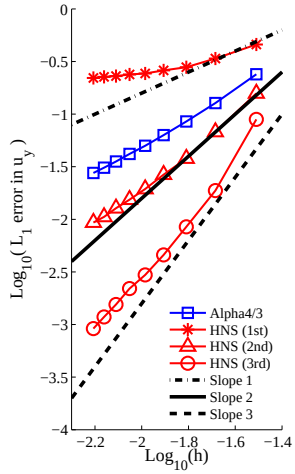

Figure 12. Error convergence of $\partial_y u$ for the compressible NS equations.

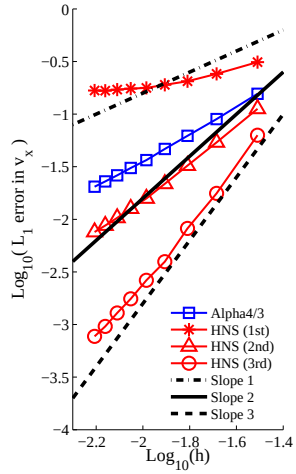

Figure 13. Error convergence of $\partial_x v$ for the compressible NS equations.

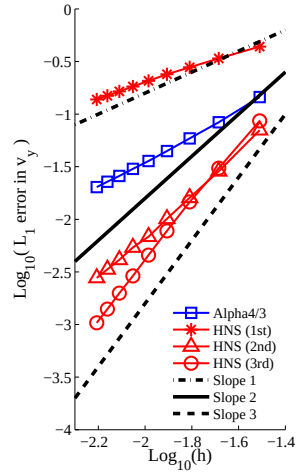

Figure 14. Error convergence of $\partial_y v$ for the compressible NS equations.

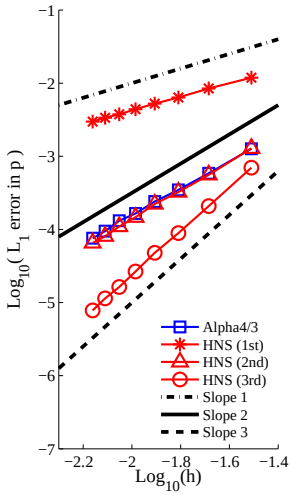

Figure 15. Error convergence of P for the incompressible NS equations.

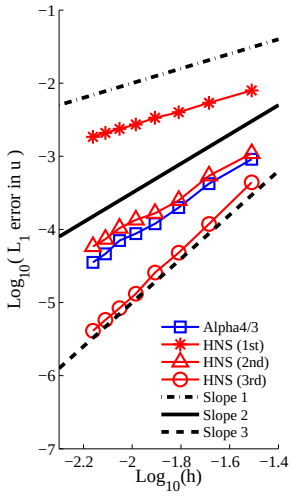

Figure 16. Error convergence of u for the incompressible NS equations.

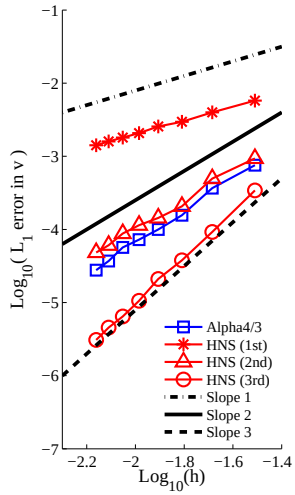

Figure 17. Error convergence of v for the incompressible NS equations.

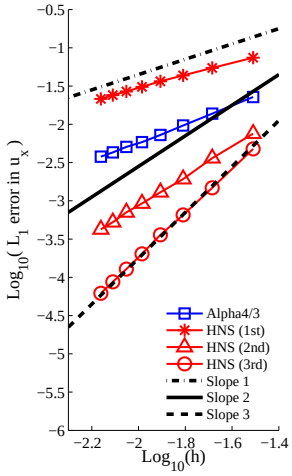

Figure 18. Error convergence of $\partial_x u$ for the incompressible NS equations.

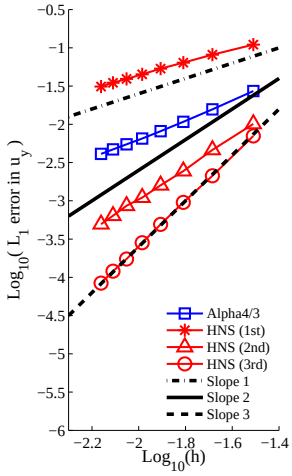

Figure 19. Error convergence of $\partial_y u$ for the incompressible NS equations.

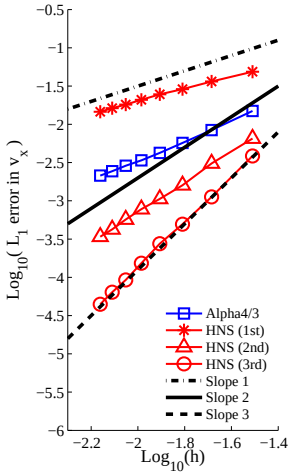

Figure 20. Error convergence of $\partial_x v$ for the incompressible NS equations.

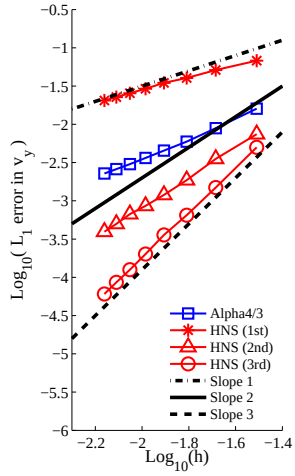

Figure 21. Error convergence of $\partial_y v$ for the incompressible NS equations.

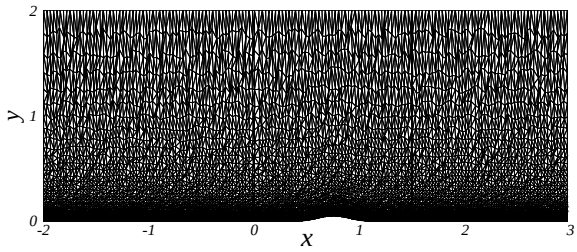

Figure 22. Grid1 for the bump case.

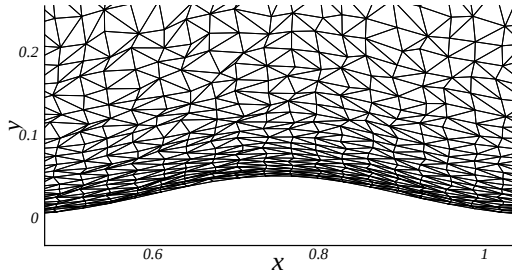

Figure 23. Zoom-up of Figure 22.

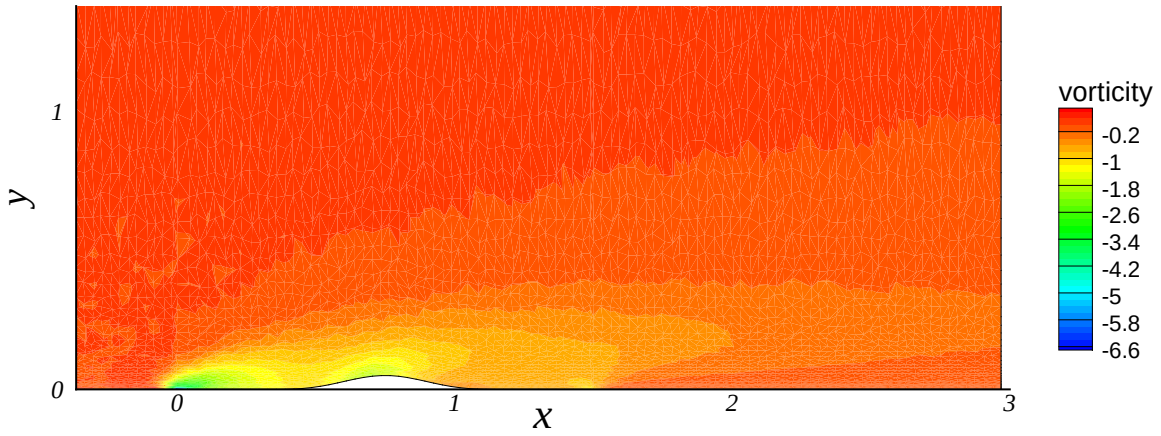

Figure 24. Vorticity contours obtained by a conventional scheme in the compressible-NS bump case (Grid1).

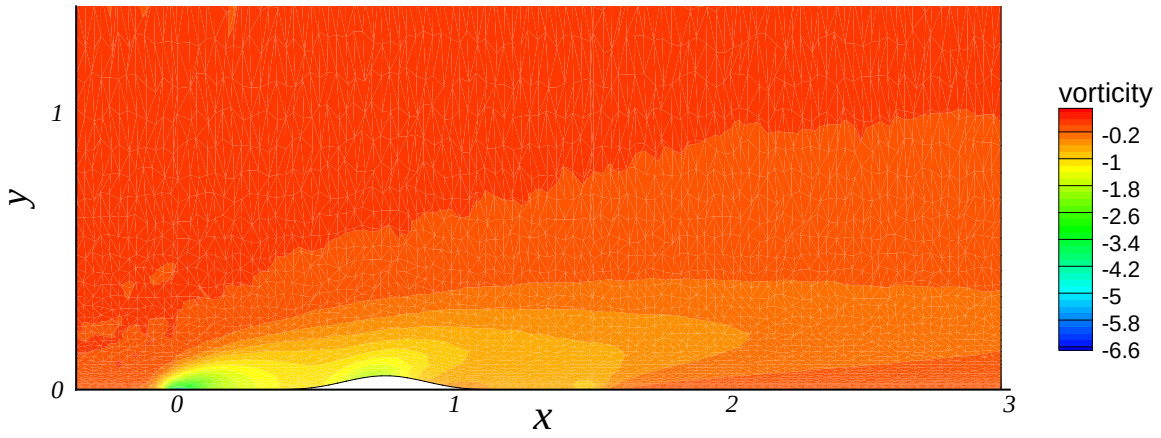

Figure 25. Vorticity contours obtained by the second-order HNS scheme in the compressible-NS bump case (Grid1).

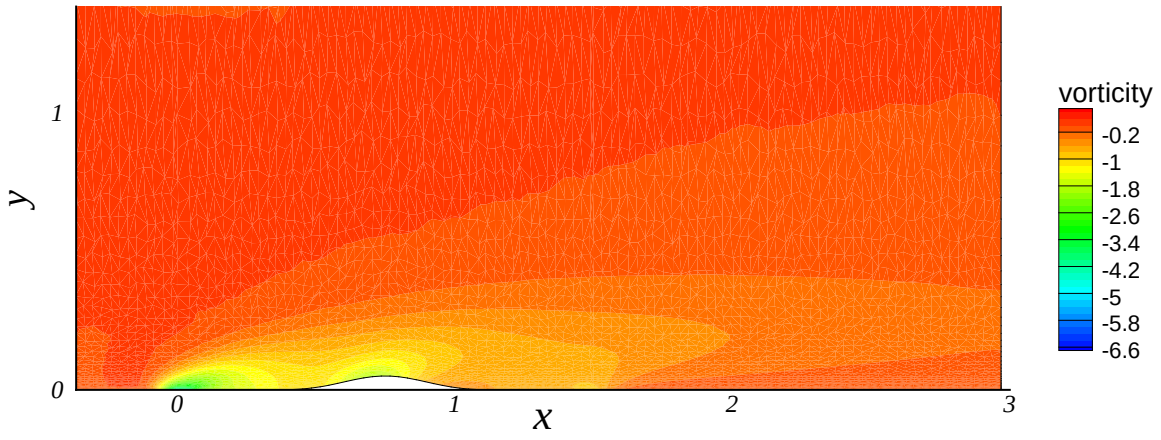

Figure 26. Vorticity contours obtained by the third-order HNS scheme in the compressible-NS bump case (Grid1).

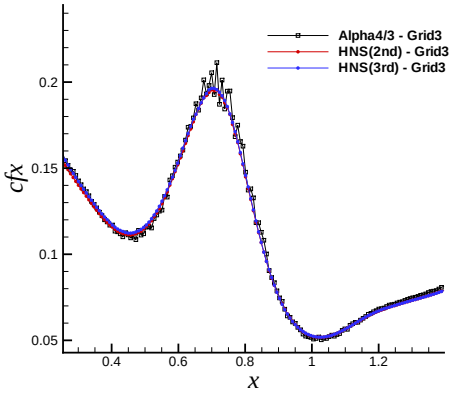

Figure 27. Skin friction coefficient distribution for the compressible-NS bump case.

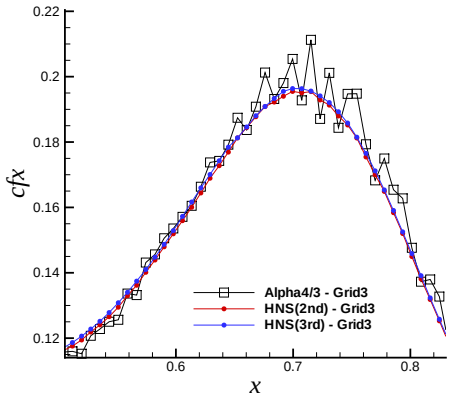

Figure 28. Zoom-up of the skin friction coefficient distribution for the compressible-NS bump case.

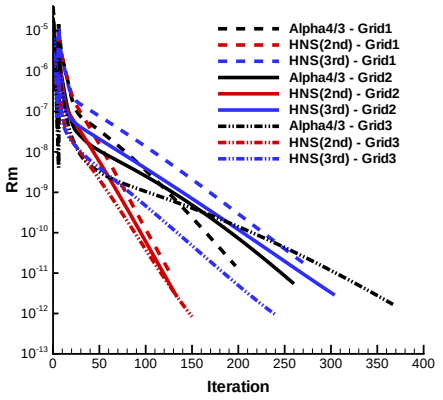

Figure 29. Maximum residual norm versus iteration for the compressible-NS bump case.

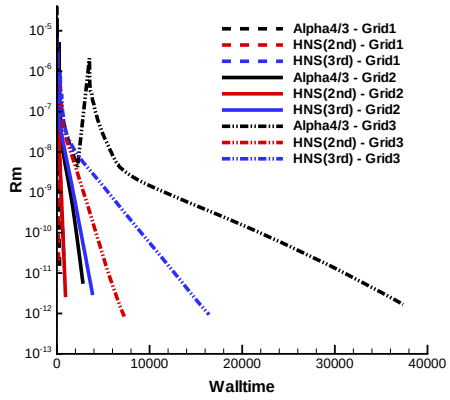

Figure 30. Maximum residual norm versus CPU time for the compressible-NS bump case.

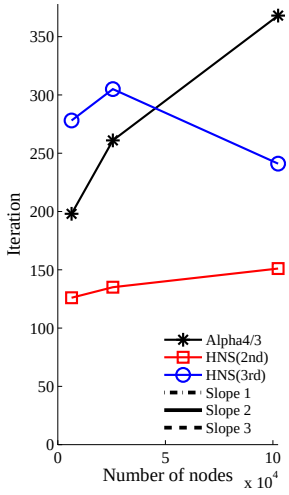

Figure 31. Iteration versus the number of nodes for the compressible-NS bump case.

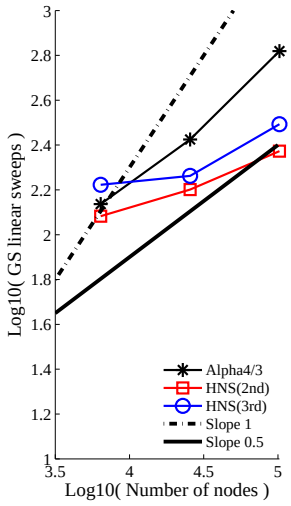

Figure 32. Linear sweep versus the number of nodes for the compressible-NS bump case.

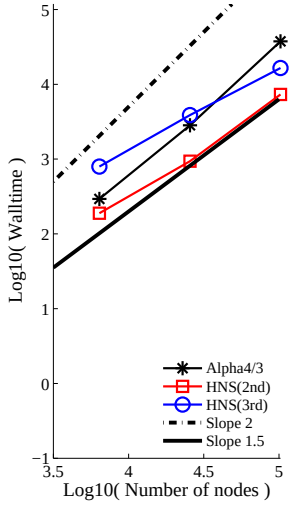

Figure 33. CPU time versus the number of nodes for the compressible-NS bump case.

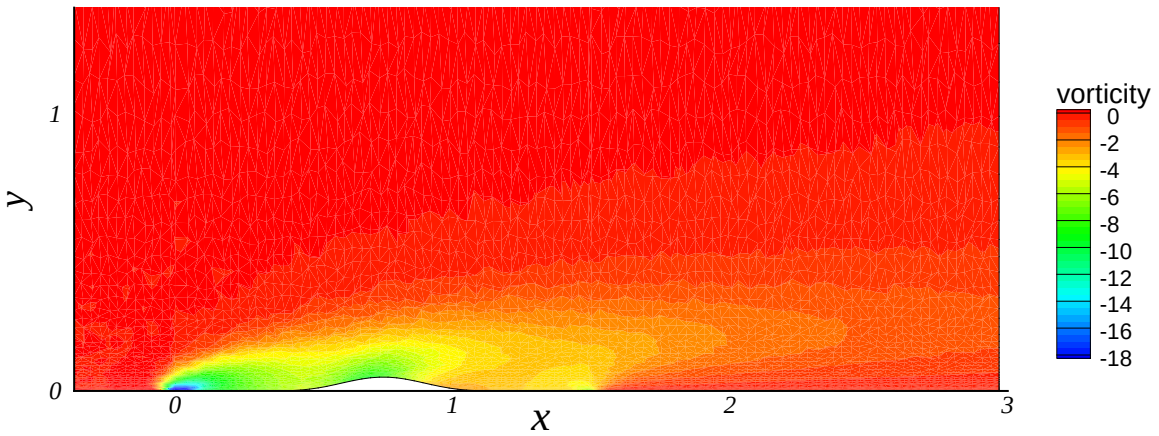

Figure 34. Vorticity contours obtained by a conventional scheme in the incompressible-NS bump case.

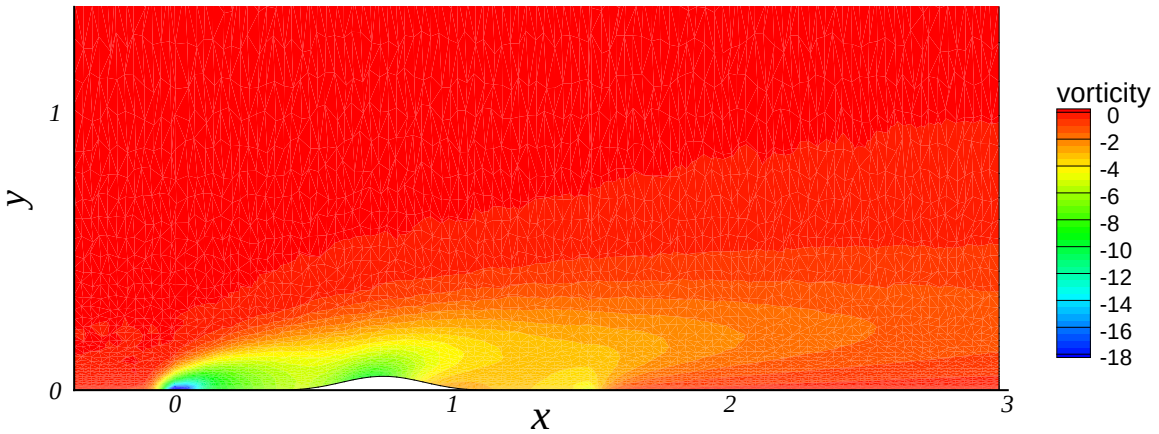

Figure 35. Vorticity contours obtained by the second-order HNS scheme in the incompressible-NS bump case.

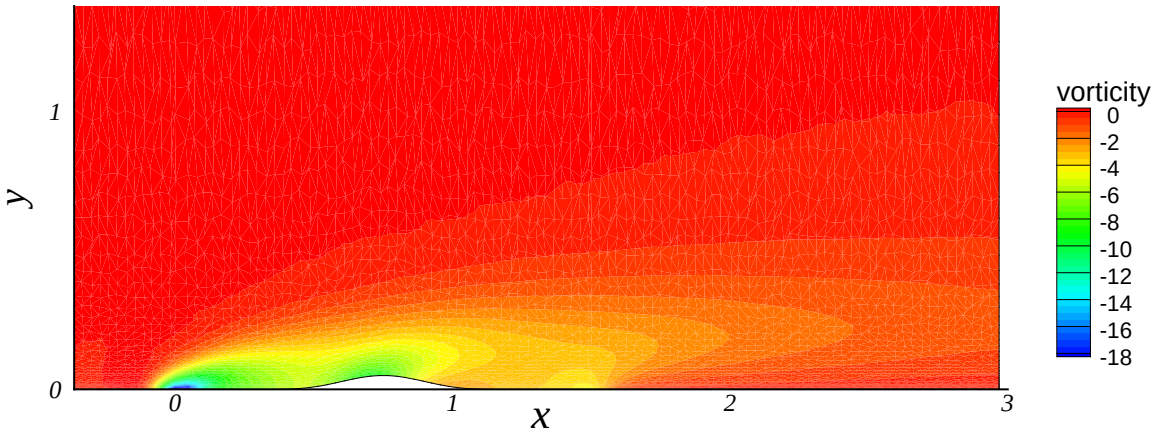

Figure 36. Vorticity contours obtained by the third-order HNS scheme in the incompressible-NS bump case.

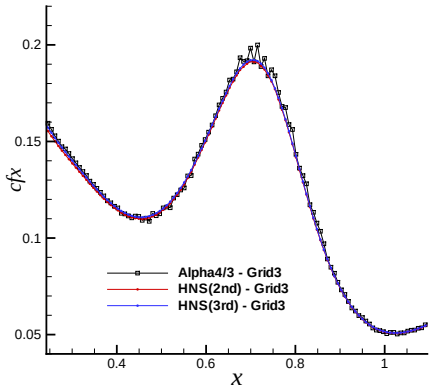

Figure 37. Skin friction coefficient distribution over the bump for the incompressible-NS bump case.

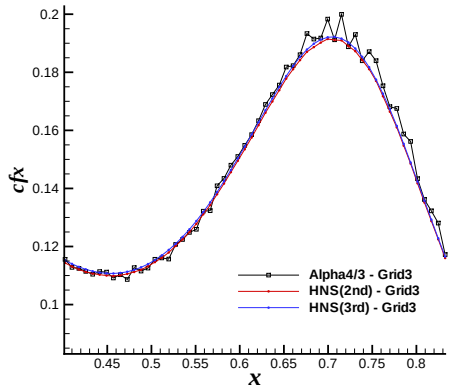

Figure 38. Zoom-up of the skin friction coefficient distribution over the bump for the incompressible-NS bump case.

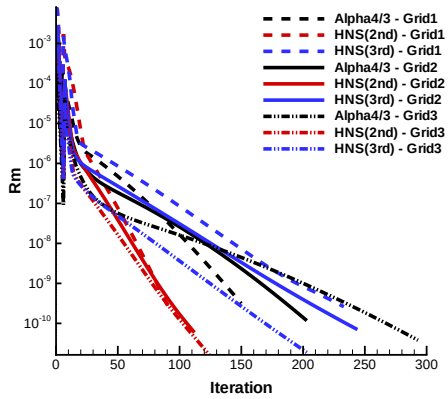

Figure 39. Maximum residual norm versus iteration for the incompressible-NS bump case.

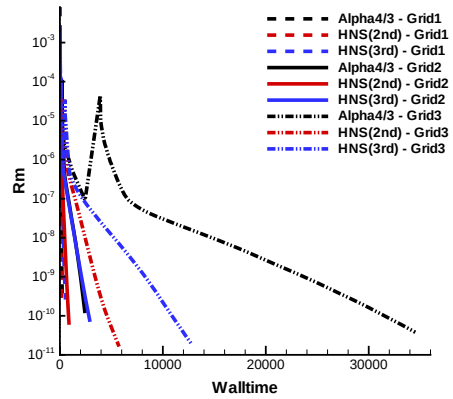

Figure 40. Maximum residual norm versus CPU time for the incompressible-NS bump case.

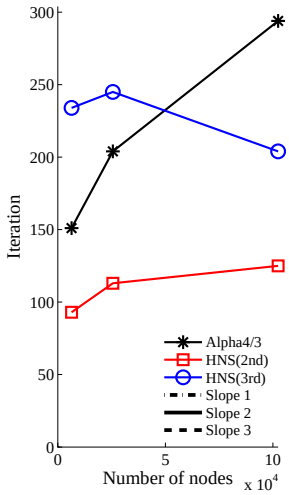

Figure 41. Iteration versus the number of nodes for the incompressible-NS bump case.

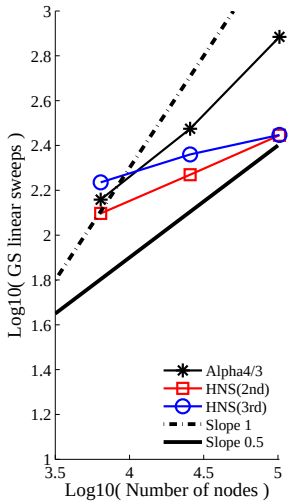

Figure 42. Linear sweep versus the number of nodes for the incompressible-NS bump case.

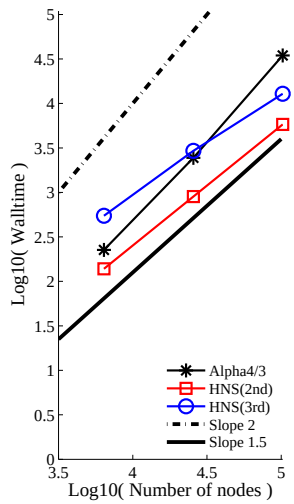

Figure 43. CPU time versus the number of nodes for the incompressible-NS bump case.

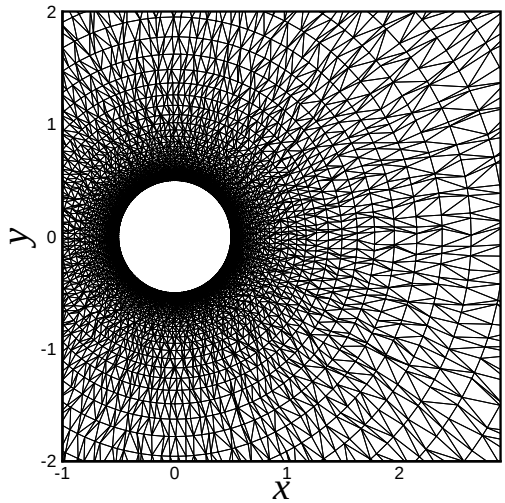

Figure 44. Grid1 for the cylinder case. The outer boundary is lcoated at a distance 500 times the diameter.

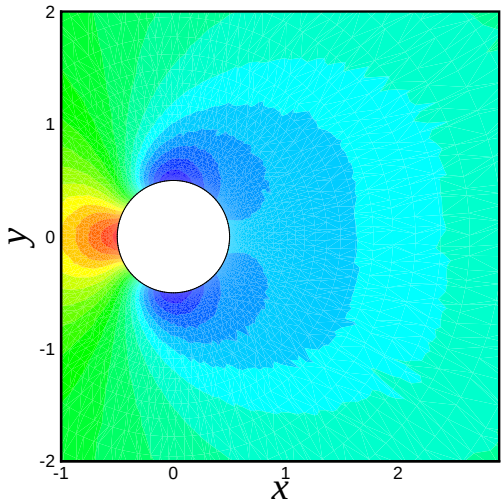

Figure 45. Pressure contours and streamlines obtained by the conventional scheme in the compressible-NS cylinder case.

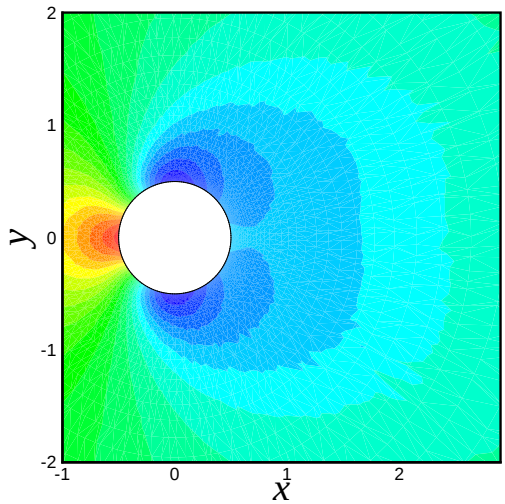

Figure 46. Pressure contours and streamlines obtained by the second-order HNS scheme in the compressible-NS cylinder case.

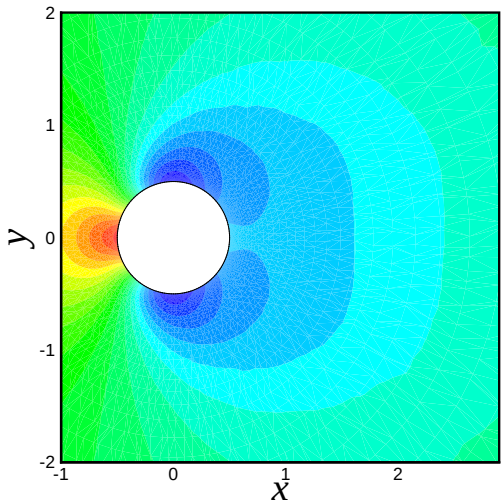

Figure 47. Pressure contours and streamlines obtained by the third-order HNS scheme in the compressible-NS cylinder case.

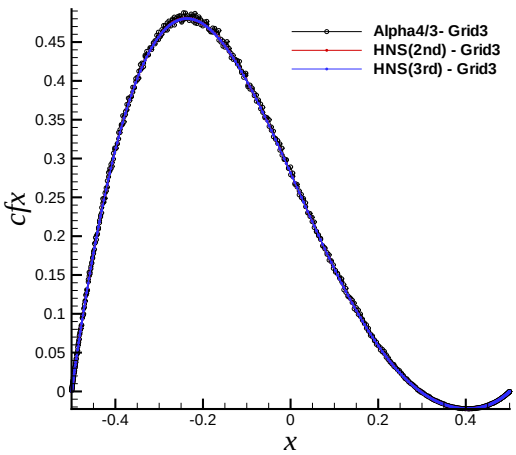

Figure 48. Skin friction coefficient distribution for the compressible-NS cylinder case.

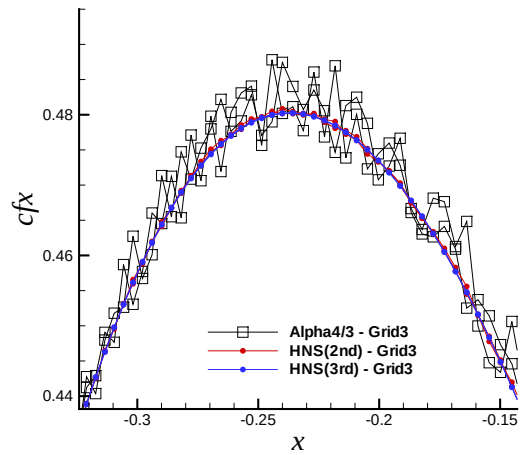

Figure 49. Zoom-up of the skin friction coefficient distribution for the compressible-NS cylinder case.

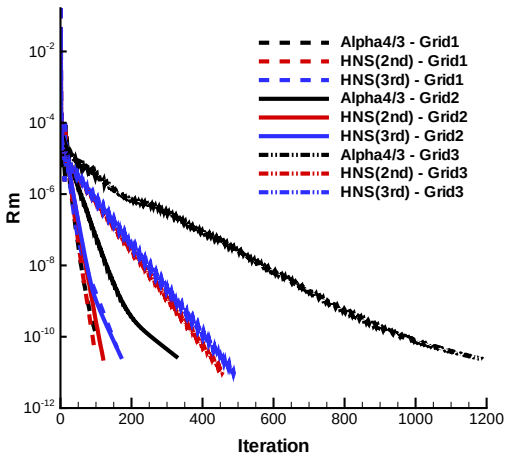

Figure 50. Maximum residual norm versus iteration for the compressible-NS cylinder case.

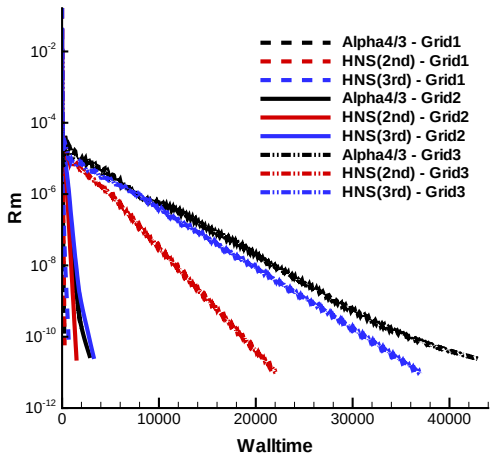

Figure 51. Maximum residual norm versus CPU time for the compressible-NS cylinder case.

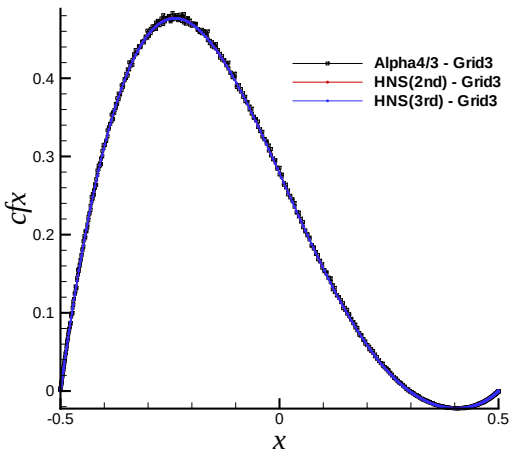

Figure 52. Skin friction coefficient distribution over the cylinder for the incompressible-NS cylinder case.

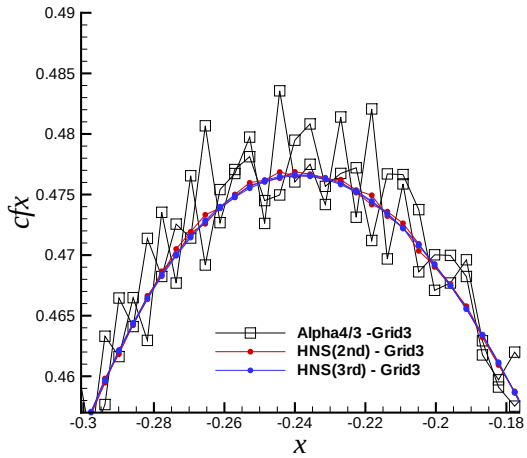

Figure 53. Zoom-up of the skin friction coefficient distribution over the cylinder for the incompressible-NS cylinder case.

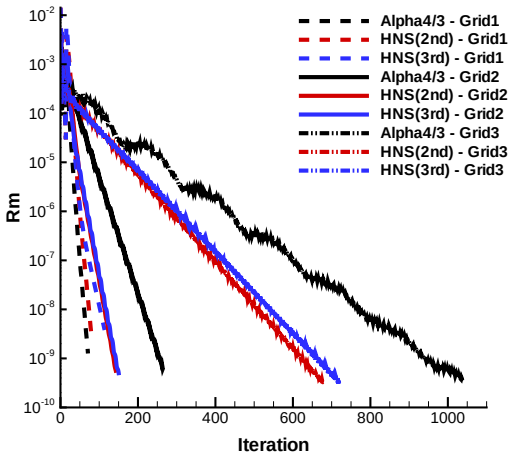

Figure 54. Maximum residual norm versus iteration for the incompressible-NS cylinder case.

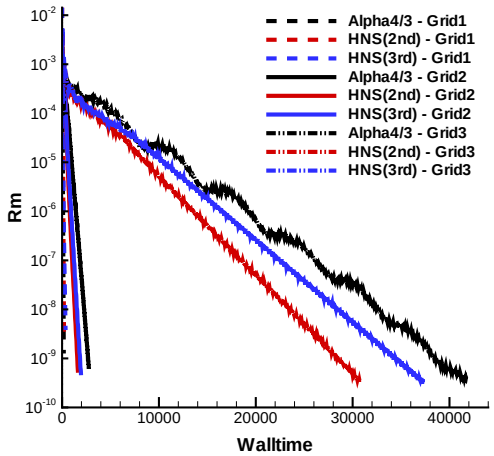

Figure 55. Maximum residual norm versus CPU time for the incompressible-NS cylinder case.