On the Efficiency of Higher-Order Implicit Solvers for Compressible Flow

Zaid H. Sabri*, Ray Hixon†, and Jeffrey Severino‡
University of Toledo, Toledo, OH, 43606

Time marching techniques constitute a significant area of interest within Computational Aeroacoustics. Explicit time-marching schemes solve the governing equations but may require excessive time steps due to numerical stability limits. Traditional implicit methods, which often achieve fast convergence rates, suffer through high computational work. Implicit time marching can offer low costs per sub-iteration if coupled with low-order differencing stencils. The present research explores using traditional implicit methods with high-order stencils efficiently. The numerics of the governing equations are preconditioned using low-order differences while using highly accurate stencils on the physics of the equations. The developed scheme is stable and validated against several benchmark problems.

I. Nomenclature

ADI	=	Alternating Direction Implicit
CAA	=	Computational Aeroacoustics
CFL	=	Courant-Friedrichs-Lewy
DRP	=	Dispersion Relation Preserving
LHS	=	Left Hand Side
LU-SGS	=	Lower-Upper Symmetric Gauss-Seidel
ODE	=	Ordinary Differential Equations
RDRP	=	Rational Dispersion Relation Preserving
RHS	=	Right Hand Side
TriDi	=	Tridiagonal Matrix
α	=	Phase Speed
ϵ	=	Truncation Error
ε	=	Shift Operator
ν	=	CFL
σ	=	Scaling Factor
j	=	$\sqrt{-1}$
^{n}	=	time level counter
^{l}	=	iteration counter

II. Introduction

Computational Aeroacoustics (CAA) is focused on the simulation of acoustic fields produced by unsteady flows [1–4]. To achieve this goal, it is necessary to accurately simulate the unsteady evolution of flow disturbances, which requires using high-accuracy spatial and temporal numerical schemes. As these schemes have been developed and validated, the application problems of interest have also increased in complexity, typically involving realistic nonlinear flows about complex geometries.

Time marching, a numerical integration technique, can obtain steady-state solutions for steady-flow problems and time-accurate solutions for unsteady-flow problems. There are two time steps of interest in a time marching scheme for an unsteady flow problem. The first is the largest time step that can be taken while retaining an accurate unsteady

*Graduate Research Assistant, Mechanical, Industrial, and Manufacturing Engineering (MIME) Department, Member AIAA.
†Professor, Mechanical, Industrial, and Manufacturing Engineering (MIME) Department, Senior Member AIAA.
‡Graduate Research Assistant, Mechanical, Industrial, and Manufacturing Engineering (MIME) Department, Member AIAA.

solution (the accuracy limit). The second time step of interest is the largest time step that can be taken while retaining a stable calculation (the stability limit). On the other hand, the stability limit is the only time step of interest for steady flow problems [5].

Explicit or implicit time-marching methods can integrate the governing equations of fluid flows. Explicit time marching is a numerical technique where the values of the coupled equations depend on the current time level. When using explicit schemes, the inviscid stability limit is related to the minimum time required for the fastest propagating wave to move from one grid point (the Courant-Friedrichs-Lewy (CFL) condition). Since the computational domain must be highly clustered near the body (to resolve the viscous boundary layer), the size of the time step an explicit scheme can take is restricted by the CFL condition [6, 7]. The stability restriction on the allowable time step can be eliminated by using an A-stable or L-stable implicit time marching scheme. Implicit time marching schemes can allow large time steps [8] at the price of sub-iterations to converge the solution at each time step. Much research has focused on indirect implicit methods, since an unfactored implicit scheme that directly inverts a large block-banded matrix is impractical in multi-dimensions.

Beam and Warming [9], along with Briley and McDonald [10], developed a practical implicit scheme by approximately factorizing the implicit operator in delta form, an algorithm known as Alternating Direction Implicit (ADI). Widely disseminated codes such as ARC2D/3D [11, 12], CFL3D [13], INS3D [14], OVERFLOW [15] and many more are based on the ADI scheme. Yoon and Jameson derived another implicit algorithm, the Lower-Upper Symmetric-Gauss-Seidel method (LU-SGS) [16]. LU-SGS does not require additional relaxation or factorization on planes of sweep and requires less computational work per iteration than most explicit schemes [7]. LU-SGS is unconditionally stable in one, two, and three dimensions, unlike ADI, which is conditionally stable in three dimensions. On the other hand, Glenn-HT [17], a general purpose multi-block 3D Navier-Stokes solver, uses an explicit left-hand-side to implement implicit time marching solutions (a method commonly referred to as dual time stepping [18–20]).

Implicit methods described above are computationally expensive when using differencing stencils with high orders of accuracy. Highly accurate spatial differencing schemes have been developed in the past (e.g., Refs. [21–26]), which improve the accuracy on a given grid. Combining highly accurate spatial differencing schemes with implicit time marching results in a computationally expensive matrix system. FDL3DI [26–28] developed by Miguel Visbal and Datta Gaitonde from the Air Force Research Lab and Ohio State University respectively, have introduced implicit time marching with higher-order differencing stencils. The numerics of the governing equations were approximated using low-order difference, and stability was possible due to the addition of implicit dissipation.

This work uses the traditional implicit method with high-order differencing stencils while reducing the computational cost. The approach used in this investigation is similar to that used by FDL3DI; high efficiency is achieved by fixing the numerics of the time marching method to a low-order differencing stencil while using highly accurate spatial differencing schemes on the physics of the governing equations. Whereas FDL3DI obtained stability by adding dissipation to the numerics, stability is obtained here by scaling errors from the numerics to overcome errors from the physics of the governing equations. The developed scheme is stable and validated against several benchmark problems.

III. Euler Equation and Implicit Algorithm

The Euler equation, written in vector form:

$$\frac{\partial Q}{\partial t} + \frac{\partial E}{\partial x} = 0 \tag{1}$$

Using the superscript $'n'$ to denote the current time level, $'n+1'$ to denote the next time level, and a subscript $'i'$ as a grid point counter, an implicit time marching scheme is written as:

$$\left(\frac{\partial Q}{\partial t}\right)_i^{n+1} + \left(\frac{\partial E}{\partial x}\right)_i^{n+1} = 0 \tag{2}$$

An iteration procedure ($'l'$ denotes the iteration counter) can be defined in order to solve Eq. (2):

$$\begin{aligned} Q_i^{n+1,l+1} &= Q_i^{n+1,l} + \Delta Q_i \\ E_i^{n+1,l+1} &= E_i^{n+1,l} + \left(\frac{\partial E}{\partial Q}\right)_i^{n+1,l} \Delta Q_i \end{aligned} \tag{3}$$

A first-order backward difference is used to approximate the time derivative, and the flow equations are written at the new iteration level $l+1$ as:

$$\left[\Delta Q_i + \Delta t \cdot \frac{\partial}{\partial x}\left(\left(\frac{\partial E}{\partial Q}\right)_i^{n+1,l} \Delta Q_i\right)\right] = -\Delta t \{RHS\}_i^{n+1,l} \tag{4}$$

The equation above represents the change in the flow variable ΔQ_i at each grid point i, during the interval from time $(n)\Delta t$ to time $(n+1)\Delta t$. The right-hand side of the equation represents a finite difference approximation to the complete governing equation:

$$\{RHS\}_i^{n+1,l} = \left(\frac{\partial Q}{\partial t} + \frac{\partial E}{\partial x}\right)_i^{n+1,l}$$

Note that the implicit iteration is used to converge the solution of the equation at the new time level – once the equation converges, the value of ΔQ_i is zero. The direct inversion of a large block banded matrix of the unfactored scheme coupled with high-order differencing stencils results in an expensive solve per iteration. To alleviate this difficulty, research has focused on indirect methods such as ADI and LU-SGS.

A. Formulation of an ADI Scheme

The traditional implicit method from Eq. (4) can be re-written as:

$$\left(I + \Delta t \frac{\partial A_i^{n+1,l}}{\partial x}\right) \Delta Q_i = -\Delta t \{RHS\}_i^{n+1,l} \tag{5}$$

If the scheme was extended into two dimensions, the ADI factorization is retained [9, 10]. This methodology replaces the implicit operator of the unfactored scheme with a product of two one-dimensional operators:

$$\left(I + \Delta t \frac{\partial A_{i,j}^{n+1,l}}{\partial x}\right)\left(I + \Delta t \frac{\partial B_{i,j}^{n+1,l}}{\partial y}\right) \Delta Q_{i,j} = -\Delta t \{RHS\}_{i,j}^{n+1,l} \tag{6}$$

where A and B are the Jacobian matrices of the convective flux vectors:

$$A_{i,j} = \frac{\partial E_{i,j}}{\partial Q_{i,j}} \qquad\qquad B_{i,j} = \frac{\partial F_{i,j}}{\partial Q_{i,j}} \tag{7}$$

Similarly, three-dimensional ADI can be written as:

$$\left(I + \Delta t \frac{\partial A_{i,j,k}^{n+1,l}}{\partial x}\right)\left(I + \Delta t \frac{\partial B_{i,j,k}^{n+1,l}}{\partial y}\right)\left(I + \Delta t \frac{\partial C_{i,j,k}^{n+1,l}}{\partial z}\right) \Delta Q_{i,j,k} = -\Delta t \{RHS\}_{i,j,k}^{n+1,l} \tag{8}$$

where A, B, and C are the Jacobian matrices of the convective flux vectors:

$$A_{i,j,k} = \frac{\partial E_{i,j,k}}{\partial Q_{i,j,k}} \qquad B_{i,j,k} = \frac{\partial F_{i,j,k}}{\partial Q_{i,j,k}} \qquad C_{i,j,k} = \frac{\partial G_{i,j,k}}{\partial Q_{i,j,k}} \tag{9}$$

Widely disseminated codes such as ARC2D/3D [11, 12], CFL3D [13], INS3D [14], OVERFLOW [15], FDL3DI [26–28], and many more approximately factor the LHS matrix operator (implicit side) and solve using an ADI scheme. The traditional implicit methods and two-dimensional ADI schemes are unconditionally stable. However, the drawback with ADI is that it becomes conditionally stable in three-dimension due to the increase of factorization error terms, leading to a restriction on the time step.

B. Formulation of an LU-SGS Scheme

Another standard indirect method used to approximate the LHS matrix is known as the Lower-Upper Symmetric-Gauss Seidel (LU-SGS) method. The LU-SGS method, developed by Yoon and Jameson in 1988, is an approximation scheme developed for steady-state solutions of the Euler and Navier-Stokes equations [16]. This scheme does not require flux splitting for approximate Newton iteration, nor does it require additional relaxation or factorization on planes of

sweep [16]. When compared to explicit schemes, this method requires less computational work per iteration [7]. Unlike Beam and Warming [9] ADI factorization, LU-SGS is unconditionally stable in one, two, and three dimensions. The one-dimensional LU-SGS method for approximate Newton iteration can be derived as:

$$LD^{-1}U\Delta Q \;=\; -\,\Delta t\,\{RHS\}_i^{n+1,l} \tag{10}$$

where:

$$\begin{aligned} L &= I + \alpha\Delta t\left(D_x^- A^+ - A^-\right) \\ D^{-1} &= I + \alpha\Delta t\left(A^+ - A^-\right) \\ U &= I + \alpha\Delta t\left(D_x^+ A^- + A^+\right) \end{aligned} \tag{11}$$

Here, D_x^- is a backward difference operator, and D_x^+ is a forward difference operator resulting in lower and upper triangular matrices, respectively.

IV. Performance of Finite Differencing Scheme on the Convection Equation

As shown in Eq. (4), it is necessary to compute the first spatial derivatives of the solution in order to obtain the time derivative; this time derivative is then integrated in time. At this point in the analysis, the time integration scheme is considered 'perfect' – the only errors will arise from the space derivative itself. The spatial derivatives used in Eq. (4) are computed numerically using finite differencing schemes. In the finite-difference formulation, the continuous problem domain is "discretized" such that the dependent variables are considered to exist only at discrete points. A finite difference stencil for the first derivative on a uniform grid is expressed as:

$$\left.\frac{\partial u}{\partial x}\right|_i \;=\; \frac{1}{\Delta x}\begin{pmatrix} a_1\left(u_{i+1}-u_{i-1}\right) \\ +\,a_2\left(u_{i+2}-u_{i-2}\right) \\ +\,a_3\left(u_{i+3}-u_{i-3}\right)\end{pmatrix}$$

where Δx is the distance between grid points. Consider a simple harmonic solution of the equation:

$$u(x,t) = e^{jkx}$$

where k is the wavenumber and $j = \sqrt{-1}$. In this solution, the wave convects in the positive x direction at a speed of c and loses amplitude as it convects. The analytical space derivative of this solution is:

$$\left.\frac{\partial u}{\partial x}\right|_{exact} \;=\; jke^{jkx}$$

Substituting the simple harmonic solution into the finite difference stencil:

$$\begin{aligned} \left.\frac{\partial u}{\partial x}\right|_{numerical} &= \frac{j}{\Delta x}\begin{pmatrix} 2a_1\sin\left(k\Delta x\right) \\ +\,2a_2\sin\left(2k\Delta x\right) \\ +\,2a_3\sin\left(3k\Delta x\right)\end{pmatrix}e^{jkx} \\ &= \underbrace{\left(\frac{(k\Delta x)^*}{k\Delta x}\right)}_{\alpha}\left.\frac{\partial u}{\partial x}\right|_{exact} \end{aligned}$$

where α is the error in the numerical convection speed (phase speed). Plots of the phase speed error vs. wavenumber, $(k\Delta x)$, are shown in Fig. 1. In general, increasing the order of accuracy of a spatial differencing scheme will also improve its accuracy on a given grid by increasing the range of values of $(k\Delta x)$ over which α is close to a value of one. The maximum order of accuracy first derivative schemes are:

$$\begin{aligned} \left.\frac{\partial u}{\partial x}\right|_i &= \frac{u_{i+1}-u_{i-1}}{2\Delta x} + \varphi\left(\Delta x^2\right) \\ \left.\frac{\partial u}{\partial x}\right|_i &= \frac{-u_{i+2}+8u_{i+1}-8u_{i-1}+u_{i-2}}{12\Delta x} + \varphi\left(\Delta x^4\right) \\ \left.\frac{\partial u}{\partial x}\right|_i &= \frac{u_{i+3}-9u_{i+2}+45u_{i+1}-45u_{i-1}+9u_{i-2}-u_{i-3}}{60\Delta x} + \varphi\left(\Delta x^6\right) \end{aligned}$$

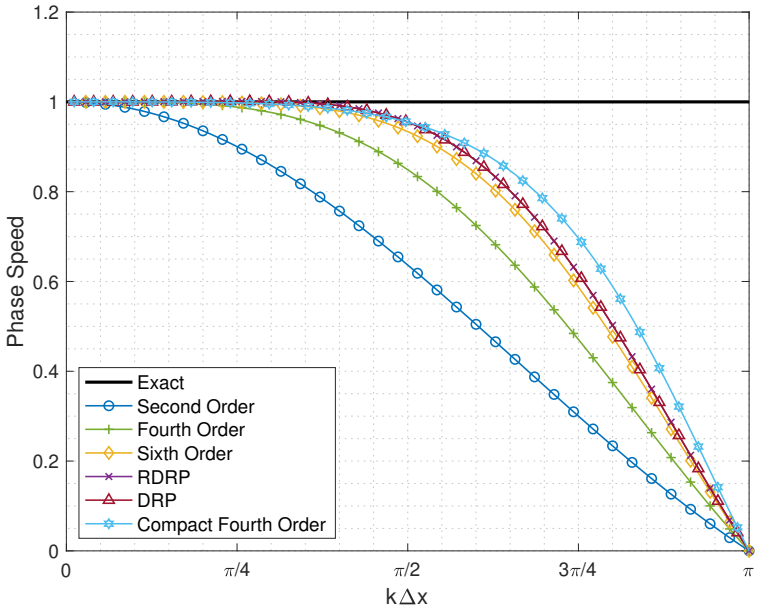


Fig. 1 Convection Speed Comparison

Tam and Shen DRP [22] and Hixon RDRP [29] schemes are widely robust optimized schemes given as:

- DRP:

$$\left.\frac{\partial u}{\partial x}\right|_i = \frac{1}{\Delta x}\begin{pmatrix} 0.0208431427703u_{i+3} - 0.166705904415u_{i+2} + 00.570882380518u_{i+1} \\ - 00.570882380518u_{i-1} + 0.166705904415u_{i-2} - 0.0208431427703u_{i-3}\end{pmatrix} + \varphi\left(\Delta x^4\right)$$

- RDRP:

$$\left.\frac{\partial u}{\partial x}\right|_i = \frac{u_{i+3} - 8u_{i+2} + 37u_{i+1} - 37u_{i-1} + 8u_{i-2} - u_{i-3}}{48\Delta x} + \varphi\left(\Delta x^4\right)$$

Where compact schemes can obtain higher order derivatives without increasing the stencil size:

$$\frac{1}{4}\left.\frac{\partial u}{\partial x}\right|_{i+1} + \left.\frac{\partial u}{\partial x}\right|_i + \frac{1}{4}\left.\frac{\partial u}{\partial x}\right|_{i-1} = \frac{3}{4}\frac{u_{i+1} - u_{i-1}}{\Delta x} + \varphi\left(\Delta x^4\right)$$

However, an issue when using compact differences or larger stencils for implicit schemes is that the matrix on the LHS is expensive to solve (due to the number of diagonals).

In order to decrease the computational work faced when solving for the flow variable $\{\Delta Q_i\}$, the LHS matrix is approximated (preconditioned) using second-order differences while maintaining high-order differences in the physics of the governing equations. For example, solving the governing equations using the RDRP differencing stencil:

$$\left[I + \Delta t\left(\frac{A_{i+3} - 8A_{i+2} + 37A_{i+1} - 37A_{i-1} + 8A_{i-2} - A_{i-3}}{48\Delta x}\right)^{n+1,l}\right]\Delta Q_i = -\Delta t\left\{RHS\right\}_i^{n+1,l} \tag{12}$$

is preconditioned as:

$$\left[I + \Delta t\left(\frac{A_{i+1} - A_{i-1}}{2\Delta x}\right)^{n+1,l}\right]\Delta Q_i = -\Delta t\left\{RHS\right\}_i^{n+1,l} \tag{13}$$

where $\{RHS\}_i^{n+1,l}$ in Eq. (12) **and** Eq. (13) is:

$$\left\{RHS\right\}_i^{n+1,l} = \left(\left.\frac{\partial Q}{\partial t}\right|_i^{n+1,l} + \frac{\partial}{\partial x}\left(\frac{E_{i+3} - 8E_{i+2} + 37E_{i+1} - 37E_{i-1} + 8E_{i-2} - E_{i-3}}{48\Delta x}\right)^{n+1,l}\right)$$

Approximation made to the LHS matrix will cause numerical errors, which can yield instability. A von Neumann stability analysis is performed to check the stability of the preconditioned implicit time step.

V. Stability for a Preconditioned Implicit Scheme

A. Fourier or Von Neumann Analysis

The Fourier or Von Neumann stability analysis checks the stability of finite difference schemes once applied to PDEs. The truncation error for the inviscid convection equation is defined as:

$$u_i^n \;=\; D_i^n + \epsilon_i^n$$

where D exactly satisfies the discretized equation. Substituting in and eliminating D gives the error equation:

$$\frac{\partial \epsilon}{\partial t} + \left(c \cdot \frac{\partial \epsilon}{\partial x} \right)^{n+1} \;=\; 0 \tag{14}$$

Assuming that the error is a linear combination of Fourier modes in space:

$$\epsilon_n \;=\; \sum_{k=-\frac{\pi}{\Delta x}}^{\frac{\pi}{\Delta x}} A_k^n e^{jkx} \qquad \epsilon_{n+1} \;=\; \sum_{k=-\frac{\pi}{\Delta x}}^{\frac{\pi}{\Delta x}} A_k^{n+1} e^{jkx} \qquad \Delta\epsilon \;=\; \sum_{k=-\frac{\pi}{\Delta x}}^{\frac{\pi}{\Delta x}} \Delta A_k e^{jkx} \tag{15}$$

substituting into equation (13), and defining:

$$\nu \;=\; \frac{c\Delta t}{\Delta x}$$

gives the equation for each Fourier mode as:

$$\begin{aligned} \Delta\epsilon + \frac{\nu}{2}\left(\Delta\epsilon_{i+1} - \Delta\epsilon_{i-1}\right) \;&=\; -\Delta t \left(\left.\frac{\partial}{\partial x}\right|_{RHS} (c) \right) \epsilon_i^n \\ \left| \frac{A_k^{n+1}}{A_k^n} \right|^2 \;&= 1 + \frac{\nu^2 \alpha_2 \left(\alpha_2 - 2\alpha_1\right)}{1 + \nu^2 \alpha_1^2} \end{aligned} \tag{16}$$

where α_1 and α_2 are the second order and RHS difference errors in the numerical convection speed, respectively. A finite difference scheme is stable if the errors made at one-time step of the calculation do not cause the errors to be amplified. Such a scheme exists if the amplification factor is less than or equal to one.

It can be proved stability is only possible in Eq. (16) at high CFL if:

$$\alpha_2 - 2\alpha_1 \leq 0$$

Plotting the phase speed error difference (Fig. 2), instability is present at high wavenumbers. If Eq. (16) is approximated once again such that a scaling factor, σ, is multiplying the second order difference:

$$\begin{aligned} \Delta\epsilon + \frac{\sigma\nu}{2}\left(\Delta\epsilon_{i+1} - \Delta\epsilon_{i-1}\right) \;&=\; -\Delta t \left(\left.\frac{\partial}{\partial x}\right|_{RHS} (c) \right) \epsilon_i^n \\ \left| \frac{A_k^{n+1}}{A_k^n} \right|^2 \;&= 1 + \frac{\nu^2 \alpha_2 \left(\alpha_2 - 2\sigma\alpha_1\right)}{1 + \nu^2 \sigma^2 \alpha_1^2} \end{aligned} \tag{17}$$

for stability:

$$\alpha_2 - 2\alpha_1\sigma \leq 0$$

and the scaling factor values are shown in Table 1. Hence the scaling factor is physically insuring the numerical errors from the second-order difference can overcome the numerical errors from the RHS differencing stencil. Finally, the new preconditioned scheme can be written as follows:

$$\left[I + \Delta t\sigma \left(\frac{A_{i+1} - A_{i-1}}{2\Delta x} \right)^{n+1,l} \right] \Delta Q_i \;=\; -\Delta t \left\{ RHS \right\}_i^{n+1,l} \tag{18}$$

this will be referred to as the preconditioned matrix with scaling factors.

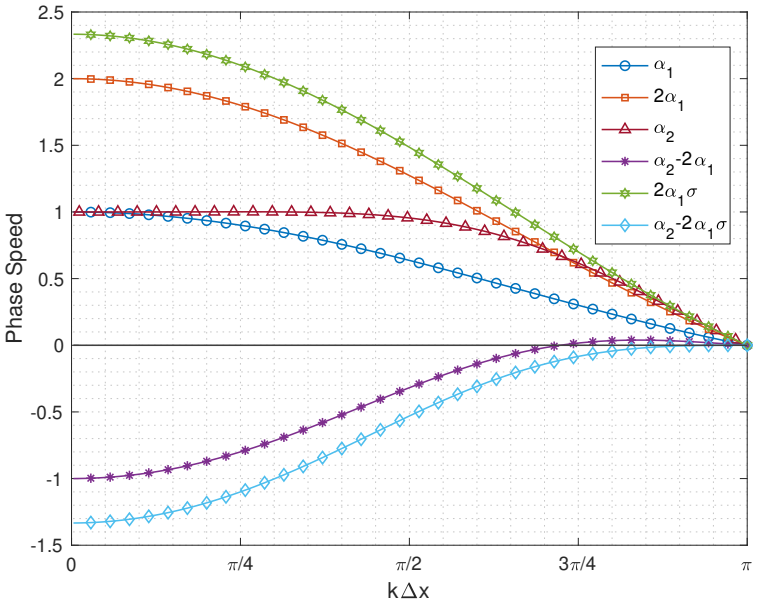


Fig. 2 Convection Speed, $\alpha_1 = \left(\frac{(k\Delta x)^*}{k\Delta x}\right)_{E2}$ and $\alpha_2 = \left(\frac{(k\Delta x)^*}{k\Delta x}\right)_{RDRP}$

B. Bounded Flow and Scaling Factors

For problems with periodic boundary conditions, the Fourier analysis presented above may be used to assess the global errors and calculate periodic scaling factors. However, this is not possible with other boundary conditions, such as a bounded flow. Hence an eigendecomposition is needed to calculate the numerical stability of a time step.

Consider a one-dimensional linear advection case discretized into N equal intervals in space. A second-order spatial difference is used on the LHS, and the governing equation is:

$$\begin{aligned}
\left(I + \Delta t \cdot \left(\left.\frac{\partial}{\partial x}\right|_{LHS}(c)\right)\right)\Delta u &= -\Delta t\left(\left.\frac{\partial}{\partial x}\right|_{RHS}(c)\right)u_i^n \\
\Delta u + \frac{\sigma\nu}{2}\left(\Delta u_{i+1} - \Delta u_{i-1}\right) &= -\Delta t\left(\left.\frac{\partial}{\partial x}\right|_{RHS}(c)\right)u_i^n \\
[A]\{\Delta u\} &= [B]\left\{u_i^n\right\}
\end{aligned} \tag{19}$$

Where $[A]$ and $[B]$ are $N \times N$ (sparse) matrices and Δu, u_i^n are N vectors representing the value of the function at the nodes $x_i = i/N$. To analyze the stability of a time step, re-write Eq. (19) as:

$$\begin{aligned}
[A]\{\Delta u_i\} &= [B]\{u_i^n\} \\
\left\{\frac{\Delta u_i}{u_i^n}\right\} &= [A]^{-1}[B] \\
\frac{u_i^{n+1}}{u_i^n} - [1] &= [A]^{-1}[B] \\
\frac{u_i^{n+1}}{u_i^n} &= [A]^{-1}[B] + [1] \\
&= [C]
\end{aligned} \tag{20}$$

The eigenvalues are calculated as follows:

$$\left|\frac{u_i^{n+1}}{u_i^n}\right| = |\lambda\,[C]| \tag{21}$$

If the magnitude of the eigenvalues is less than or equal to one, then the time step is stable. Otherwise, error terms will amplify, and the time step will be unstable. Note that matrices $[A]$ and $[B]$ are functions of the CFL, ν. Therefore the eigenvalues obtained are also functions of ν.

For this analysis, all schemes use the Hixon et al. 'ghost point correction' method, [29, 30] in which the governing equations are directly solved at the boundary points. As described in the reference, the spatial derivatives are first computed for all directions, using the '*noBC*' stencils in the direction normal to the boundary. The boundary conditions of:

$$\left.\frac{\partial u}{\partial x}\right|_i^{BC} = 0 \tag{22}$$

is then applied as a 'correction' to the normal derivative at the boundary, modifying the flux value at a nonphysical' ghost point' outside the computational domain. This modified ghost point value affects the normal derivative at every grid point whose spatial derivative uses the ghost point value. These corrections are given in terms of the boundary normal derivative correction on the appropriate grid line in the reference [29, 30]. Values of the bounded scaling factors are calculated numerically and shown in Table 1.

C. LU-SGS Stability

Yoon and Jameson's LU-SGS numerical time-step stability is analyzed in this section. The eigendecomposition setup explained in Section V.B is used again here. Consider a one-dimensional linear advection case that is discretized into N equal intervals in space; the LU-SGS method is written as:

$$LD^{-1}U\Delta u = -\Delta t \{RHS\}_i^n \tag{23}$$

note that using the linear advection equation, the jacobian of the flux, A, and the corresponding eigenvalue are:

$$\begin{aligned} A &= \frac{\partial E}{\partial Q} = c \\ \epsilon_A &= c \end{aligned} \tag{24}$$

giving the decomposed positive and negative eigenvalues, A^+ and A^-, respectively:

$$\begin{aligned} A^+ &= \frac{1}{2}(A + \epsilon_A I) = c \\ A^- &= \frac{1}{2}(A - \epsilon_A I) = 0 \end{aligned} \tag{25}$$

thus:

$$\begin{aligned} L &= I + \alpha\Delta t \left(D_x^- A^+\right) \\ D &= I + \alpha\Delta t \left(A^+\right) \\ U &= I + \alpha\Delta t \left(A^+\right) \end{aligned} \tag{26}$$

where D_x^- is a *first order backward difference operator*. To analyze the stability of a time step, re-write Eq. (23) as:

$$\begin{aligned} [L][D]^{-1}[U]\{\Delta u_i\} &= [B]\{u_i^n\} \\ [A]\{\Delta u_i\} &= [B]\{u_i^n\} \\ \frac{u_i^{n+1}}{u_i^n} &= [A]^{-1}[B] + [1] \\ &= [C] \end{aligned} \tag{27}$$

The eigenvalues are calculated as:

$$\left|\frac{u_i^{n+1}}{u_i^n}\right| = |\lambda\,[C]| \tag{28}$$

If the magnitude of the eigenvalues is less than or equal to one, then the time step is stable. Values of the periodic and bounded scaling factors for the LU-SGS scheme are calculated numerically and shown in Table 1.

Table 1 Periodic and Bounded Flow Scaling Factors

	σ_{E2}	σ_{E4}	σ_{E6}	σ_{DRP}	σ_{RDRP}	σ_{C4}
Periodic	*1.0*	*1.0*	$11/10$	*1.166823*	$7/6$	*1.49982*
LUSGS-Periodic	*1.0*	*1.0*	*1.0*	*1.0*	*1.0*	*1.0*
Bounded	*1.0*	*2.12179*	*6.3527*	*4.97487*	*6.36206*	*3.80018*
LUSGS-Bounded	*1.0*	*1.0*	*1.0*	*1.0*	*1.0*	*1.0*

D. Upwinding Connection to Central Differences

Perhaps the most striking result emerging from Table 1 is that the LU-SGS approximation does not require scaling factors where the proposed preconditioned matrix does. A closer look at the difference between both schemes will explain why this is the case. The LU-SGS scheme uses an upwinding scheme to approximate the space differences, whereas the proposed preconditioned scheme uses central differences. In previous work by Thomas Pulliam [31], it was shown that there is an upwinding connection between artificial dissipation and central differences. Using a first-order forward and backward difference as an approximation to the first derivative:

$$\begin{aligned} D_x^- &= \frac{I - \varepsilon^{-1}}{\Delta x} \\ D_x^+ &= \frac{\varepsilon^{+1} - I}{\Delta x} \end{aligned} \tag{29}$$

where ε^j is the shift operator i.e. $\varepsilon^{\pm j} = u_i = u_{i\pm j}$. A second-order central derivative can be derived since:

$$\frac{D_x^- + D_x^+}{2} = \frac{\varepsilon^{+1} - \varepsilon^{-1}}{2\Delta x} \tag{30}$$

On the other hand, a second-order dissipation operator can also be derived since:

$$\frac{D_x^- - D_x^+}{2} = \frac{-\varepsilon^{+1} + 2I - \varepsilon^{-1}}{2\Delta x} \tag{31}$$

Which allows a second-order central difference to be re-written as a first-order backward difference minus a second-order dissipation. Hence if the LU-SGS is upwinding (central + dissipation) and requires no scaling factors, then the proposed preconditioned approximation, which uses central differences and requires scaling factors, can be modified to behave like the LU-SGS scheme. This finding suggests stability is possible in a central differencing scheme if dissipation is added implicitly. The view is supported by FDL3DI [26–28], who have introduced implicit time marching with higher-order differencing stencils and obtained stability due to the addition of implicit dissipation.

VI. Numerical Results for Benchmark Problems

One way to analyze the stability of an implicit time step was shown in Section V; form the associated matrices using a finite number of points. Then use computer software to compute the eigenvalues, which can be used to quantify the stable nature of each scheme. While this provides at least a first-order measure of linear stability, the analysis is not rigorous or general. Hence, real-life nonlinear benchmark steady problems are studied in this section to validate the conclusions from the eigenvalue analysis.

In Computational Fluid Dynamics and Computational Aeroacoustics, emphasis is given to calculating both the steady and unsteady flow. Various numerical schemes are developed to understand and predict noise generation and propagation physics correctly. To verify the correct implementation of those schemes and their performances, a range of validation problems with solutions are listed in the CAA Workshops [32–34].

This paper studies two problems from the Third CAA workshop: *Category 1 Problem 1* and *Category 1 Problem 2* [34]. Category 1 problems deal with internal propagation; the propagation of sound through a narrow passage with flow exists in many applications. Problem 1 models the upstream propagation of sound through a nozzle with near sonic

conditions. The Quasi-1D nonlinear Euler equations are used to solve this problem:

$$\frac{\partial}{\partial t}\begin{Bmatrix}\rho\\ \rho u\\ E_{tot}\end{Bmatrix}+\frac{\partial}{\partial x}\begin{Bmatrix}\rho u\\ \rho u^2+p\\ u(E_{tot}+p)\end{Bmatrix}+\frac{1}{A}\frac{\partial A}{\partial x}\begin{Bmatrix}\rho u\\ \rho u^2\\ u(E_{tot}+p)\end{Bmatrix}=0 \tag{32}$$

For the time step, the CFL parameter ν is defined as:

$$\nu_{physical} = \frac{(|u_i|+c_i)\,\Delta t}{\Delta x}\cdot(k\Delta x)^* \tag{33}$$

The area of the duct is given by:

$$A(x) = \begin{cases}0.536572 - 0.198086e^{(-ln2)\left(\frac{x}{0.6}\right)^2} & x\geq 0\\ 1.0 - 0.661514e^{(-ln2)\left(\frac{x}{0.6}\right)^2} & x\leq 0\end{cases} \tag{34}$$

on a computational domain of $-10 \leq x \leq 10$ while using 401 grid points.

The numerical solver for this work uses an implicit time marching scheme with the preconditioned matrix on LHS. The LHS spatial difference is coupled with scaling factors for stability. While the following spatial differences are used on the RHS:

- Second Order (E2)
- Fourth Order (E4)
- Sixth Order (E6)
- DRP (DRP)
- RDRP (RDRP)
- Prefactored Fourth Order Compact (C4), [35]

In order to accurately predict the flow, it is crucial to apply the boundary conditions correctly; otherwise, the boundaries may (and will) generate spurious fluctuations and thereby contaminate the entire solution. Thompson's nonlinear boundary conditions are used in this paper (referred to as characteristic boundaries in the literature) [36, 37].

A. Category 1 Problem 1: Propagation of Sound Waves through a Transonic Nozzle

The problem is the upstream propagation of an acoustic wave through a transonic, nearly choked nozzle flow. The mean flow is set as follows:

$$\begin{Bmatrix}\overline{\rho}\\ \overline{u}\\ \overline{p}\end{Bmatrix} = \begin{Bmatrix}1.0\\ 0.4\\ \frac{1}{\gamma}\end{Bmatrix}$$

An explicit tenth-order Kennedy and Carpenter's artificial dissipation operator that provides damping for the inviscid nonlinear calculations is added to the equation [38]. The initial condition is set to the mean flow values to solve the steady mean distribution.

In this test case, an inviscid flow travels through a converging-diverging nozzle. This is a fully subsonic flow since there is no discontinuity in pressure. A pressure drop is expected as the flow accelerates through the throat and then rises again as the flow backs down in Fig. 3. The convergence of each scheme was determined by calculating the residual of the fluxes. Figure 4 shows the residual vs. iteration for the schemes. The figure indicates converged solutions for the various RHS difference formulation. A value of $\Delta t \to \infty$ is set to reduce the LHS schemes to a Newton iteration.

The second-order differencing stencil had the quickest convergence rate, as seen in Fig. 4a. Using second-order differences on the RHS yields a quicker convergence rate since no approximation is made to solve the equations. A larger approximation (such as using an RDRP scheme on the RHS) yields a larger factorization error, resulting in slower convergence rates.

Figure 4b compares the high-order scheme's convergence rate (RDRP) vs. a low-order scheme. Adding implicit dissipation to an unfactored scheme naturally enhances stability. As noted in the figure, this results in a high and quick convergence rate. When LU-SGS is coupled with the second order, it can be proven that the approximation matrix is nearly equivalent to an unfactored scheme; hence both schemes have similar behaviors. Factorization errors are much larger when using a higher-order differencing scheme on the RHS, such as RDRP. LU-SGS is favorable over scaling factors and dissipation in this case, as seen in Fig. 4b.

Once the steady mean-flow results are obtained, the upstream propagating perturbation starts at the boundary, and the unsteady results are studied. Here, a small amplitude acoustic wave is generated way downstream and propagates upstream through the narrow passage of the nozzle throat:

$$\begin{Bmatrix} \rho' \\ u' \\ p' \end{Bmatrix}_{outflow} = \kappa \begin{Bmatrix} 1 \\ -1 \\ 1 \end{Bmatrix} \cos\left[\omega\left(\frac{x_{outflow}}{1 - M_{outflow}} + t\right)\right]$$

where $\kappa = 10^{-5}$ and $\omega = 0.6\pi$.

Figure 5 shows the maximum pressure perturbation using vs. flow domain. The unsteady solution, taken once a periodic steady state is reached, uses 32 steps per cycle. An optimized 2^{nd} order optimized backward difference formulation [39] is used to approximate the time derivative:

$$\left.\frac{\partial Q}{\partial t}\right|_i = \frac{10Q_i^{n+1,l} - 15Q_i^n + 6Q_i^{n-1} - Q_i^{n-2}}{6\Delta t} \tag{35}$$

The unsteady CFL is calculated as:

$$\nu_{physical} = \frac{(|u_i| + c_i)}{\Delta x} \cdot \left(\frac{2\pi}{\omega} \cdot \frac{1}{\cdot \text{Steps Per Cycle}}\right) \cdot (k\Delta x)^*$$

For each implicit solve, the solution is converged down to machine precision. It is seen that second-order spatial difference cannot accurately predict the pressure perturbation where the higher-order schemes agree very well with the exact. The value of $\nu_{physical}$ for each test case is shown in Table 2.

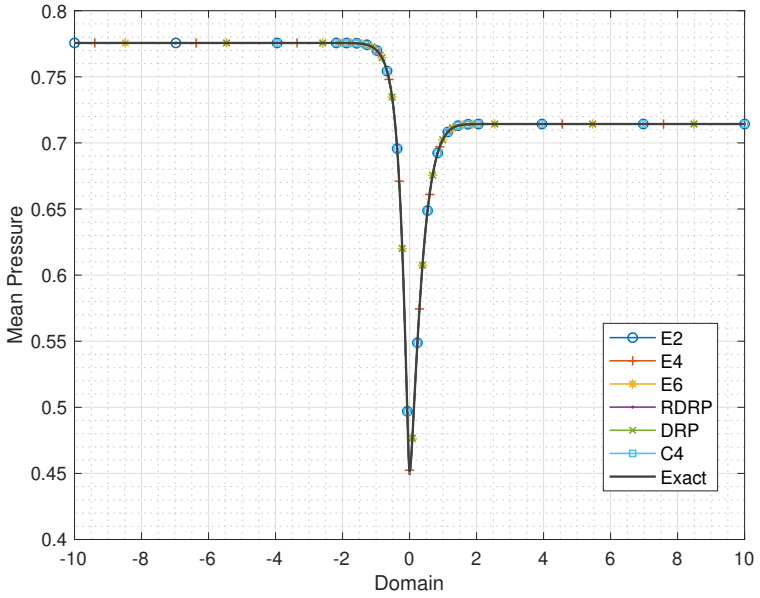
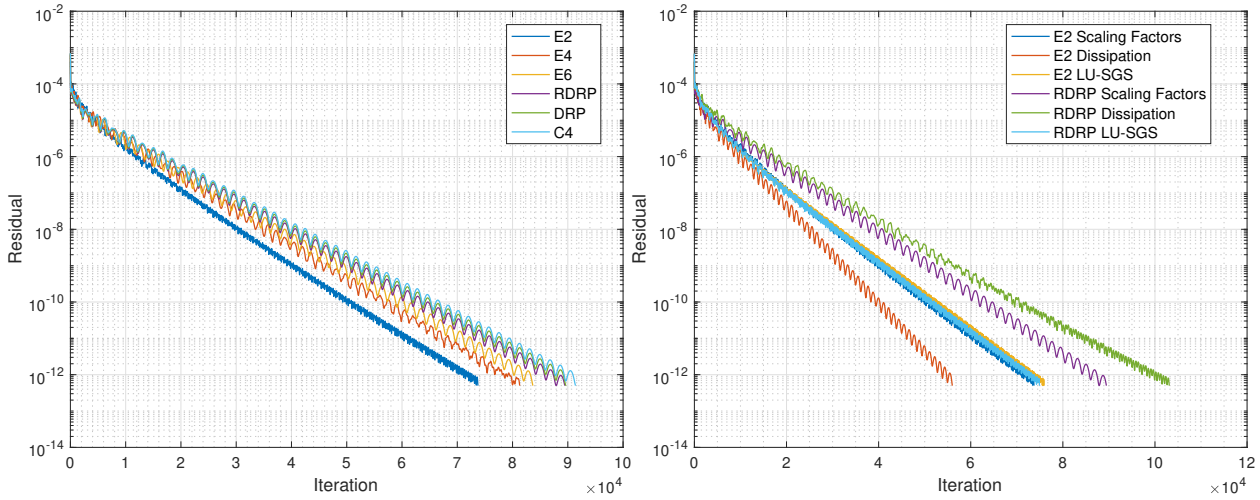


Fig. 3 Domain vs. Mean Pressure for Category 1 Problem 1

Table 2 CFL Values for Category 1 Problem 1

	E2	**E4**	**E6**	**DRP**	**RDRP**	**C4**
$\nu_{physical}$	*12.521*	*17.518*	*19.858*	*20.834*	*20.834*	*22.237*



(a) Preconditioned Scheme with Scaling Factors **(b) E2 vs RDRP**

Fig. 4 Rate of Convergence (Residual vs. Iteration) for Category 1 Problem 1

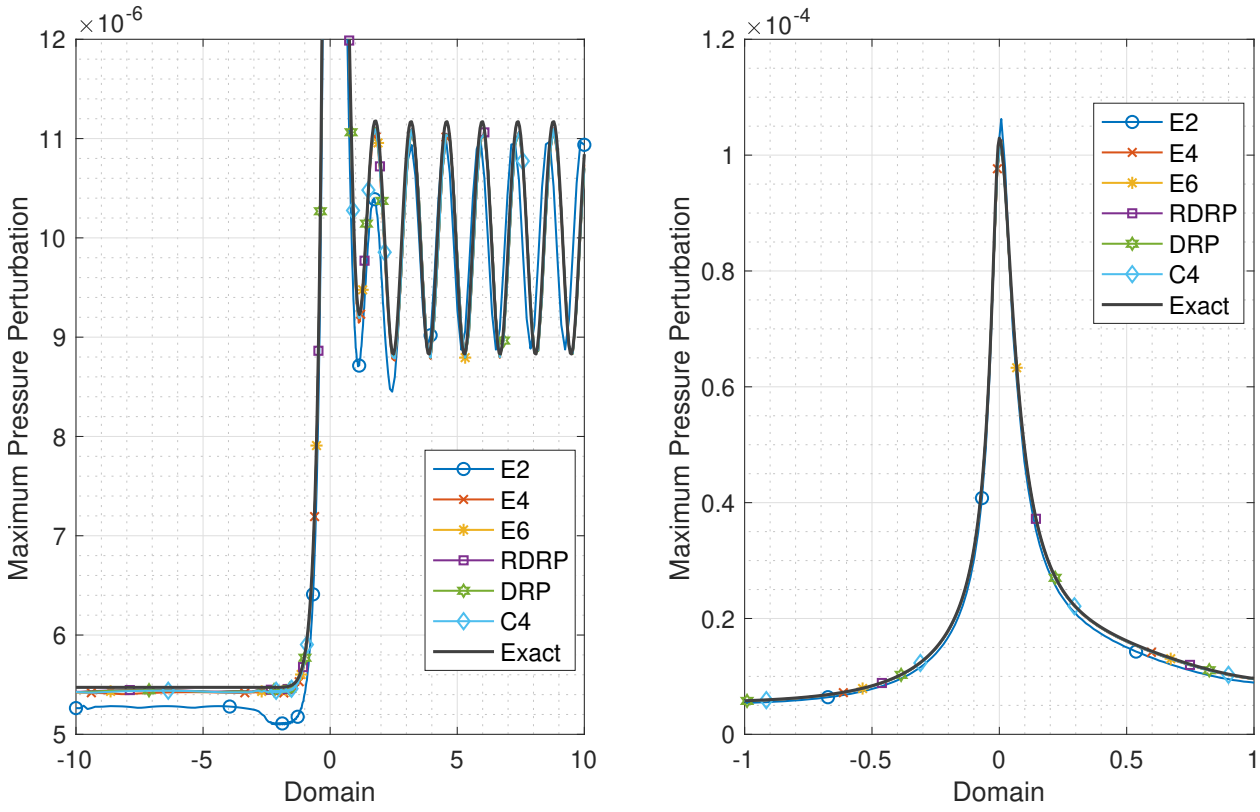


Fig. 5 Maximum Pressure Disturbance for Category 1 Problem 1

B. Category 1 Problem 2: Shock-Sound Interaction

The problem to be solved is the upstream propagation of an acoustic wave through a shock wave in a convergent-divergent nozzle. The mean flow conditions are:

$$\begin{Bmatrix} \overline{\rho} \\ \overline{u} \\ \overline{p} \end{Bmatrix} = \begin{Bmatrix} 1.0 \\ 0.2006533 \\ \frac{1}{\gamma} \end{Bmatrix}$$

At the outflow boundary, the pressure is set such that it creates a shock:

$$p_{exit} = 0.6071752$$

An explicit tenth-order Kennedy and Carpenter's artificial dissipation operator that dampens the inviscid nonlinear calculations is added to the equation [38]. The initial condition is set to the mean flow values to solve the steady mean distribution.

In this problem, a supersonic shock is present downstream of the nozzle. A shock can be seen in a steep increase in pressure in Fig. 6. The same geometry as problem one is also used here with the same governing equations. Figure 7 shows the residual vs. iteration for the schemes. The figure indicates converged solutions for the various RHS difference formulation. A value of $\Delta t \rightarrow \infty$ is set to reduce the LHS schemes to a Newton iteration.

The second-order differencing stencil had the quickest convergence rate, as seen in Fig. 4a. Using second-order differences on the RHS yields a quicker convergence rate since no approximation is made to solve the equations. A larger approximation (such as using an RDRP scheme on the RHS) yields a larger factorization error, resulting in slower convergence rates.

Figure 4b compares the high-order scheme's convergence rate (RDRP) vs. a low-order scheme. Adding implicit dissipation to an unfactored scheme naturally enhances stability. As noted in the figure, this results in a high and quick convergence rate. When LU-SGS is coupled with the second order, it can be proven that the approximation matrix is nearly equivalent to an unfactored scheme. However, in this specific test case, a shock has been formed in the solution. Initially, the LU-SGS scheme's convergence rate follows that of the subsonic case (Category 1 Problem 1). The flow then evolves into two subsonic regions connected via a supersonic region. In a supersonic flow region, no flow disturbances can propagate upstream, which typically enhances the convergence rate for an unfactored scheme. However, the LU-SGS scheme factors the numerics into positive running eigenvalue, A^+, and negative running eigenvalues, A^-. When this approximation is coupled with a regime where flow disturbances can only propagate upstream, an increase in factorization errors is present; thus hindering the convergence rate.

Once the steady mean-flow results are obtained, the upstream propagating perturbation starts at the boundary, and the unsteady results are studied. The acoustic wave at the upstream boundary is defined as:

$$\begin{Bmatrix} \rho' \\ u' \\ p' \end{Bmatrix}_{inflow} = \kappa \begin{Bmatrix} 1 \\ -1 \\ 1 \end{Bmatrix} \sin\left[\omega\left(\frac{x_{inflow}}{1 - M_{inlet}} + t\right)\right]$$

where $\kappa = 10^{-5}$ and $\omega = 0.6\pi$.

Figure 8 shows the perturbation pressure history at the exit vs. time. The unsteady solution, taken once a periodic steady state is reached, uses 32 steps per cycle. An optimized 2^{nd} order optimized backward difference formulation [39] is used to approximate the time derivative seen in Eq. (35). Where the unsteady CFL is calculated as:

$$\nu_{physical} = \frac{(|u_i| + c_i)}{\Delta x} \cdot \left(\frac{2\pi}{\omega} \cdot \frac{1}{\cdot\text{Steps Per Cycle}}\right) \cdot (k\Delta x)^*$$

For each implicit solve, the solution is converged down to machine precision. It is seen that second-order spatial difference cannot accurately predict the pressure perturbation where the higher-order schemes agree very well with the exact. The value of $\nu_{physical}$ for each test case is shown in Table 3.

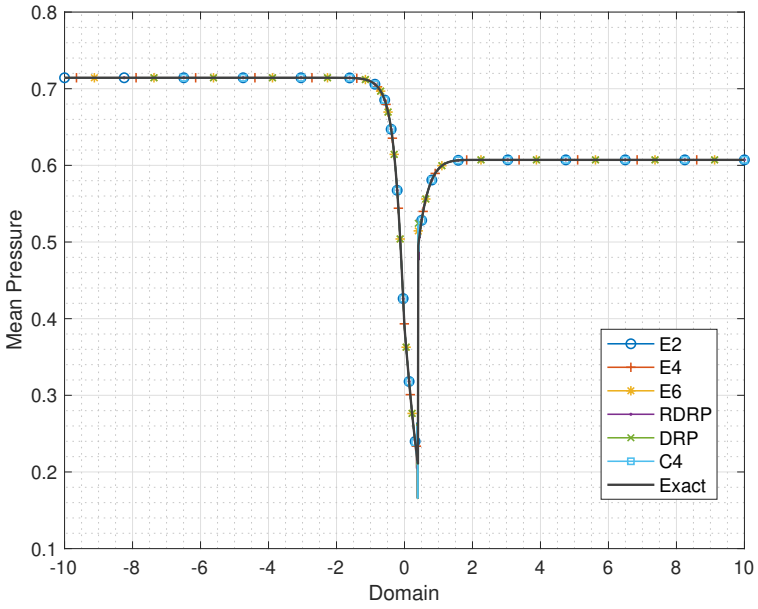
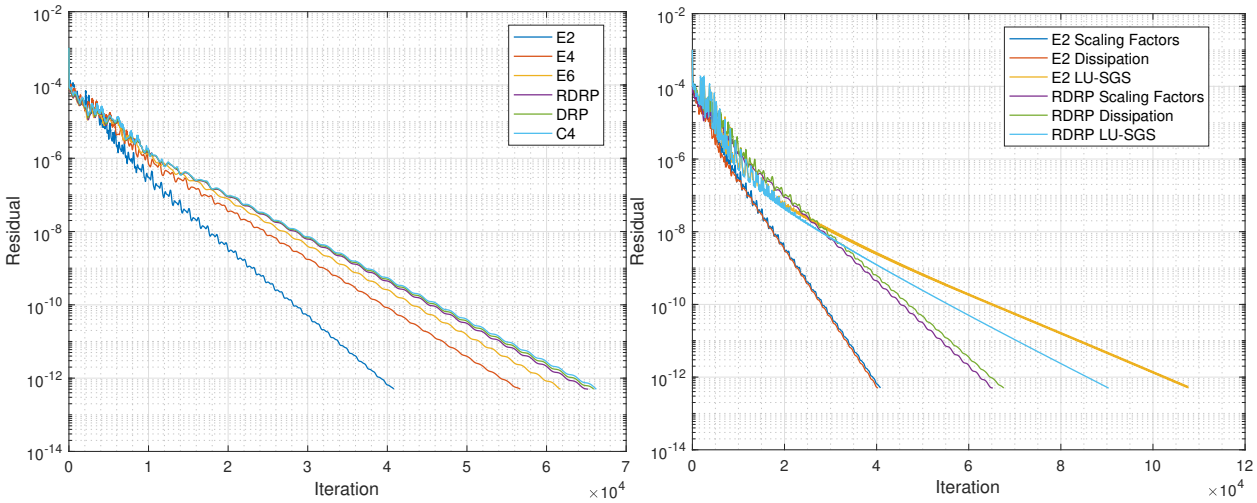


Fig. 6 Domain vs. Mean Pressure for Category 1 Problem 2



(a) Preconditioned Scheme with Scaling Factors **(b) E2 vs RDRP**

Fig. 7 Rate of Convergence (Residual vs. Iteration) for Category 1 Problem 2

Table 3 CFL Values for Category 1 Problem 2 from the CAA Workshop

	E2	**E4**	**E6**	**DRP**	**RDRP**	**C4**
$\nu_{physical}$	*14.792*	*20.815*	*23.713*	*24.922*	*24.922*	*25.118*

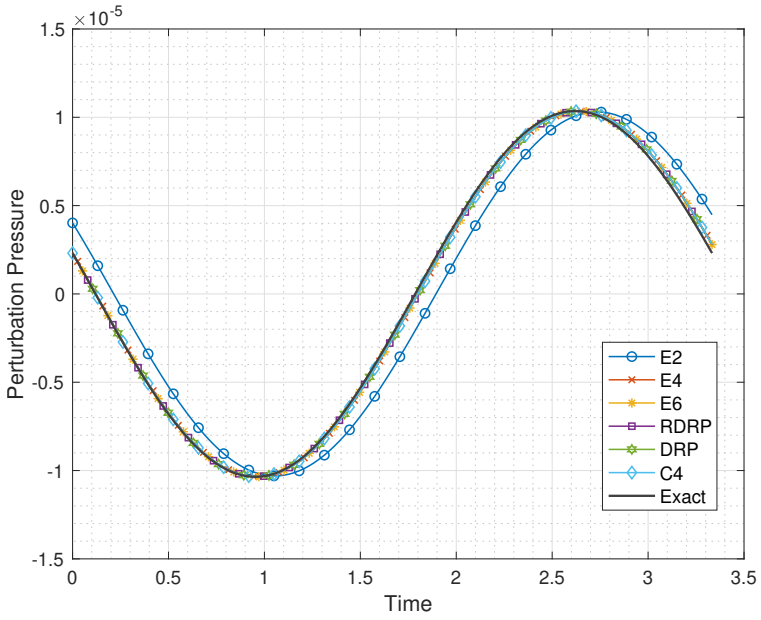


Fig. 8 Perturbation Pressure History at Exit Plane vs. Time for Category 1 Problem 2

VII. Conclusion and Future Work

The present research aimed to establish a highly efficient implicit solver using higher-order spatial differencing schemes. Through the use of scaling factors, the LHS matrix was replaced with an easier-to-solve matrix hence cutting down on the computational work needed for the iterative steady-state solver. A von-Neumann stability analysis was conducted to confirm the source of instability. Errors from the preconditioned matrix were scaled to overcome error from the physics of the governing equations and ensure overall stability. The proposed preconditioned matrix was later evaluated vs. multiple steady-state and unsteady CAA workshop problems. Numerical solutions agree very well with the exact solution.

Although the current study is based on one-dimensional problems, the findings suggest stability is possible for traditional implicit formulation if scaling factors are used on the LHS spatial difference. Future work will evaluate the proposed scheme's stability for multidimensional and viscous flow problems.

Acknowledgments

This work was supported by the NASA Advanced Air Transportation Technologies (AATT) Project. The authors would like to thank Dr. Edmane Envia of the NASA Glenn Research Center, who was the technical monitor for this work.

References

[1] Tam, C. K. W., "Computational aeroacoustics - Issues and methods," *AIAA J.*, Vol. 33, No. 10, 1995, pp. 1788–1796. https://doi.org/10.2514/3.12728.

[2] Lele, S. K., "Computational Aeroacoustics: A Review," *AIAA Paper 97-0018*, 1997.

[3] Tam, C. K., "Computational Aeroacoustics: An Overview of Computational Challenges and Applications," *Int. J. Comp. Fluid Dyn.*, Vol. 18, No. 6, 2004, pp. 547–567. https://doi.org/10.1080/10618560410001673551.

[4] Colonius, T., and Lele, S. K., "Computational aeroacoustics: progress on nonlinear problems of sound generation," *Prog. Aero. Sci.*, Vol. 40, No. 6, 2004, pp. 345–416. https://doi.org/https://doi.org/10.1016/j.paerosci.2004.09.001.

[5] Lomax, H., Pulliam, T., and Zingg, D., *Fundamentals of Computational Fluid Dynamics*, Scientific Computation, Springer Berlin Heidelberg, 2003.

[6] Maccormack, R., *Current status of numerical solutions of the Navier-Stokes equations*, chapter and pages. https://doi.org/10.2514/6.1985-32.

[7] Yoon, S., and Kwak, D., "Implicit Methods for the Navier-Stokes Equations," *Computing Systems in Engineering*, Vol. 1, No. 2, 1990, pp. 535–547. https://doi.org/https://doi.org/10.1016/0956-0521(90)90034-I, computational Technology for Flight Vehicles.

[8] Rizzetta, D. P., and Visbal, M. R., "Numerical Simulation of Separation Control for Transitional Highly Loaded Low-Pressure Turbines," *AIAA J.*, Vol. 43, No. 9, 2005, pp. 1958–1967. https://doi.org/10.2514/1.12376.

[9] Beam, R. M., and Warming, R. F., "An Implicit Factored Scheme for the Compressible Navier-Stokes Equations," *AIAA J.*, Vol. 16, No. 4, 1978, pp. 393–402. https://doi.org/10.2514/3.60901.

[10] Briley, W. R., and McDonald, H., "Solution of the three-dimensional compressible Navier-Stokes equations by an implicit technique," *Proceedings of the Fourth International Conference on Numerical Methods in Fluid Dynamics*, edited by R. D. Richtmyer, Springer Berlin Heidelberg, ????, pp. 105–110.

[11] Pulliam, T., "ARC2D - Efficient Solution Methods For The Navier-Stokes Equations (Cray Version)," 1994.

[12] Pulliam, T., and Steger, J., *Recent Improvements In Efficiency, Accuracy, and Convergence for Implicit Approximate Factorization Algorithms*, chapter and pages. https://doi.org/10.2514/6.1985-360.

[13] Anderson, W., Thomas, J., and Rumsey, C., *Extension and Applications of Flux-Vector Splitting to Unsteady Calculations on Dynamic Meshes*, chapter and pages. https://doi.org/10.2514/6.1987-1152.

[14] Kwak, D., Chang, J. L. C., Shanks, S. P., and Chakravarthy, S. R., "A Three-Dimensional Incompressible Navier-Stokes Flow Solver Using Primitive Variables," *AIAA J.*, Vol. 24, No. 3, 1986, pp. 390–396. https://doi.org/10.2514/3.9279.

[15] Derlaga, J. M., Jackson, C. W., and Buning, P. G., *Recent Progress in OVERFLOW Convergence Improvements*, chapter and pages. https://doi.org/10.2514/6.2020-1045.

[16] Yoon, S., and Jameson, A., "Lower-Upper Symmetric-Gauss-Seidel Method for the Euler And Navier-Stokes Equations," *AIAA J.*, Vol. 26, No. 9, 1988, pp. 1025–1026. https://doi.org/10.2514/3.10007.

[17] Heidmann, J. D., Ameri, A. A., Rigby, D. I., Garg, V. K., Fabian, J. C., Lucci, B. L., and Steinthorsson, E., "Object-Oriented Version of Glenn-HT Code Released: Glenn-HT2000," Sti, NASA Glenn Research Center Cleveland, OH, United States, September 7, 2013 2005.

[18] Jameson, A., and Shankaran, S., *An Assessment of Dual-Time Stepping, Time Spectral and Artificial Compressibility Based Numerical Algorithms for Unsteady Flow with Applications to Flapping Wings*, chapter and pages. https://doi.org/10.2514/6.2009-4273.

[19] Jameson, A., *Time dependent calculations using multigrid, with applications to unsteady flows past airfoils and wings*, chapter and pages. https://doi.org/10.2514/6.1991-1596.

[20] Shieh, C., and Morris, P., *High-order accurate dual time-stepping algorithm for viscous aeroacoustic simulations*, chapter and pages. https://doi.org/10.2514/6.1998-2361.

[21] Lele, S. K., "Compact finite difference schemes with spectral-like resolution," *J. Comp. Phys.*, Vol. 103, No. 1, 1992, pp. 16–42. https://doi.org/https://doi.org/10.1016/0021-9991(92)90324-R.

[22] Tam, C. K., and Webb, J. C., "Dispersion-Relation-Preserving Finite Difference Schemes for Computational Acoustics," *J. Comp. Phys.*, Vol. 107, No. 2, 1993, pp. 262–281. https://doi.org/https://doi.org/10.1006/jcph.1993.1142.

[23] Liu, Z., Huang, Q., Zhao, Z., and Yuan, J., "Optimized Compact Finite Difference Schemes with High Accuracy and Maximum Resolution," *Int. J. Aeroacoustics*, Vol. 7, No. 2, 2008, pp. 123–146. https://doi.org/10.1260/147547208784649464.

[24] Hixon, R., "Prefactored Small-Stencil Compact Schemes," *J. Comp. Phys.*, Vol. 165, No. 2, 2000, pp. 522–541. https://doi.org/https://doi.org/10.1006/jcph.2000.6631.

[25] Visbal, M., and Gaitonde, D., *Computation of aeroacoustic fields on general geometries using compact differencing and filtering schemes*, chapter and pages. https://doi.org/10.2514/6.1999-3706.

[26] Visbal, M. R., and Gaitonde, D. V., "Very High-Order Spatially Implicit Schemes for Computational Acoustics on Curvilinear Meshes," *J. Comp. Acoustics*, Vol. 09, No. 04, 2001, pp. 1259–1286. https://doi.org/10.1142/S0218396X01000541.

[27] Rizzetta, D. P., and Visbal, M. R., "Numerical Simulation of Separation Control for Transitional Highly Loaded Low-Pressure Turbines," *AIAA J.*, Vol. 43, No. 9, 2005, pp. 1958–1967. https://doi.org/10.2514/1.12376.

[28] Visbal, M., and Gaitonde, D., *Computation of Aeroacoustic Fields on General Geometries Using Compact Differencing and Filtering Schemes*, chapter and pages. https://doi.org/10.2514/6.1999-3706.

[29] Hixon, R., "Optimized Rational Dispersion Relation Preserving (R-DRP) Method for High-Accuracy Flow Simulations," *Aeroacoustics*, 2019. https://doi.org/10.2514/6.2019-2956.

[30] Hixon, R., Nallasamy, M., and Sawyer, S., "A method for the Implementation of Boundary Conditions in High-Accuracy Finite-Difference Schemes," *AIAA Paper 2005-0608*, 2005.

[31] Pulliam, T. H., "Artificial dissipation models for the Euler equations," *AIAA J.*, Vol. 24, No. 12, 1986, pp. 1931–1940. https://doi.org/10.2514/3.9550.

[32] Hardin, J. C., Ristorcelli, J. R., and Tam, C. K. W. e. (eds.), *ICASE/LaRC Workshop on Benchmark Problems in Computational Aeroacoustics (CAA)*, NASA Conference Publication 3300, 1995.

[33] Tam, C. K. W., and Hardin, J. C. (eds.), *Second Computational Aeroacoustics (CAA) Workshop on Benchmark Problems*, NASA Conference Publication 3352, 1997.

[34] Dahl, M. D. (ed.), *Third Computational Aeroacoustics (CAA) Workshop on Benchmark Problems*, NASA Conference Publication 2000-209790, 2000.

[35] Hixon, R., "Prefactored Small-Stencil Compact Schemes," *J. Comp. Phys.*, Vol. 165, No. 2, 2000, pp. 522–541. https://doi.org/https://doi.org/10.1006/jcph.2000.6631.

[36] Thompson, K. W., "Time Dependent Boundary Conditions for Hyperbolic Dystems," *J. Comp. Phys.*, Vol. 68, No. 1, 1987, pp. 1–24. https://doi.org/https://doi.org/10.1016/0021-9991(87)90041-6.

[37] Thompson, K. W., "Time-Dependent Boundary Conditions for Hyperbolic Systems, II," *J. Comp. Phys.*, Vol. 89, 1990, pp. 439–461.

[38] Kennedy, C., and Carpenter, M., "Comparison of Several Numerical Methods for Simulation of Compressible Shear Layers," 1998.

[39] Hixon, R., and Visbal, M., *Comparison of High-Order Implicit Time Marching Schemes for Unsteady Flow Calculations*, chapter and pages. https://doi.org/10.2514/6.2007-4324.