Journal of Computational Physics **156,** 272–299 (1999)
Article ID jcph.1999.6360, available online at http://www.idealibrary.com on IDEAL®

# Implicit, High-Resolution, Compact Schemes for Gas Dynamics and Aeroacoustics

John A. Ekaterinaris

*Nielsen Engineering and Research, 526 Clyde Avenue, Mountain View, California 94043-2212*
E-mail: ekaterin@nearinc.com

Received August 13, 1998; revised August 26, 1999

Implicit, high-order schemes are developed for time-accurate numerical solutions of hyperbolic equation systems. High-order spatial accuracy for the implicit operators is obtained at no additional computing cost by performing compact differentiation. The resulting alternating direction implicit and unfactored algorithms yield improved dispersion characteristics compared to second-order accurate in space implicit schemes which makes them suitable for high-resolution numerical simulations in gas dynamics and computational aeroacoustics. First, a fourth-order accurate in space implicit, factorized scheme, which requires block-tridiagonal matrix inversion, is presented. Next, a class of implicit factorized schemes, which require scalar matrix inversions, is presented. Higher order of accuracy in space of the implicit operators is achieved at the expense of inverting scalar matrices with larger bandwidth. Finally, extensions to unfactored algorithms, which use upwind compact schemes, are obtained. The proposed high-order schemes can be implemented with little modification of existing second-order accurate in space, implicit CFD methods. The efficiency, accuracy, and convergence characteristics of the new, high-resolution implicit schemes are demonstrated by their implementation for test problems. 

*Key Words:* hyperbolic equations; implicit, high-order methods; compact schemes.

## 1. INTRODUCTION

High-resolution, time-accurate numerical solutions of hyperbolic equations are of interest in many fields, such as direct numerical simulations (DNS) and large eddy simulations (LES) in gas dynamics, computational aeroacoustics, and computational electromagnetics. In numerical solution of these problems, time integration is often performed with explicit methods, which are the schemes of choice when the time scales of interest are small and comparable to spatial scales. In many applications, however, one deals with lower frequency phenomena or simulations which must resolve a wide spectrum of spatial scales, cases for which explicit schemes lead to very large computing times. For these cases, one needs to develop high-order accurate, implicit methods where the time step is not severely limited



0021-9991/99 $30.00

by the cell size. The purpose of this paper is to present a new class of fully implicit, compact, high-order accurate in space schemes for the numerical solution of multidimensional, nonlinear problems of gas dynamics and linear problems of computational aeroacoustics and electromagnetics. For problems in gas dynamics we consider numerical solutions of the full, time-dependent Euler equations. Aeroacoustic problems require numerical solution of the linearized Euler equations.

In recent years, there has been an increased interest in developing high-order, time-accurate methods [30, 42, 15, 2, 40] for DNS and LES of high Reynolds number, turbulent, compressible flows. These numerical simulations involve long integration times, a requirement which makes preservation of phase characteristics important. It was pointed out by Hu *et al.* [22] that even high-order accurate in time explicit methods applied to high-order accurate space discretizations can cause dispersion errors. These errors and the numerical stability characteristics of the method may be improved with the optimized methods they proposed [22]. The situation is worse when the order of accuracy is different in the spatial and temporal discretizations. Apart from time integration, large dispersion errors are also caused by low-order space discretization or insufficient grid resolution in numerical simulations performed either with centered or upwind schemes. In many cases, therefore, use of methods with second-order accuracy in time and high-order accuracy in space [17] is a reasonable compromise because time marching with smaller time steps, needed for numerical stability and/or prevention of dispersion errors, moderately increases the computing time but not the storage requirements. As a result, globally high-order accurate in space methods provide an attractive alternative to high grid resolution because it is preferable to perform simulations with a smaller number of grid points and a larger number of time steps both for memory management and computational efficiency. Direct simulations of wall-bounded flows in domains with stretched meshes [30, 40] use factorized, implicit schemes. In these computations, the convective terms were high-order accurate and the implicit operators were low-order accurate in space. Newton-type subiterations were used within each physical time step during the time-accurate computations to obtain time accuracy by removing the factorization errors and to reduce errors caused by the low order discretization in space of the implicit operators. High-order accurate in space implicit schemes proposed in this paper can achieve the same level of accuracy with fewer subiterations, and since they introduce smaller dispersion errors are more suitable for long time integration.

Accurate, steady-state and unsteady CFD numerical simulations, on the other hand, can also be achieved with less computing effort when the space discretization is performed with a high order of accuracy. During the past decades many methods have been developed which can be used to evaluate the convective terms with a high order of accuracy. These methods use centered schemes [18] with added artificial dissipation [28] or spatial filtering [23, 40]. Shock capturing, high-resolution, TVD upwind methods with flux-vector splitting [34], or flux-difference splitting [31, 26] and high-order ENO schemes [35] are well-developed. Modern CFD codes perform implicit integration with unfactored relaxation schemes [10, 32, 41] or with alternating direction implicit (ADI) methods, such as the Beam–Warming [4] algorithm. The Beam–Warming algorithm uses either central second-order accurate differentiation in space [4] or first-order accurate upwinding [13, 30]. The accuracy, efficiency, and convergence characteristics of implicit schemes can be improved by the application of high-order accurate in space discretization methods. Unfactored operators can be inverted with relaxation methods [10, 41] or GMRES methods [32]. It is straightforward to implement high-order space discretization with GMRES methods. These methods, however, require

very large memory resources and they are not practical for three-dimensional applications. High-order spatial accuracy also increases the computing cost of ADI schemes and relaxation methods. The additional expense due to loss of computational efficiency makes use of higher order implicit discretization impractical for three-dimensional solutions. Therefore high-resolution schemes which provide high order of accuracy in space for both the implicit and explicit operators, such as the fourth-order compact schemes recently presented by Abarbanel and Kumar [1] and Yee [42] and the implicit high-order schemes developed in this paper, can be used to achieve faster convergence and high resolution.

Recently there has been an increased interest in the development of methods suitable for numerical simulations of unsteady phenomena related to wave propagation in a variety of disciplines, such as aeroacoustics and electromagnetics. As a result, computational aeroacoustics (CAA) [36] and computational electromagnetics (CEM) [33] have emerged as new, rapidly evolving fields in computational sciences. Both CAA and CEM require long integration times for far-field predictions. Recently, Carpenter and Kennedy [9] developed explicit, high-order, low-storage, Runge–Kutta methods for aeroacoustics and wave propagation. Hu *et al.* [22] developed optimized, explicit, Runge–Kutta schemes with better dissipation and dispersion characteristics than the traditional, third- and fourth-order Runge–Kutta methods. These optimized schemes chose the coefficients in order to obtain the maximum possible order of accuracy in time for a given number of stages. Optimized, explicit methods have increased stability limits but they are suitable only for aeroacoustic problems where the grid spacing is much larger than the grid spacing typically used near solid walls in fluid dynamics viscous flow simulations. Numerical schemes with high order of accuracy in space are also needed in order to enable solutions with a reasonable number of grid points per wavelength and small dispersion errors. Tam and Webb [37] presented a low dispersion scheme for the numerical solution of the linearized Euler equations. Improved dispersion characteristics in wave propagation problems were also demonstrated by Zingg *et al.* [43] and Haras and Ta'Asan [20] who optimized high-order, centered, finite-difference schemes. These schemes, which are analyzed in Refs. [37, 43, 16, 20], chose the coefficients of the differencing formulas so that the wave space resolution characteristics of the scheme are improved. Low dispersion and dissipation criteria necessary for aeroacoustic computations are also fulfilled by the fourth-order accurate in space predictor corrector-type scheme of Ref. [17]. This scheme was developed by Gottlieb and Turkel [17], as an extension of the MacCormack scheme known as the 2-4 MacCormack scheme, and it was used [21] for aeroacoustics applications. In this paper we develop implicit schemes which achieve high-order accuracy in space for the Euler equations of gas dynamics and the linearized Euler equations for aeroacoustics. Fourth-order accuracy in space is obtained at no additional computing cost compared to the second-order accurate, ADI algorithm of Beam and Warming [4], because both methods require block tridiagonal matrix inversions. The proposed high-order implicit schemes can be used for wave propagation, time advancement of unsteady numerical simulations with large time steps, or faster convergence to a steady state.

The paper is organized as follows: Section 2 presents the Euler equations of gas dynamics and the linearized Euler equations for sound propagation in the presence of a uniform free stream used in aeroacoustic simulations. Section 3 starts the preliminaries with the presentation of high-resolution space discretization and implicit time integration methods. Then, the new, fourth-order accurate in space, implicit, compact, block tridiagonal ADI algorithm is presented. In Section 4 we present the compact schemes for the diagonalized ADI algorithm. Section 5 suggests extensions of the high-order, compact methods to unfactored algorithms. Section 6 presents a stability analysis of the ADI, implicit fourth-order, compact

scheme. Implementation of the new schemes is discussed in Section 7. Section 8 is devoted to illustrative examples where we compare analytical solutions with computed results and demonstrate the improved convergence characteristics of the new implicit schemes in computations with stretched meshes for flows of practical interest. In Section 9, the concluding remarks are presented.

## 2. GOVERNING EQUATIONS

For the schemes developed in this paper, it is sufficient to consider, without loss of generality, only the two-dimensional form of the governing equations for gas dynamics and aeroacoustics. The Euler equations of gas dynamics in strong conservation law form and Cartesian coordinates are

$$\frac{\partial \mathbf{q}}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{q})}{\partial x} + \frac{\partial \mathbf{g}(\mathbf{q})}{\partial y} = 0, \tag{1}$$

where

$$\mathbf{q} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ e \end{pmatrix}, \qquad \mathbf{f} = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(e+p) \end{pmatrix}, \qquad \mathbf{g} = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(e+p) \end{pmatrix}. \tag{2}$$

Here, $\rho$ is the density; $u$, $v$ are the Cartesian velocity components along the $x$ and $y$ coordinate directions, respectively; $e$ is the total energy; and $p$ is the pressure which, for a calorically perfect gas, is related to the other variables through the equation of state as $p = (\gamma - 1)[e - 0.5\rho(u^2 + v^2)]$. In curvilinear, body-fitted coordinates $(\xi, \eta)$ used for problems with nontrivial geometries the Euler equations become

$$\frac{\partial \mathbf{Q}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{Q})}{\partial \xi} + \frac{\partial \mathbf{G}(\mathbf{Q})}{\partial \eta} = 0, \tag{3}$$

where $\mathbf{Q} = J^{-1}[\rho, \rho u, \rho v, e]^T$ is the solution vector, $J$ is the Jacobian of the coordinate transformation, and $\mathbf{F}$ and $\mathbf{G}$ are the flux vectors in curvilinear coordinates. For example, $\mathbf{F} = J^{-1}(\xi_x \mathbf{f} + \xi_y \mathbf{g}) = \mathbf{F} = J^{-1}[\rho U,\ \rho u U + \xi_x p,\ \rho v U + \xi_y p,\ (e+p)U - \xi_t p]^T$, where $U$ is the contravariant velocity component $U = \xi_x u + \xi_y v + \xi_t$.

Similarly, the two-dimensional, linearized Euler equations, which describe propagation of acoustic disturbances in the presence of a uniform mean flow with Mach number $M_\infty^2 = M_x^2 + M_y^2$ are

$$\frac{\partial \mathbf{q}'}{\partial t} + \frac{\partial \mathbf{f}'}{\partial x} + \frac{\partial \mathbf{g}'}{\partial y} = 0, \tag{4}$$

where

$$\mathbf{q}' = \begin{pmatrix} \rho' \\ u' \\ v' \\ p' \end{pmatrix}, \qquad \mathbf{f}' = \begin{pmatrix} M_x \rho' + u' \\ M_x u' + p' \\ M_x v' \\ M_x p' + u' \end{pmatrix}, \qquad \mathbf{g}' = \begin{pmatrix} M_y \rho' + v' \\ M_y u' \\ M_y v' + p' \\ M_y p' + v' \end{pmatrix}. \tag{5}$$

In these equations, $\rho'$, $u'$, $v'$, $p'$ are the acoustic density, velocities, and pressure, respectively. $M_x$ and $M_y$ are the velocity components of the uniform mean flow along the $x$ and $y$ coordinate directions, respectively. The linearized Euler equations in curvilinear coordinates are given in Appendix A.

The nonlinear Euler equations are integrated in time to obtain steady-state and unsteady numerical solutions of compressible flows. For a steady-state solution it is desirable to use large time steps for fast convergence. Time-dependent flow problems and aeroacoustic computations require time-accurate solutions, and it is desirable to use time steps determined by the physics rather than the numerics. Therefore, in many cases use of explicit algorithms, which have time step restrictions due to CFL stability, is impractical. For wave propagation problems and long time integration, on the other hand, phase preservation and minimization of dispersion errors becomes important, and development of high-order implicit schemes which fulfill all these requirements is shown next.

## 3. HIGH-RESOLUTION SCHEMES

### 3.1. *High-Resolution in Time and Space Discretizations*

Numerical approximations to the convective terms of the nonlinear Euler equations of gas dynamics and their linearized form which describes sound propagation in the presence of a uniform free stream give rise to dispersive errors. These errors occur [39] because in the discrete approximation the phase velocity of propagation, of the harmonic components into which an arbitrary function is resolved by Fourier analysis, is not constant. For central difference schemes, the dispersive errors are contributed mostly by the odd order derivative terms of the modified equation. For simple sinusoidal solutions of linear problems, [39] the deviation of the numerical phase velocity of propagation from the exact value at low frequencies is in general minimized by using higher-order spatial approximations. It is, therefore, natural to use fourth-order spatially accurate algorithms not only in order to achieve better accuracy (measured by the $L_2$ error norm) but also to obtain better dispersive properties needed for long time integration in wave propagation and unsteady aerodynamic problems. For these problems, optimal resolution is obtained when high-order of accuracy in time and space is used. However, few implicit algorithms [12] are higher than second order both in time and space.

Central-difference spatial discretization is nondiffusive and constitutes a natural choice for the propagation of acoustic waves which are isotropic, and nondiffusive when governed by the Euler equations. High-order accurate spatial discretization with a small stencil size can be achieved by central, compact schemes where the spatial derivatives are computed in a coupled fashion which implies solution of a tridiagonal or larger bandwidth system of linear equations. Various compact schemes were derived and analyzed by Lele [23]. For example, the first order derivative $f'$ of a function $f$ is evaluated with the fourth-order accurate compact scheme as

$$f'_{j-1} + 4f'_j + f'_{j+1} = 3(f_{j+1} - f_{j-1}). \tag{6}$$

Evaluation of fourth-order accurate first derivatives with Eq. (6) requires scalar tridiagonal matrix inversion, which is obtained efficiently. Compact differentiation involves only a three-point stencil as does the explicit second-order accurate central differencing formula.

Computing a first derivative with sixth order of accuracy requires inversion of a tridiagonal matrix but involves a five-point stencil as

$$f'_{j-1} + 3f'_j + f'_{j+1} = \frac{1}{12}[f_{j+2} + 28(f_{j+1} - f_{j-1}) - f_{j-2}]. \tag{7}$$

Evaluation of the first derivative with an order of accuracy higher than six [23] implies inversion matrices of bandwidth larger than three and requires the use of standard lower-upper decomposition methods. Significant savings in computing time are obtained when the inverses of these matrices are stored. Then the computing cost for the evaluation of the first derivative with compact schemes becomes comparable to the cost for the evaluation of the first derivatives with standard, explicit, high-order central difference forms, which due to their large stencils require more multiplications and additions for the same order of accuracy. The compact schemes in addition to their high-resolution characteristics in wave space [23] provide also an advantage compared to their noncompact counterparts in the implementation, because they require fewer spatial formulations near computational boundaries. Compact schemes can be applied for the entire computational domain using the one-sided formulas for high-order compact schemes which were presented by Carpenter *et al.* [8].

Compact central difference schemes for the evaluation of the space derivatives were used with both explicit Runge–Kutta time integration [23] and implicit factorized, iterative algorithms [40] for time marching. For explicit methods, the spatial order of accuracy used for the evaluation of the fluxes determines the resolution of the scheme [22, 37, 16]. Commonly used explicit Runge–Kutta methods are third- or fourth-order accurate in time. The low-storage, four-stage, fourth-order accurate Runge–Kutta (R–K-4) method is used in many CFD codes [38] in combination with centered discretization in space. The three-stage, two-time level storage, third-order accurate in time Runge–Kutta (R–K-3) method [35], which is TVD in the sense that the temporal operator by itself does not increase the total variation of the solution, was also used with ENO and TVD schemes. Explicit time integration with the optimized Runge–Kutta methods of [22, 9] provide low-dispersion characteristics and when Eqs. (6) or (7) are used to compute the spatial derivatives of the residual term, global high-order spatial accuracy is obtained at least for periodic problems where the overall accuracy level does not degrade by the application of the boundary conditions. Explicit methods, however, due to stability limitations, require large computing times, and become impractical for numerical solution of time-dependent problems in fine stretched meshes. Implicit algorithms presented in the next section can overcome these problems.

### 3.2. *Implicit Algorithms*

Implicit algorithms which provide higher order accuracy in time are the fourth-order method for time integration of wave problems proposed by de Froutos and Sanz-Serna [12] and high-order implicit Runge–Kutta methods [7]. Each physical time step of the de Froutos and Sanz-Serna method [12] is a succession of three implicit midpoint rule time integration steps. For nonlinear equations, not directly solvable for the dependent variables, each intermediate step is carried out with a suitable subiteration process until a certain convergence criterion is fulfilled. Linearization of the convective fluxes, on the other hand, cannot be used because it introduces a second-order error in time. Implicit Runge–Kutta

methods [7] also require large storage, and they are not straightforward to implement for multidimensional, nonlinear problems.

Second-order accurate in time ADI algorithms, on the other hand, are suitable for time-accurate solutions of nonlinear problems, such as the compressible flow equations, and they have been successfully used in LES and DNS [30, 40]. A common ADI method for the Euler equation is the Beam–Warming algorithm [4, 5]. This algorithm is obtained when the nonlinear inviscid fluxes, $\mathbf{F}^{n+1}$, are linearized with a second-order accurate in time Taylor series expansion, $\mathbf{F}^{n+1} = \mathbf{F}^n + (\partial \mathbf{F}/\partial \mathbf{Q})^n (\mathbf{Q}^{n+1} - \mathbf{Q}^n) + \mathcal{O}(\Delta t)^2 = \mathbf{F}^n + \mathbf{A}^n \Delta \mathbf{Q}^n + \mathcal{O}(\Delta t)^2$, where $\mathbf{A}$ is the flux Jacobian matrix for $\mathbf{F}^n$. Using this second-order accurate in time flux linearization and adding the term $(\Delta t)^2 \delta_\xi (\mathbf{A} \Delta Q) \cdot \delta_\eta (\mathbf{B} \Delta Q)$, where $\mathbf{B}$ denotes the flux Jacobian matrix for $\mathbf{G}$, the implicit operator can be approximately factored. The error introduced by the approximate factorization is represented by the added factor term and has a norm $(\Delta t \, \|\mathbf{A}\|) \cdot (\Delta t \, \|\mathbf{B}\|)$, which is proportional to the CFL numbers along the $\xi$ and $\eta$ directions and imposes a limitation on the time step $\Delta t$.

Factorization errors are reduced by the iterative form of the Beam–Warming algorithm [4, 5],

$$\left[\mathbf{I} + \frac{\theta_1 \Delta t}{1+\theta_2} \frac{\partial \mathbf{A}}{\partial \xi}\right]^p \left[\mathbf{I} + \frac{\theta_1 \Delta t}{1+\theta_2} \frac{\partial \mathbf{B}}{\partial \eta}\right]^p \Delta \mathbf{Q}^p = [\mathbf{I} + h \delta_\xi \mathbf{A}_{i,j}]^p [\mathbf{I} + h \delta_\eta \mathbf{B}_{i,j}]^p \Delta \mathbf{Q}^p$$
$$= -\left(\mathbf{Q}^p_{i,j} - \mathbf{Q}^n_{i,j}\right) - \frac{\Delta t}{1+\theta_2} \left(\delta_\xi \mathbf{F}^p_{i,j} + \delta_\eta \mathbf{G}^p_{i,j}\right) + \frac{\theta_2}{1+\theta_2} \Delta \mathbf{Q}^{n-1}_{i,j} = R^p_{i,j}. \tag{8}$$

In this equation, the superscript $^p$ refers to the number of internal subiterations which may be used during time advancement of the solution from time level $n$ to the new time level $n+1$ with physical time step $\Delta t$ in order to eliminate linearization and factorization errors, and errors arising from employing a lower order space discretization of the implicit side. For $p \geq 2$ the residuals typically drop by an order of magnitude for each additional subiteration. In Eq. (8), $h = \sigma \Delta t$, $\sigma = \theta_1/(1+\theta_2)$, $\Delta \mathbf{Q}^p = (\mathbf{Q}^{p+1}_{i,j} - \mathbf{Q}^p_{i,j})$, and $\Delta \mathbf{Q}^{n-1} = \mathbf{Q}^n - \mathbf{Q}^{n-1}$. The first-order accurate in time, Euler implicit scheme is obtained with $\theta_1 = 1, \theta_2 = 0$. Second-order accuracy in time yields the trapezoidal rule with $\theta_1 = 1/2, \theta_2 = 0$, and the three-point backward, implicit time discretization is obtained with $\theta_1 = 1, \theta_2 = 1/2$. Time-accurate computations require second-order accuracy in time and use the trapezoidal rule or the three-point backward scheme. Steady-state calculations can be performed with the first-order accurate in time Euler implicit method or with the diagonal form of the algorithm developed by Pulliam and Chaussee [29].

Pulliam and Chaussee [29] diagonalized the block matrices in Eq. (8) and obtained scalar diagonal operators for the $\xi$ and $\eta$ sweeps. The diagonalized algorithm solves the equations in a decoupled fashion by independent scalar tridiagonal inversions, which were found to be three to four times faster than the block tridiagonal algorithm inversions. Unfortunately, however, the diagonalized algorithm is only first-order accurate in time. Second-order time-accuracy can be obtained if the diagonalized algorithm is combined with a dual time step subiteration scheme [6]. These schemes with internal subiterations were proposed for time-accurate solutions of incompressible flows with the artificial compressibility method by Merkle and Athavale [24] for explicit inner subiterations. Time-accurate solutions of the compressible flow equations were also obtained in Ref. [38] where a "pseudo" time variable was introduced for the internal subiteration which was performed with an explicit scheme and multigrid acceleration techniques.

The left-hand side differentiations of the Beam–Warming algorithm of Eq. (8) or its diagonal form are evaluated using second-order accurate central differences with added second-order implicit smoothing [28], or first-order accurate upwinding of the flux-split [34] Jacobians as in Refs. [13, 30]. For both cases, along each line, only three-point stencils are involved and the block tridiagonal structure of the implicit operator matrices is retained. The tridiagonal structure of the matrices yields a significant advantage in computing speed because inversion is performed efficiently. Recently, time-accurate, high-resolution computations were performed [40] where the right-hand side derivatives were computed using high-order, compact schemes given by Eqs. (6) and (7), while the implicit operators were second-order accurate in space. Internal subiterations within each physical time step were used to drive the residuals to zero. In the next paragraph it will be shown how to obtain fourth-order accuracy for the implicit operators, which helps to reduce the number of internal subiterations and improve the dispersion characteristics of time integration. The block tridiagonal structure of the implicit operators is retained by using compact differentiation formulas for the evaluation of the first-order derivatives.

A single sweep of Eq. (8), along the $\xi$ direction, for example, involves solution of a linear equation system written in operator form as

$$[\mathbf{I} + h\delta_\xi \mathbf{A}_{i,j}](\Delta\mathbf{Q}^*)^p = R^n, \tag{9}$$

where $(\Delta\mathbf{Q}^*)^p = [\mathbf{I} + h\delta_\eta \mathbf{B}_{i,j}]\Delta\mathbf{Q}^p$. This system in expanded form is

$$[(\Delta\mathbf{Q}^*)^p]_{i,j} + h[\mathbf{A}(\Delta\mathbf{Q}^*)^p]'_{i,j} = R^n_{i,j}, \tag{10}$$

where $[\cdot]'$ denotes a first derivative which in the Beam–Warming algorithm is typically computed using second-order accurate central differences as $[\mathbf{A}\cdot(\Delta\mathbf{Q}^*)^p]'_{i,j} = \delta_\xi[\mathbf{A}\cdot(\Delta\mathbf{Q}^*)^p] = 0.5([\mathbf{A}\cdot(\Delta\mathbf{Q}^*)^p]_{i+1,j} - [\mathbf{A}\cdot(\Delta\mathbf{Q}^*)^p]_{i-1,j})$.

In order to achieve high-order, compact differentiation for the implicit operators, we first write two additional relations for the grid points $i-1$ and $i+1$, similar to the one given for the grid point $i$ by (10). Multiplying Eq. (10) by 4, adding the relations for $i+1$ and $i-1$, and using Eq. (6) to evaluate the first derivatives by a fourth-order accurate compact scheme we make the substitution

$$\begin{aligned}&[\mathbf{A}(\Delta\mathbf{Q}^*)^p]'_{i-1,j} + 4[\mathbf{A}(\Delta\mathbf{Q}^*)^p]'_{i,j} + [\mathbf{A}(\Delta\mathbf{Q}^*)^p]'_{i+1,j}\\ &\quad = 3\big([\mathbf{A}(\Delta\mathbf{Q}^*)^p]_{i+1,j} - [\mathbf{A}(\Delta\mathbf{Q}^*)^p]_{i-1,j}\big).\end{aligned} \tag{11}$$

This formula evaluates the derivatives with a fourth-order accurate, three-point, compact stencil. Therefore, when $R^n$ in Eq. (11) is computed with fourth-order spatial accuracy, then an algorithm with fourth-order formal accuracy in space is obtained. This algorithm retains the block tridiagonal matrix structure for the implicit operators because it uses only three-point stencils. Each time step involves two sweeps along the $i$ and $j$ directions as

$$\begin{aligned}&(\mathbf{I} - 3h\mathbf{A}_{i-1,j})(\Delta\mathbf{Q}^*)^p_{i-1,j} + 4(\Delta\mathbf{Q}^*)^p_{i,j} + (\mathbf{I} + 3h\mathbf{A}_{i+1,j})(\Delta\mathbf{Q}^*)^p_{i+1,j}\\ &\quad = R^n_{i-1,j} + 4R^n_{i,j} + R^n_{i+1,j} = \mathbf{R}^n_4\\ &(\mathbf{I} - 3h\mathbf{B}_{i,j-1})(\Delta\mathbf{Q})^p_{i,j-1} + 4(\Delta\mathbf{Q})^p_{i,j} + (\mathbf{I} + 3h\mathbf{B}_{i,j+1})(\Delta\mathbf{Q})^p_{i,j+1}\\ &\quad = (\Delta\mathbf{Q}^*)^p_{i,j-1} + 4(\Delta\mathbf{Q}^*)^p_{i,j} + (\Delta\mathbf{Q}^*)^p_{i,j+1}.\end{aligned} \tag{12}$$

This algorithm is the most second-order accurate in time and in many cases requires additional subiteration indicated by the superscript $^p$ to remove factorization errors.

The fourth-order accurate algorithm requires only a few more multiplications and additions compared to the original ADI algorithm given by Eq. (8) where the left-hand side operators are second-order accurate in space. Therefore, substantial savings in computer time can be obtained because it is not necessary to perform a large number of subiterations for high-order accurate solutions. Numerical experiments have shown that for time-accurate solutions, factorization and linearization errors are removed with only one additional subiteration. Extension of the algorithm in (12) to the compressible Navier–Stokes equations is possible [5]. Viscous terms in the implicit operators in the direction normal to the wall in both algorithms (8) and (12) can be included by using the thin-layer approximation [3] which retains derivatives of the viscous terms only in the normal to wall direction. In the derivation of Eq. (12), the standard, compact, fourth-order accurate central difference formula was used for the differentiations. This formula maximizes the formal order of accuracy but is sub-optimal from the point of view of minimizing dispersion errors. Compact schemes developed in [23, 44] which optimize the dispersion characteristics of the operators can also be used. High-order, implicit, upwind schemes can be obtained by using upwind high-order compact space differentiation [11, 44] with flux-vector splitting [34] instead of first-order upwinding for the ADI algorithms of [30, 13]. A similar, ADI algorithm can be applied to the solution of the linearized Euler equations. This form of the algorithm is given in Appendix A. Compact differentiation which yielded the fourth-order accurate in space algorithm of Eq. (12) can be used in order to obtain order of accuracy higher than four. The computing cost of a sixth-order accurate compact scheme, however, increases significantly because the five-point stencils of Eq. (7) yield block pentadiagonal matrices. Global order of accuracy higher than four can be obtained at a smaller additional computing cost when the diagonal form of the Beam–Warming algorithm is used as it is shown next.

## 4. DIAGONALIZED COMPACT ALGORITHM

The standard diagonalized Beam–Warming algorithm is only first-order accurate in time, and for second-order accuracy in space requires tridiagonal matrix inversions. Using standard fourth-order accurate central differencing formulas in the diagonalized left-hand side operators increases the bandwidth of the scalar matrices to pentadiagonal. Scalar pentadiagonal matrices can be inverted quite efficiently, and the computational cost of the fourth-order accurate diagonalized algorithm is still reasonable. For higher order of accuracy, however, the computing cost increases rapidly. Compact high-order accurate in space implicit operators are constructed as follows.

### *Fourth-Order Diagonalized Algorithm*

Application of fourth-order compact differentiation as in Eq. (11), and substitution of the one-dimensional diagonalized operators yields

$$
\begin{aligned}
&(\mathbf{I} - 3h[\hat{\Lambda}_\xi]_{i-1,j})(\Delta\mathbf{Q}^*)^n_{i-1,j} + 4(\Delta\mathbf{Q}^*)^n_{i,j} + (\mathbf{I} + 3h[\hat{\Lambda}_\xi]_{i+1,j})(\Delta\mathbf{Q}^*)^n_{i+1,j} \\
&\quad = R^n_{i-1,j} + 4R^n_{i,j} + R^n_{i+1,j} = \mathbf{R}^n_4 \\
&(\mathbf{I} - 3h[\hat{\Lambda}_\eta]_{i,j-1})(\Delta\mathbf{Q})^n_{i,j-1} + 4(\Delta\mathbf{Q})^n_{i,j} + (\mathbf{I} + 3h[\hat{\Lambda}_\eta]_{i,j+1})(\Delta\mathbf{Q})^n_{i,j+1} \\
&\quad = (\Delta\mathbf{Q}^*)^n_{i,j-1} + 4(\Delta\mathbf{Q}^*)^n_{i,j} + (\Delta\mathbf{Q}^*)^n_{i,j+1}. \qquad (13)
\end{aligned}
$$

The scalar tridiagonal structure of the implicit operators is retained and fourth-order accuracy in space is obtained. Second-order accuracy in time can be obtained when using a dual time step inner subiteration as in Ref. [6].

*Sixth-Order Diagonalized Algorithm*

Using the sixth-order accurate compact differencing formula (7) to evaluate the derivatives of the left-hand side we obtain the following formula which evaluates the one-dimensional operator derivatives as

$$\begin{aligned}&\mathbf{I}((\Delta\mathbf{Q}^*)^n_{i-1,j}+3(\Delta\mathbf{Q}^*)^n_{i,j}+(\Delta\mathbf{Q}^*)^n_{i+1,j}+h[\Lambda_\xi]'_{i-1,j}(\Delta\mathbf{Q}^*)^n_{i-1,j}\\&\quad+3h[\Lambda_\xi]'_{i,j}(\Delta\mathbf{Q}^*)^n_{i,j}+h[\Lambda_\xi]'_{i+1,j}(\Delta\mathbf{Q}^*)^n_{i+1,j}\\&\quad=R^n_{i-1,j}+3R^n_{i,j}+R^n_{i+1,j}=\mathbf{R}^n_6.\end{aligned}\tag{14}$$

Making the substitution for the derivatives using a sixth-order accurate, five-point stencil $(\Lambda'_{i-1}+3\Lambda'_i+\Lambda'_{i+1}=[\Lambda_{i+2}+28(\Lambda_{i+1}+\Lambda_{i-1})-\Lambda_{i-2})]/12)$ we obtain the following implicit operator for the sweep along $i$ direction,

$$\begin{aligned}&-\left(\frac{h}{12}[\Lambda_\xi]_{i-2,j}\right)(\Delta\mathbf{Q}^*)^n_{i-2,j}+\left(\mathbf{I}-\frac{7h}{3}[\Lambda_\xi]_{i-1,j}\right)(\Delta\mathbf{Q}^*)^n_{i-1,j}+3\mathbf{I}(\Delta\mathbf{Q}^*)^n_{i,j}\\&\quad+\left(\mathbf{I}+\frac{7h}{3}[\Lambda_\xi]_{i+1,j}\right)(\Delta\mathbf{Q}^*)^n_{i+1,j}+\left(\frac{h}{12}[\Lambda_\xi]_{i+2,j}\right)(\Delta\mathbf{Q}^*)^n_{i+2,j}=\mathbf{R}^n_6.\end{aligned}\tag{15}$$

In this equation, the bandwidth of the scalar matrices was increased to pentadiagonal. The term $\mathbf{R}^n_6$ denotes the explicit right-hand-side term which for consistency must be computed using the sixth-order accurate compact finite-differences given by Eq. (7).

The high-order in space implicit time integration schemes discussed in this section are summarized in Tables I and II. The operation count, referring to the total number of additions, multiplications, and divisions to form the implicit operators for a single point, is indicated. The operation count can be considered as a measure of the computing cost of the scheme.The accuracy and efficiency of each method can only be assessed by the actual implementation. Under Illustrative Examples we demonstrate the efficiency and accuracy of the high-order accurate in space compact implicit schemes by comparing the error of the computed solutions with the error obtained with other methods. In the last column of the table the inversion method is indicated. For comparison, the original block tridiagonal algorithm and the diagonalized algorithm are also included. Scalar matrix inversion of (13) is up to 30% faster than the block tridiagonal inversions of (12), but the diagonalized algorithms

**TABLE I**
**Operation Count of Block Tridiagonal Algorithms**

| Implicit algorithm type and/or name | Ref. No. | Eq. No. | Accuracy in time | Accuracy in space | Operation count | Inversion method |
|---|---|---|---|---|---|---|
| Beam–Warming, BW | [4] | (8) | 2nd | 2nd | 746 | Block 3diag. |
| Abarbanel–Kumar, AC | [1] | — | 2nd | 4th | 818 | Block 3diag. |
| Compact, BW-C4 | Present | (12) | 2nd | 4th | 858 | Block 3diag. |

**TABLE II**
**Operation Count of Diagonalized Scalar Tridiagonal Algorithms**

| Implicit algorithm type and/or name | Ref. No. | Eq. No. | Accuracy in time | Accuracy in space | Operation count | Inversion method |
|---|---|---|---|---|---|---|
| Pulliam–Chaussee, PC | [28] | — | 1st | 2nd | 384 | Scalar 3diag. |
| Compact, PC-C4 | Present | (13) | 1st | 4th | 456 | Scalar 3diag. |
| Compact, PC-C6 | Present | (15) | 1st | 6th | 488 | Scalar 5diag. |

are first-order accurate in time and typically run at smaller time steps. The sixth-order diagonalized algorithm requires approximately 25% more computing effort because it involves pentadiagonal matrix inversion.

## 5. UNFACTORED COMPACT ALGORITHM

Beneficial properties of modern upwind schemes, such as diagonal dominance [10] naturally arising at least in first-order spatially accurate upwind schemes, make possible the construction of unfactored algorithms leading to rapid convergence to a time asymptotic steady state [32]. Unfactored implicit algorithms can also be constructed for time-accurate solutions of gas dynamic and aeroacoustic problems. The algorithms were used for efficient time-accurate computations of truly time-dependent solutions [14, 32]. In their nonlinear form, unfactored algorithms correspond to a Newton method which shows quadratic convergence even for very large time steps, but can only be implemented in two-dimensional problems due to memory limitations. Therefore, numerous authors [10, 32, 41] have used relaxation methods for the solution of the unfactored form avoiding factorization errors.

Unfactored implicit algorithms are obtained after linearization of the convective fluxes around an intermediate state $p$ as shown by the following Newton-type iterative method,

$$[\mathbf{I} + h(\delta_\xi \mathbf{A}_{i,j} + \delta_\eta \mathbf{B}_{i,j})]\Delta\mathbf{Q}^p = -[\mathbf{Q}^p - \mathbf{Q}^n + \Delta t R^n], \tag{16}$$

where $\Delta\mathbf{Q}^p = \mathbf{Q}^{p+1} - \mathbf{Q}^p$.

The flux Jacobian matrices in (16) are split as $\mathbf{A} = \mathbf{A}^+ + \mathbf{A}^-$ and the difference operators, $\delta$, are substituted by forward/backward operators for the evaluations of the derivatives associated with $\mathbf{A}^-$, $\mathbf{A}^+$, respectively. A similar process used for the derivation of the factorized algorithms, cf. (12), (13), and (15), is applied, equations analogous to (16) for $i-1$, $i+1$ and $j-1$, $j+1$ are written, and the derivatives are evaluated using the compact, upwind, third-order accurate formulas of Refs. [11, 44]. Then the left-hand-side operators are third-order accurate in space, and time-accurate solutions can be obtained using the following point relaxation scheme,

$$\left[\mathbf{I} + [\mathbf{D}]^p_{i,j}\right]\Delta\mathbf{Q}^p = -\omega[\mathbf{Q}^p - \mathbf{Q}^n + \Delta t R^n] + [\mathbf{D}^\mathbf{C}]^p, \tag{17}$$

where $[\mathbf{D}]$ denotes a matrix with diagonal elements, $[\mathbf{D}^\mathbf{C}]$ is the complementary matrix with the off-diagonal elements resulting from space discretization, and $\omega$ is the relaxation parameter. The plus/minus Jacobian matrices of the flux vectors $\mathbf{A}^\pm$ are obtained either exactly or approximately and the discretized form given by Eq. (17) is solved with a point Gauss–Seidel iteration.

## 6. STABILITY ANALYSIS

We consider the linearized, i.e., constant coefficient, version of the algorithm which is applied, for example, to the numerical solution of the linear wave equation

$$u_t + u_x + u_y = 0. \tag{18}$$

The Von Newmann stability analysis of the fourth-order algorithm given by Eq. (12) is carried out in Fourier space. The Fourier component of the solution vector, $u^n_{l,m}$ at time step $n$ is

$$\begin{aligned} U^n_{l,m} &= \hat{u}^n e^{\mathbf{i}[l(L\Delta x)]} e^{\mathbf{i}[m(M\Delta x)]}, \qquad -\infty \le L \le \infty, \quad -\infty \le M \le \infty \\ &= \hat{u}^n e^{\mathbf{i}[l\phi]} e^{\mathbf{i}[m\theta]}, \qquad -\pi \le \phi \le \pi, \quad -\pi \le \theta \le \pi. \end{aligned} \tag{19}$$

For simplicity assume that standard fourth-order central-differencing is used to evaluate the residual term $(\mathcal{R}_{i,j})_4 = -\Delta t[U_x + U_y]_{i,j}$; then the right-hand side of Eq. (12) becomes $\mathbf{R}^n_4 = (\mathcal{R}_{i-1,j})_4 + 4(\mathcal{R}_{i,j})_4 + (\mathcal{R}_{i+1,j})_4$. The discretized term $(\mathbf{R}^n_4)_{i,j}$ in expanded form is

$$\begin{aligned} \mathbf{R}^n_{i,j} = \frac{\Delta t}{12}[&(U_{i-3,j} - 4U_{i-2,j} - 31U_{i-1,j} + 31U_{i+1,j} + 4U_{i+2,j} - U_{i+3,j}) \\ &+ (U_{i,j-3} - 4U_{i,j-2} - 31U_{i,j-1} + 31U_{i,j+1} + 4U_{i,j+2} - U_{i,j+3})]. \end{aligned} \tag{20}$$

We define the amplification factor matrix as $\hat{Z}$ as $\hat{U}^{n+1} = \hat{Z}\hat{U}^n$. With these definitions and simplifications the constant coefficient version of the algorithm for implicit integration of Eq. (18) is mapped into the Fourier space as

$$\begin{aligned} &\left[(1 - 3h_x)e^{-\mathbf{i}\phi} + 4 + (1 + 3h_x)e^{\mathbf{i}\phi}\right]\left[(1 - 3h_y)e^{-\mathbf{i}\theta} + 4 + (1 + 3h_y)e^{\mathbf{i}\theta}\right](\hat{Z} - 1) \\ &= -\frac{\Delta t}{12}[(e^{-3\mathbf{i}\phi} - 4e^{-2\mathbf{i}\phi} - 31e^{-\mathbf{i}\phi} + 31e^{\mathbf{i}\phi} + e^{2\mathbf{i}\phi} - e^{3\mathbf{i}\phi}) \\ &\quad + (e^{-3\mathbf{i}\theta} - 4e^{-2\mathbf{i}\theta} - 31e^{-\mathbf{i}\theta} + 31e^{\mathbf{i}\theta} + e^{2\mathbf{i}\theta} - e^{3\mathbf{i}\theta})]. \end{aligned} \tag{21}$$

Furthermore, without loss of generality, we can consider the special case where $\Delta x = \Delta y$ or $h_x = h_y = h = \sigma\Delta t$ for which Eq. (21) becomes

$$\begin{aligned} &[2\cos\phi + 4 + \mathbf{i}6h\sin\phi][2\cos\theta + 4 + \mathbf{i}6h\sin\theta](\hat{Z} - 1) \\ &= -\left(\frac{\Delta t}{12}\right)(2\mathbf{i})[-\sin 3\phi + 4\sin 2\phi + 31\sin\phi - \sin 3\theta + 4\sin 2\theta + 31\sin\theta]. \end{aligned} \tag{22}$$

Solving this equation for the amplification factor we obtain

$$\hat{Z} = \frac{L_R + \mathbf{i}(\sigma L_I - \Delta t R_R)}{L_R + \mathbf{i}\sigma L_I}, \tag{23}$$

where $L_R$, $L_I$, and $R_R$ represent the Fourier maps of the left-hand-side and right-hand-side operators, respectively. These terms are given by

$$\begin{aligned} L_R &= 4\sigma[(\cos\phi + 2)(\cos\theta + 2) + (3h)^2 \sin\phi\sin\theta] \\ L_I &= 12[\sin\phi(\cos\theta + 2) + \sin\theta(\cos\phi + 2)] \\ R_R &= [31(\sin\phi + \sin\theta) + 4(\sin 2\phi + \sin 2\theta) - (\sin 3\phi - \sin 3\theta)]/6. \end{aligned} \tag{24}$$

Equation (23) shows that $\hat{Z} \to 1 - \mathbf{i}(R_R/36\sigma^2 \sin\phi\sin\theta\Delta t)$ for very large time step. Therefore, when $\Delta t \to \infty$, the amplification factor magnitude is $\|\hat{Z}\| < 1$, and both the

second- and the first-order accurate in time, fourth-order accurate in space implicit algorithms are unconditionally stable. The same analysis can show that the two-dimensional algorithm is unconditionally stable for all values of the cell aspect ratio.

## 7. IMPLEMENTATION

Numerical solutions of nonlinear problems require damping of high frequency modes which when not resolved by the mesh are aliased and appear as low frequency modes. This is accomplished by either using numerical dissipation or by applying spatial filtering to the solution vector. For compact space differentiation, the compact filtering schemes suggested in Refs. [23, 16] can be used. Implicit seven-point filter formulas were suggested in Ref. [23] where a sixth-order pentadiagonal and a fourth-order tridiagonal spatial filter were implemented and the filter coefficients $a_n$ were derived with Taylor series and Fourier analysis. Tridiagonal eight- and ten-order implicit filters were used in [40]. Filtering is applied sequentially one direction at a time on the conserved variables after each subiteration for implicit time integration or after the final stage of the Runge–Kutta scheme for explicit time integration. At the computational domain boundaries, instead of using one-sided, high-order formulas we can reduce the filter order and still obtain improved, low dissipation spectral characteristics, as suggested in [40].

The fourth-order accurate in space algorithm of (12) can be applied in the domain between $i = 2$ to $i = Imax - 1$ and $j = 2$ to $j = Jmax - 1$, because the derivatives in the right-hand-side term $R_{i,j}$ are evaluated using a three-point compact stencil. For explicit update of the boundaries of the computational domain the residuals are set to zero ($\mathbf{R}_{i,1} = \mathbf{R}_{i,Jmax} = \mathbf{R}_{1,j} = \mathbf{R}_{Imax,j} = 0$) in Eqs. (12), (13), or (15). Application of the sixth-order algorithm of (15) requires, in addition, dropping the accuracy at the point next to the boundary. The residual at the boundary points may be also computed with the one-sided operators suggested by Carpenter *et al.* [8]. Nondissipative algorithms, which are used for direct simulation of turbulent flows and in computational aeroacoustics, have very low dispersion errors and require incorporation of accurate, nonreflective boundary conditions to avoid numerical instabilities and dampen spurious wave reflections at the computational boundaries [27]. Some computations in the next section are performed by assuming periodicity at the computational boundaries. In these cases, the computational domain is folded on itself and no boundary conditions are actually required. In other cases, however, solid wall and radiation boundary conditions are prescribed. These conditions are briefly described where used in the next section.

## 8. ILLUSTRATIVE EXAMPLES

### 8.1. *Linearized Euler Equations*

The performance of the proposed implicit, high-resolution compact schemes is tested first for the linear problem of a pulse reflecting from a solid wall. The governing equations are the linearized Euler equations given by (4), and (A1) in Appendix A. The fourth-order compact scheme given by Eq. (A2) is used for implicit time integration. The computation starts with zero velocity field and pressure field given by

$$p'(x, y) = \exp\left\{-\ln 2\left[\frac{x^2 + (y - y_o)^2}{w_o}\right]\right\}$$

which represents a Gaussian shape pulse of width $w_o$, centered at $x = 0$ and $y = y_o$. This pulse spreads symmetrically in the absence of a freestream and reflects from the solid wall.

On the solid wall surface the nonpenetration condition is imposed on the normal to the wall contravariant velocity component $V = \eta_x u + \eta_y v = 0$ or $\partial V/\partial t = \eta_x u_t + \eta_y v_t = 0$ which yields the following relation for the normal derivative of pressure at the wall:

$$p_\eta = \frac{(\xi_x \eta_x + \xi_y \eta_y)}{\eta_x^2 + \eta_y^2} p_\xi .$$

The wall pressure is obtained by discretizing $p_\eta$ using one-sided, high-order accurate formulas.

The Cartesian velocity components at the wall are obtained as $u = \eta_y U/J$, $v = -\eta_x U/J$ where $J = \eta_y \xi_x - \eta_x \xi_y$. At the far field the radiation boundary condition of Tam and Webb [37] was imposed. Simulations were performed with both uniform grid spacing $\Delta x = 1$ and with numerical meshes stretched in the normal direction in order to provide higher grid resolution in the near wall region. The computed solutions with uniform and stretched meshes were practically identical. The computed results can be compared with the exact solution given in [19]. The acoustic pressure field, for example, is given by the closed-form solution,

$$p(x, y, t) = \frac{1}{2\alpha} \int_0^\infty e^{-\frac{\xi^2}{4\alpha}} \cos(\xi t)[J_o(\xi x_1) + J_o(\xi x_2)]\xi \, d\xi ,$$

where $\alpha = \ln 2/w_o$, $x_1 = [x^2 + (y - y_o)^2]^{1/2}$, $x_2 = [x^2 + (y + y_o)^2]^{1/2}$, and $J_o$ is the zeroth-order Bessel function of the first kind.

First, a solution is computed for a pulse located far from the solid wall at $y_o = 100$. The computation is carried out until $T = 50$, so that the the spreading pulse wavefront is still far from the wall and the reflected acoustic field is very weak. For this model problem, the effects of the boundaries are insignificant and the errors in the computed solution are caused mainly from the interior numerical scheme and not from boundary treatment. Computed results are compared with the exact solution in Fig. 1. The comparison of Fig. 1a shows that at $T = 25$ the solution computed with the second-order accurate in space implicit scheme of Eq. (8), without subiterations, with fourth-order accurate spatial derivative evaluation of the right-hand-side terms using Eq. (6), deviates from the exact solution. Results computed with the same grid spacing and second-order accurate in space right-hand side (not shown here) are in poor agreement with the exact solution. The error of computed solutions is plotted in Fig. 1b. Table I shows that the solution obtained with the fourth-order accurate in space implicit scheme of Eq. (12) requires 15% more computing time, compared to the solution obtained with second order accuracy by using the algorithm of Eq. (8). Figure 1b, however, shows that for this increase of computing time we obtain a reduction of the error by an order of magnitude.

Next, a computation was carried out using Eq. (12), up to $T = 100$ nondimensional time units for a pulse located at $y_o = 25$. Figure 2 shows comparisons of the computed results with the exact solution along the line normal to $x = 0$ for different times. Figures 2a and 2b show comparisons before the pressure front reaches the wall. Figures 2c and 2d show comparisons during reflection, and Figs. 2e and 2f show the incident and reflected pulses. Note that in Fig. 2f the scale for the acoustic pressure has been changed. Each pulse is resolved with 15 points. All comparisons show very good agreement of the computation

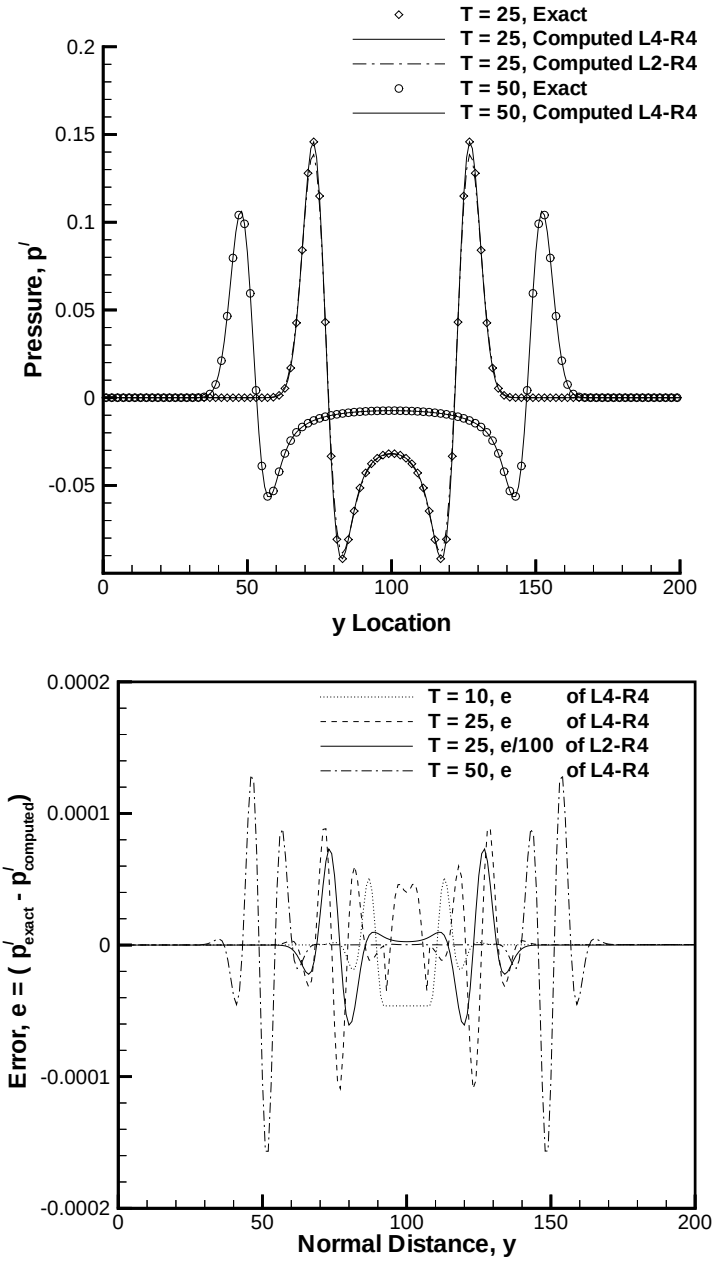


**FIG. 1.** Comparison of computed results with the exact solution and error for the computed pressure at T = 10, T = 25, and T = 50; spreading of the pulse, $p'(x, y) = \exp\{-\ln 2[(x^2 + (y - 100)^2)/25]\}$.

with the exact solution. No phase errors are observed for long integration times performed with large time steps.

## 8.2. *Full Euler Equations*

*8.2.1. Convection of a density disturbance.* Solutions of the full nonlinear Euler equations are presented next. First, numerical solutions for simple convection of a

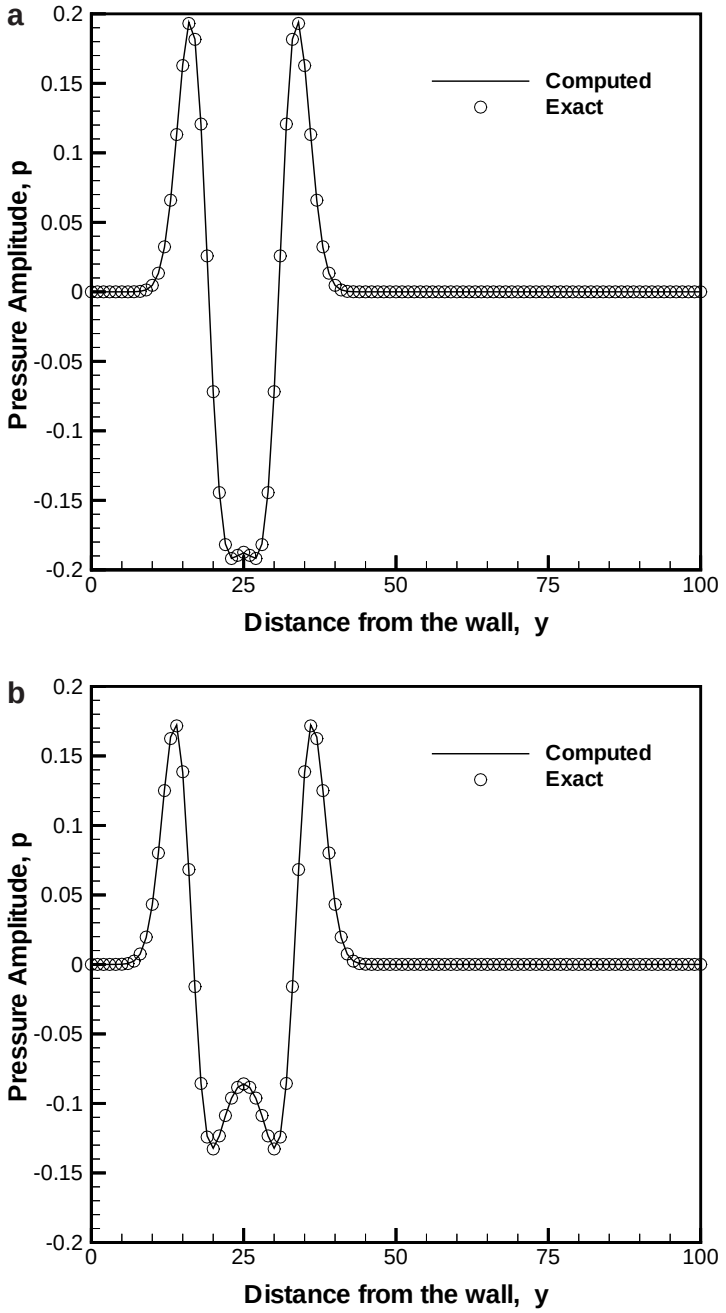


**FIG. 2.** Comparison of the computed acoustic pressure along the normal at $x = 0$ with the exact solution at times (a) T = 7.5, (b) T = 10, (c) T = 20, (d) T = 25, (e) T = 50, and (f) T = 100; reflection of the pulse, $p'(x, y) = \exp\{-\ln 2[(x^2 + (y - 25)^2)/25]\}$, from the solid wall at $y = 0$.

Gaussian-type density disturbance of peak amplitude $\rho' = 0.1\rho_\infty$ by a uniform subsonic free stream $M = 0.2$ are obtained using implicit and explicit time integration methods. This density disturbance in the absence of diffusion must convect without change in shape and amplitude. The same time step was used for time integration with all implicit schemes and

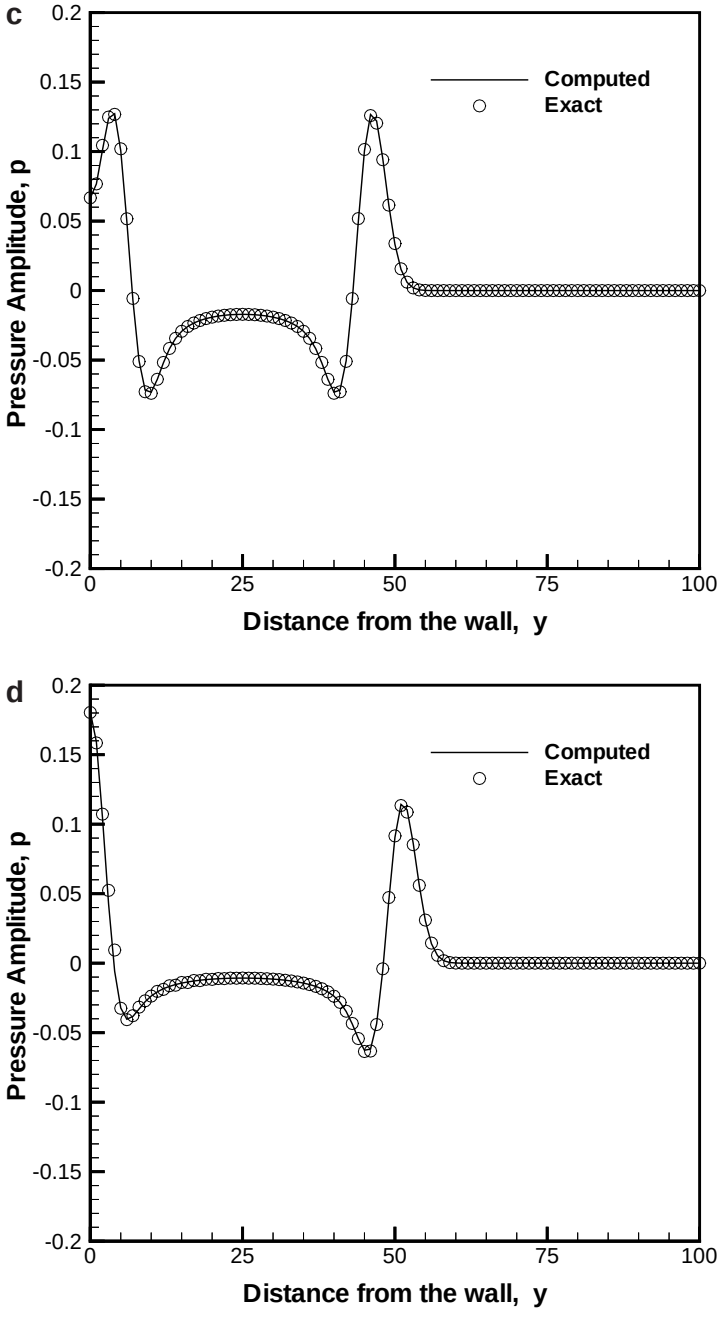


**FIG. 2**—*Continued*

the pulse was convected 50 nondimensional lengths. The explicit solution was computed with one-tenth of the time step. The numerical solutions were obtained without adding any form of numerical dissipation or explicit filtering and without subiterations. The same grid spacing $\Delta x = 1$ was used. The solution obtained with the present fourth-order implicit schemes of Eq. (12), the fourth-order compact scheme of Abarbanel and Kumar [1], and the fourth-order Runge–Kutta (R–K-4) method where the space discretization is performed with

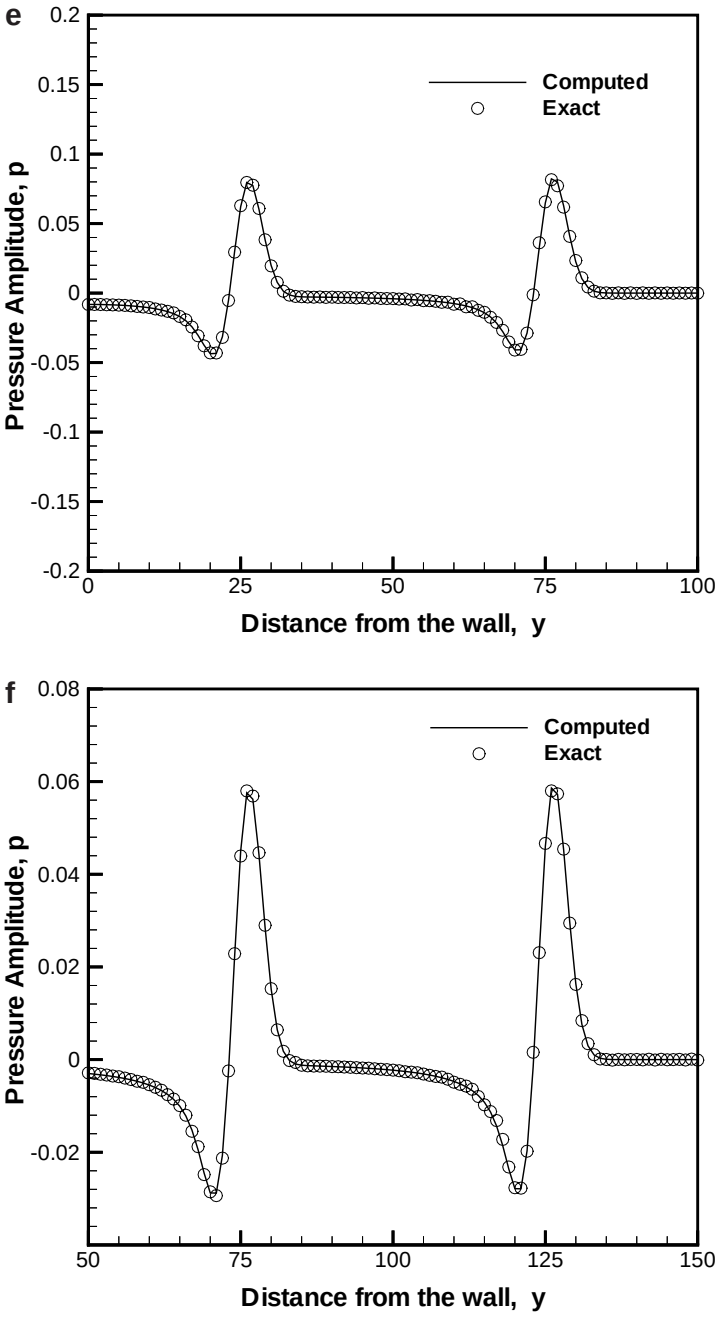


**FIG. 2**—*Continued*

a fourth-order accurate formula of (6), shown in Fig. 3, are practically identical. The pulse shape obtained from time integration with the fourth-order accurate in space diagonalized, compact scheme of Eq. (13) shows phase errors because it is more dispersive since it is only first-order accurate in time. The solution obtained with the standard, second-order accurate in space implicit Beam–Warming algorithm of Eq. (8), where the right-hand-side term was computed with fourth-order accurate finite differences, shows differences from

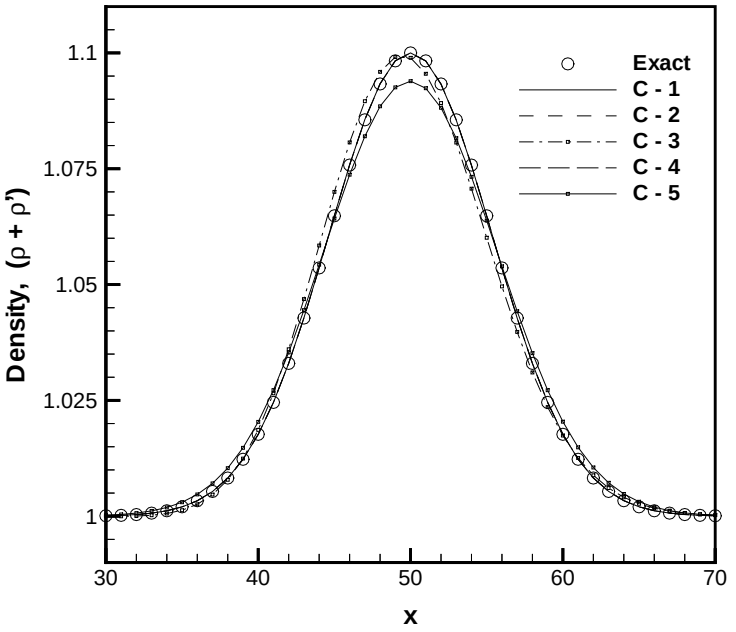


**FIG. 3.** Comparison of the computed density amplitude with the exact solution for the convection of a density disturbance: (C-1) 4th order, block tridiagonal, compact implicit; (C-2) 4th order, Abarbanel–Kumar; (C-3) 4th order, diagonal, compact, implicit; (C-4) explicit R–K-4, 4th order accurate, compact right-hand side (RHS); (C-5) Beam–Warming 2nd order in space implicit, 4th order accurate RHS.

the exact solution and development of larger dispersion errors. The maximum error ≈7% is obtained for the second order accurate in space and time implicit algorithm (C-5). The fourth order accurate diagonalized algorithm (C-3) shows a maximum error of ≈2%, and the other schemes have very little error.

*8.2.2. Vorticity convection.* Convection and preservation of vorticity is important for DNS/LES simulations and the key element for many CFD applications. Vortex-generated sound is also of importance to many aeroacoustic applications. The ability of numerical methods to obtain acoustic fields generated by vortices was the subject of recent numerical investigations [25]. In helicopter rotor aerodynamics currently available direct wake capturing methods, which are usually second-order accurate in space, demand a very large number of grid points in order to prevent numerical diffusion of the wake vorticity. Therefore the ability of the proposed high-order implicit algorithms to accurately convect vorticity with less diffusion is demonstrated next. A model problem of vortex convection is considered where initial vortical flow conditions are imposed at $t = 0$ by prescribing velocities and pressure using the following relations as in Refs. [27, 40],

$$u = U_\infty - \frac{C(y - y_c)}{R_c^2} e^{-r^2/2}$$

$$v = V_\infty - \frac{C(x - x_c)}{R_c^2} e^{-r^2/2}$$

$$p = p_\infty - \frac{C^2 \rho}{2R_c^2} e^{-r^2/2}$$

for an inviscid vortex of strength $C$ and vortex core radius $R_c$ centered at $x_c$, $y_c$. The relation for pressure was obtained by integrating the radial momentum equation $\partial p/\partial r = \rho u_\theta^2/r$ about the vortex center, where $r$ is the radial distance given by $r = \sqrt{[(x - x_c)^2 + (y - y_c)^2]/R_c^2}$. Accuracy tests for the convection of this inviscid vortex by a uniform stream $M = 0.2$ are performed and solutions of the Euler equations are obtained using the explicit third-order Runge–Kutta method, the second-order accurate in space implicit method [4] of Eq. (8), and the new compact schemes. Implicit integration was performed without subiterations. Numerical solutions are obtained with a uniform grid spacing $\Delta x = \Delta y$ for a vortex of strength $C = 0.05$ and $R_c = 10$. The computations were initialized with a density obtained by using an isentropic flow assumption. First, the ability of the explicit method in predicting the correct vortex strength after the vortex has been convected $x/R_c = 50$ nondimensional length units is demonstrated. The three-stage Runge–Kutta method (R–K-3) is used for time advancement, and space discretization is performed with standard fourth-order accurate central-difference formulas and added sixth-order artificial dissipation. Solutions obtained with grid spacing of $\Delta x = \Delta y = 0.5$, 1.0, and 2.0 are compared with the exact solution in Fig. 4.

The solution obtained with a grid density of $\Delta x = 2.0$ shows phase errors. The results obtained with the highest grid density yielded a quality required in LES simulations while the solution obtained with $\Delta x = 1$ provides an accuracy level sufficient for most CFD solutions. The quality of numerical solutions, which use the Beam–Warming algorithm for time integration and different levels of grid refinement, and their ability to predict the correct vortex strength, is shown in Fig. 5. For reference, the centerline vorticity obtained from the numerical solution with the explicit (R–K-3) method and grid spacing $\Delta x = 2.0$

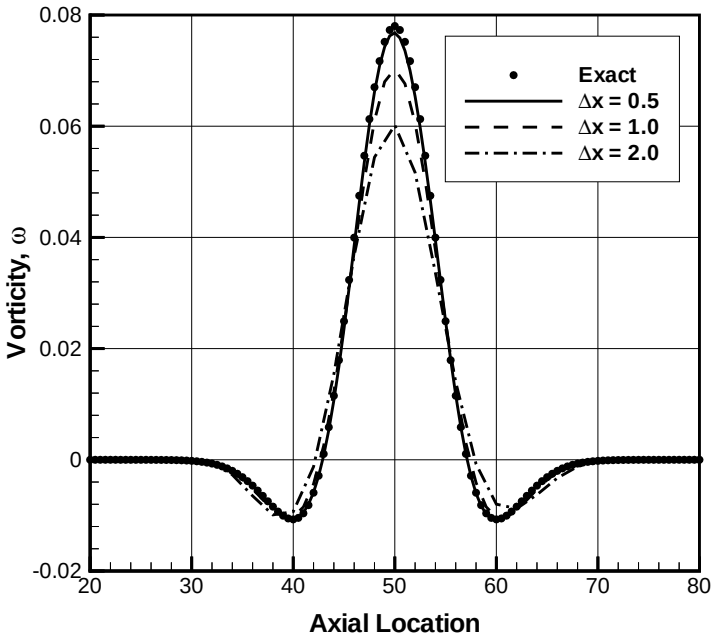


**FIG. 4.** Advection of vorticity with different grid densities. Explicit time integration with the third-order accurate in time Runge–Kutta (R–K-3) method. Comparison of the computed vorticity along the centerline with the exact solution.

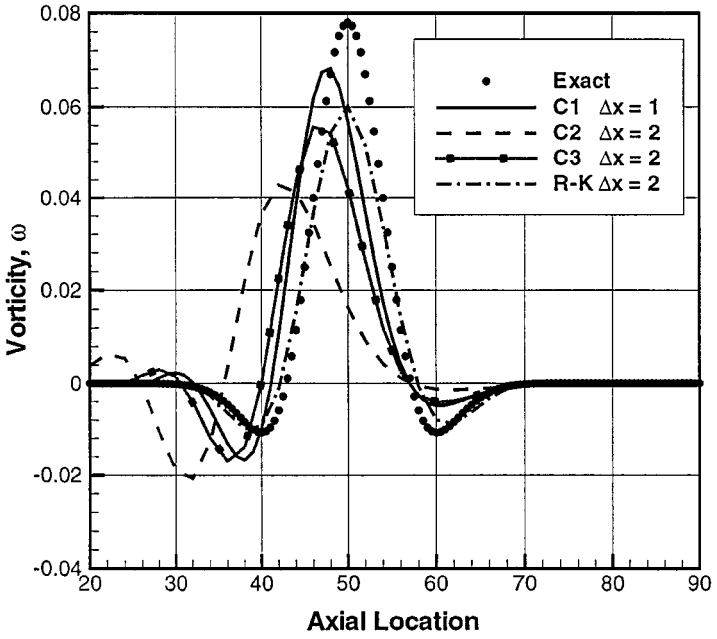


**FIG. 5.** Advection of vorticity with different grid densities. Implicit integration with the second-order accurate in space and time Beam–Warming algorithm; (C1) 4th-order accurate right-hand side (RHS); (C2) 2nd-order accurate RHS; (C3) 4th-order accurate RHS. Comparison of the computed vorticity along the centerline with the exact solution.

is shown in the same figure. Increasing the grid density (curve C1) and higher order of accuracy in space yields a better solution quality. The numerical solution obtained with the second-order of accuracy in space for both the right-hand side and the implicit integration has very large dispersion errors. The solution obtained with fourth-order accurate central differences for the convective fluxes has larger phase errors than the explicit time solution which used compact space differentiation.

The solution obtained with the fourth-order accurate implicit compact scheme of Eq. (12) is compared in Fig. 6 with the numerical solutions obtained using the second-order accurate in space implicit Beam–Warming algorithm where the derivatives on the right-hand side are computed using fourth-order accuracy in space. The solution computed with the present, fourth-order accurate in space block tridiagonal scheme, as shown in Table I, has a slightly higher computing cost. It shows, however, good agreement with the exact solution and the phase errors are small even though half the number of points along each direction was used. The maximum error in amplitude of the solution computed with the algorithm of Eq. (12) and $\Delta x = 1$ is approximately 4% while the solutions computed with the second-order accurate in space implicit scheme have large dispersion and maximum amplitude errors 15% and 6% for $\Delta x = 1$ and $\Delta x = 0.5$, respectively.

Finally, results computed with the present scheme are compared in Fig. 7 with the results obtained from a solution using the Abarbanel and Kumar [1] fourth-order compact scheme. Both schemes yield almost identical levels of accuracy for grid spacing $\Delta x = \Delta y = 2$. For reference a solution computed with a Runge–Kutta method and fourth-order accurate explicit central difference evaluation of the convective derivatives is shown in the same figure.

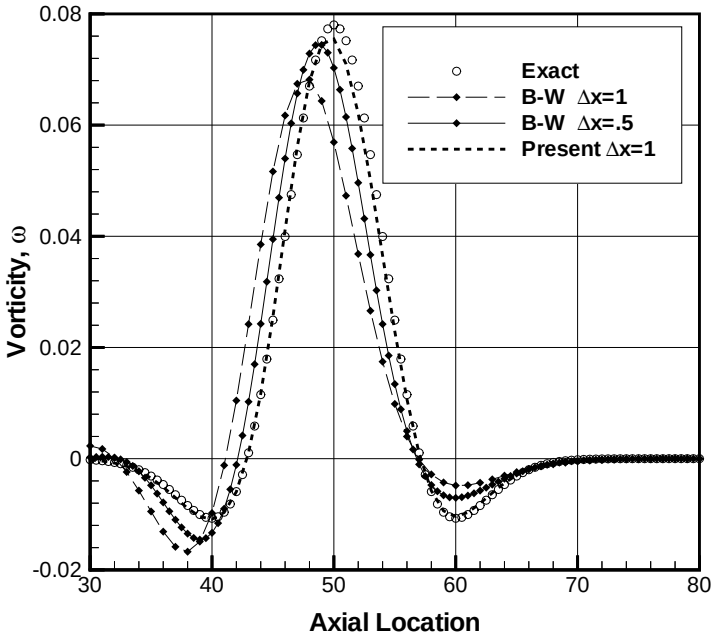


**FIG. 6.** Comparison of the vorticity along the centerline computed with the Beam–Warming algorithm and the present compact scheme.

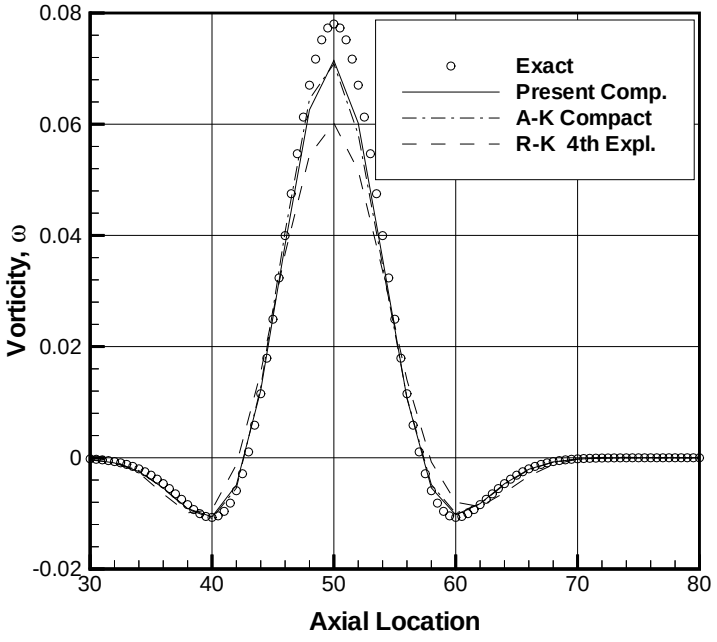


**FIG. 7.** Comparison of the vorticity along the centerline computed with $\Delta x = 2$ and using the Abarbanel and Kumar compact scheme and the present compact scheme.

The implicit solutions were obtained with a time step 20 times larger than the explicit solution. As a result the total computation cost of the implicit solution was lower. The numerical solutions for the convection of the vortex required filtering. Solutions obtained with second-order accurate in space implicit methods used the standard fourth-order difference operators [29] for explicit smoothing and second-order difference operators for implicit smoothing. The solutions obtained with compact schemes and the explicit (R–K-3) method used compact implicit filtering described in Section 7. The maximum error of the solution computed with the fourth-order accurate compact scheme is ≈5% for $\Delta x = 1$ and ≈12% for $\Delta x = 2$. For the same grid spacing $\Delta x = 2$ the same level of accuracy was achieved with the compact scheme of Abarbanel and Kumar [1].

*8.2.3. Airfoil flows.* The fourth-order block tridiagonal algorithm of Eq. (12) and the diagonalized scheme of Eq. (13) are used for numerical solutions with stretched meshes required in computations of flows of practical interest. In these computations, the convergence characteristics of the proposed schemes are demonstrated. Solutions are computed for viscous and inviscid subsonic flow at Mach number $M_\infty = 0.3$ over a NACA-0012 airfoil at an angle of incidence $\alpha = 10$ deg. Inviscid solutions were computed with a C-type $201 \times 41$ point grid. High-order accurate compact finite differences of (6) and (7) were used to evaluate the right-hand-side terms. The computed surface pressure distribution (not shown here) for all cases was in agreement with the measurements. The convergence rates were obtained by the second-order accurate in space algorithms, and the high-order compact schemes of Eqs. (12), (13), and (15) are shown in Figs. 8 and 9. The convergence histories of the second- and fourth-order accurate block-tridiagonal algorithms shown in Fig. 8 were obtained for a

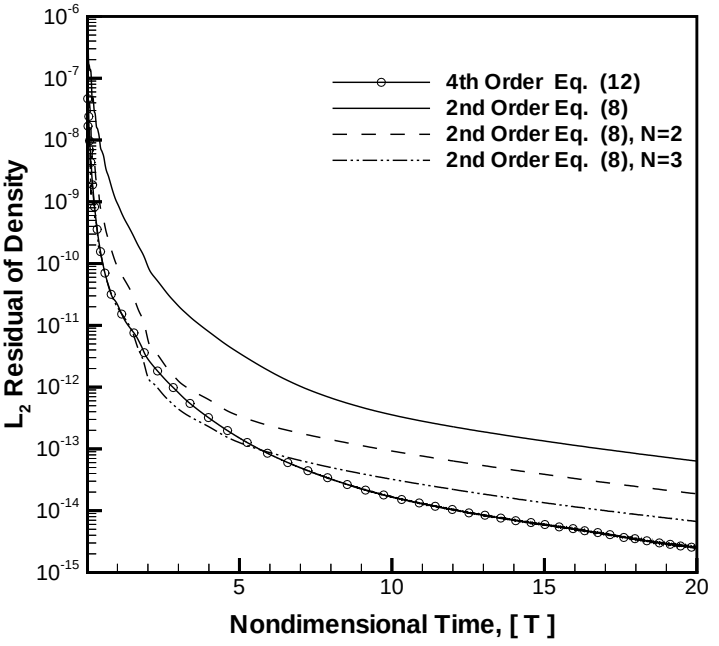


**FIG. 8.** Convergence of the block tridiagonal fourth-order accurate in space algorithm and the original second-order accurate algorithm with subiterations ($N = 2$ and $N = 3$), shown by the $L_2$ norm of density residuals for inviscid flow over an airfoil; $M = 0.3$ and $\alpha = 10^o$.

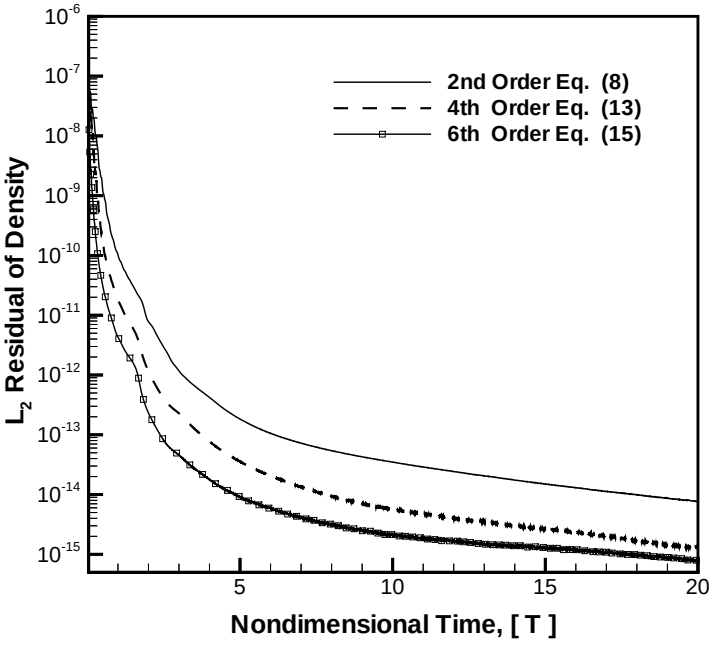


**FIG. 9.** Convergence of the second- and fourth-order accurate in space diagonalized algorithms shown by the $L_2$ norm of density residuals for inviscid flow over an airfoil; $M = 0.3$ and $\alpha = 10^o$.

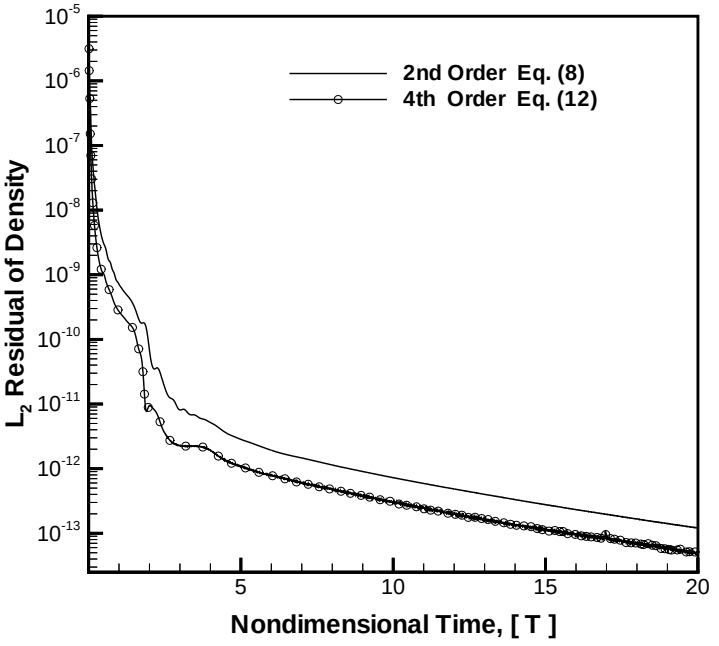


**FIG. 10.** Convergence of the second- and fourth-order accurate in space block tridiagonal algorithms, shown by the $L_2$ norm of density residuals for viscous flow over an airfoil; $M = 0.3$, $\alpha = 10^o$, and $Re_c = 5 \times 10^6$.

nondimensional time step $\Delta t = 0.01$. For comparison, steady-state computations obtained by the second-order algorithm using additional subiterations are shown in the same figure. Computations of the same flow with the first-order accurate in time diagonalized algorithms were carried out with size of the time step $\Delta t = 0.005$. Convergence rates of the fourth- and sixth-order accurate algorithms are compared with the convergence of the standard, second-order accurate in space algorithm in Fig. 9. For all cases, the high-order compact schemes yielded a better convergence rate than their second-order accurate counterparts. For viscous flow solutions, only the block tridiagonal second- and fourth-order accurate algorithms are used because the viscous Jacobians do not simultaneously diagonalize with the inviscid flux Jacobians. The viscous solution was obtained on a $281 \times 81$ point grid at $\alpha = 10$ deg, for a Reynolds number base on the chord $Re_c = 5 \times 10^6$, and the turbulent flow was simulated with an algebraic eddy viscosity model [3]. Figure 10 shows that a better convergence is obtained for the fourth-order accurate computations for viscous flow.

## 9. CONCLUDING REMARKS

Compact, high-order accurate in space, implicit integration schemes were developed. These schemes use the same methodology implemented for the development of second-order accurate in space ADI algorithms and unfactored relaxation methods, which are currently used in many CFD codes for steady-state and time-accurate computations. The time accuracy of the new algorithms is still at most second-order because of the linearization in time which is applied to the nonlinear fluxes. High-order accuracy is obtained by evaluating the spatial derivatives of the implicit operators by compact differencing schemes at no increase of the computing cost. Global high resolution is obtained when the right-hand side explicit part of the numerical scheme is computed with a high-order accurate method. The new algorithms have similar stability characteristics and faster convergence rates compared to their second-order accurate in space counterparts. The proposed schemes were tested for linear aeroacoustic problems and the computed results were in agreement with the exact solution. Numerical solutions of the two-dimensional Euler equations were also obtained, and the results of vortex convection by uniform stream have shown that significant savings in computing time and improved accuracy can be obtained from the application of the proposed implicit integration schemes.

## APPENDIX A

The linearized, compressible Euler equations in generalized curvilinear coordinates are

$$\frac{\partial \hat{\mathbf{q}}'}{\partial t} + \frac{\partial \hat{\mathbf{f}}'}{\partial \xi} + \frac{\partial \hat{\mathbf{g}}'}{\partial \eta} = 0, \tag{A1}$$

where the primitive variable vector $\hat{\mathbf{q}}'$ becomes $\hat{\mathbf{q}}' = \frac{1}{J}[\rho' \; u' \; v' \; p']^T$, and the transformed flux vectors $\hat{\mathbf{f}}'$ and $\hat{\mathbf{g}}'$ are $\hat{\mathbf{f}}' = \frac{1}{J}[\xi_x \mathbf{f}' + \xi_y \mathbf{g}']$, $\hat{\mathbf{g}}' = \frac{1}{J}[\eta_x \mathbf{f}' + \eta_y \mathbf{g}']$, where the Cartesian flux vectors $\mathbf{f}'$ and $\mathbf{g}'$ are given in (4). The flux Jacobian matrix $\mathbf{A}' = \partial \hat{\mathbf{f}}' / \partial \hat{\mathbf{q}}'$ of the linearized Euler equations is independent of $\hat{\mathbf{q}}'$ and depends on the spatial coordinates through the transformation metrics. Similarly to Eq. (12) the second-order accurate in time fourth-order

accurate in space implicit scheme for the linearized Euler equations is

$$
\begin{aligned}
&(\mathbf{I} - 3hA'_{i-1,j})(\Delta\hat{q}'^{*})^{p}_{i-1,j} + 4(\Delta\hat{q}'^{*})^{p}_{i,j} + (\mathbf{I} - 3hA'_{i+1,j})(\Delta\hat{q}'^{*})^{p}_{i+1,j} \\
&\quad = R^{n}_{i-1,j} + 4R^{n}_{i,j} + R^{n}_{i+1,j} \\
&(\mathbf{I} - 3hB'_{i,j-1})(\Delta\hat{q}')^{p}_{i,j-1} + 4(\Delta\hat{q}')^{p}_{i,j} + (\mathbf{I} - 3hB'_{i,j+1})(\Delta\hat{q}')^{p}_{i,j+1} \\
&\quad = (\Delta\hat{q}'^{*})^{p}_{i,j-1} + 4(\Delta\hat{q}'^{*})^{p}_{i,j} + (\Delta\hat{q}'^{*})^{p}_{i,j+1}. \qquad \text{(A2)}
\end{aligned}
$$

Similar to Eq. (12) the right-hand side of (A2) also implies use of points $i-2$ and $i+2$ for the terms $R^{n}_{i-1,j}$ and $R^{n}_{i+1,j}$; therefore, the algorithm applies from $i = 3$ to $i = Imax - 2$. The differentiations of $A$ and $B$ in (A2) are included to account for the metric variation.

The first-order accurate in time diagonalized algorithm is not appropriate for aeroacoustics solutions. Use of a dual time stepping [6], which yields second-order accuracy in time can make the diagonalized algorithm suitable for time-accurate numerical simulations in aeroacoustics. Furthermore, implementation of upwind compact differences for the evaluation of the time derivative can yield third-order accurate in time dual time step methods.

## ACKNOWLEDGMENTS

The author gratefully acknowledges the support of this work by the Army Research Office under Phase II SBIR Contract DAAG55-97-C-0029. Special thanks to Helen Yee, Marcel Vinokur, and Lawrence Keefe for their critical review of the manuscript.

## REFERENCES

1. S. Abarbanel and A. Kumar, Compact high-order schemes for the Euler equations, *J. Sci. Comput.* **3**(3), 275 (1988).
2. N. A. Adams and K. Shariff, A high-resolution hybrid compact-ENO scheme for shock turbulence interaction problems, *J. Comput. Phys.* **127**, 27 (1996).
3. B. S. Baldwin and H. Lomax, *Thin-Layer Approximation and Algebraic Model for Separated Turbulent Flow*, AIAA Paper 78-0257, 1978 (unpublished).
4. R. M. Beam and R. F. Warming, An implicit finite-difference algorithm for hyperbolic systems in conservation-law form, *J. Comput. Phys.* **22**, 87 (1976).
5. R. M. Beam and R. F. Warming, An implicit factored scheme for the compressible Navier–Stokes equations, *AIAA J.* **16**(4), 393 (1978).
6. E. O. Buelow, D. A. Schwer, J. Feng, C. L. Merkle, and D. Choi, *A Preconditioned Dual-Time, Diagonalized ADI Scheme for Unsteady Computations*, AIAA Paper 97-2101, 1997 (unpublished).
7. J. C. Butcher, *The Numerical Analysis of Differential Equations* (Wiley, New York, 1987).
8. M. H. Carpenter, D. Gottlieb, and S. Abarbanel, The stability of numerical boundary treatments of compact high-order finite-difference schemes, *J. Comput. Phys.* **108**, 272 (1993).
9. M. H. Carpenter and C. A. Kennedy, *Fourth-Order 2N-Storage Runge–Kutta Schemes*, NASA-TM-109112, 1994 (unpublished).
10. S. Chakravarthy, *Relaxation Methods for Unfactored Implicit Upwind Schemes*, AIAA Paper 84-0165, 1984 (unpublished).
11. B. Cockburn and C. Shu, Nonlinear stable compact schemes for shock calculations, *SIAM J. Numer. Anal.* **31**(3), 607 (1994).
12. J. de Froutos and J. M. Sanz-Serna, An easily implemented fourth-order method for the time integration of wave problems, *J. Comput. Phys.* **103**, 160 (1992).
13. J. A. Ekaterinaris and F. R. Menter, Computation of oscillating airfoil flows with one- and two-equation turbulence models, *AIAA J.* **32**(12), 2359 (1994).

14. J. A. Ekaterinaris, Numerical simulation of incompressible two-blade rotor flowfields, *J. Propulsion Power* **14**(3), 367 (1998).
15. D. Fu and Y. Ma, A high order accurate difference scheme for complex flow fields, *J. Comput. Phys.* **134**, 1 (1997).
16. D. Gaitonde and J. S. Shang, Optimized compact-difference-based finite-volume schemes for linear wave phenomena, *J. Comput. Phys.* **138**, 617 (1997).
17. D. Gottlieb and E. Turkel, Dissipative two-four method for time dependent problems, *Math. Comput.* **30**(136), 703 (1976).
18. B. Gustafsson and P. Olsson, Fourth-order difference methods for hyperbolic IBVPs, *J. Comput. Phys.* **117**, 300 (1995).
19. J. C. Hardin, J. R. Ristorcelli, and C. K. Tam, *ICASE/LaRC Workshop on Benchmark Problems in Computational Aeroacoustics (CAA)*, NASA CP 3300, 1995 (unpublished).
20. Z. Haras and S. Ta'Asan, Finite difference schemes for long-time integration, *J. Comput. Phys.* **114**, 265 (1994).
21. R. Hixon, S.-H. Shih, and R. R. Mankbadi, Evaluation of boundary conditions for computational aeroacoustics, *AIAA J.* **33**(11), 2006 (1995).
22. F. Q. Hu, M. Y. Hussaini, and J. L. Mantley, Low-dissipation and low-dispersion Runge–Kutta schemes for computational aeroacoustics, *J. Comput. Phys.* **124**, 177 (1996).
23. S. K. Lele, Compact finite difference schemes with spectral-like resolution, *J. Comput. Phys.* **103**, 16 (1992).
24. C. L. Merkle and M. Athavale, *Time-Accurate Unsteady Incompressible Flow Algorithm Based on Artificial Compressibility*, AIAA Paper 87-1137 (unpublished).
25. B. E. Mitchell, S. K. Lele, and P. Moin, Direct computation of sound from a compressible co-rotating vortex pair, *J. Fluid Mech.* **285**, 181 (1995).
26. S. Osher and S. Chakravarthy, Upwind schemes and boundary conditions with applications to Euler equations in general geometries, *J. Comput. Phys.* **50**, 447 (1983).
27. T. J. Poinsot and S. K. Lele, Boundary conditions for direct simulations of compressible viscous flows, *J. Comput. Phys.* **101**, 104 (1992).
28. T. H. Pulliam, Artificial dissipation models for the Euler equations, *AIAA J.* **24**(12), 1931 (1986).
29. T. H. Pulliam and D. S. Chaussee, A diagonal form of an implicit approximate-factorization algorithm, *J. Comput. Phys.* **39**, 347 (1981).
30. M. M. Rai and P. Moin, Direct numerical simulation of transition and turbulence in a spatially evolving boundary layer, *J. Comput. Phys.* **109**, 169 (1993).
31. P. L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, *J. Comput. Phys.* **43**, 357 (1981).
32. S. E. Rogers, Comparison of implicit schemes for the incompressible Navier–Stokes equations, *AIAA J.* **33**(11), 2066 (1995).
33. J. S. Shang, Characteristic-based algorithms for solving the Maxwell equations in the time domain, *IEEE Antennas Propagation* **37**(3), 15 (1995).
34. J. L. Steger and R. F. Warming, Flux vector splitting of the inviscid gas dynamic equations with applications to finite-difference methods, *J. Comput. Phys.* **40**, 263 (1981).
35. C. Shu and S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, *J. Comput. Phys.* **77**, 439 (1988).
36. C. K. W. Tam, Computational aeroacoustics: Issues and methods, *AIAA J.* **33**(10), 1788 (1995).
37. C. K. W. Tam and J. C. Webb, Dispersion-relation-preserving finite difference schemes for computational aeroacoustics, *J. Comput. Phys.* **107**, 262 (1993).
38. V. Venkatakrishnan and D. J. Mavriplis, Implicit method for the computation of unsteady flows on unstructured grids, *J. Comput. Phys.* **127**, 380 (1996).
39. R. Vichnevetsky and J. B. Bowels, *Fourier Analysis of Numerical Approximations of Hyperbolic Equations* (Soc. for Industr. & Appl. Math., Philadelphia, 1982).

40. M. R. Visbal and D. V. Gaitonde, High-order-accurate methods for complex unsteady subsonic flows, *AIAA J.* **37**(10), 1231 (1999).
41. S. Yoon and D. Kwak, Implicit Navier–Stokes solver for three-dimensional compressible flows, *AIAA J.* **30**(11), 2653 (1992).
42. H. Yee, Explicit and implicit multidimensional compact high-resolution shock-capturing methods: Formulation, *J. Comput. Phys.* **131**, 216 (1997).
43. D. W. Zingg, H. Lomax, and H. Jurgens, *An Optimized Finite-Difference Scheme for Wave Propagation Problems*, AIAA Paper 93-0459, 1993 (unpublished).
44. X. Zhong, High-order finite-difference schemes for numerical simulation of hypersonic boundary-layer transition, *J. Comput. Phys.* **144**, 662 (1998).