TIME ACCURACY AND THE USE OF IMPLICIT METHODS

Thomas H. Pulliam, Associate Fellow AIAA

NASA Ames Research Center

Implicit time differencing methods are usually developed to be time accurate. The typical implementation and use, though, is either non-time accurate for steady state convergence acceleration or the schemes are modified (e.g. approximate factorization, explicit boundary conditions, linearization error, etc.) to enhance their efficiency, practical use, or tractability on conventional computers. This papers examines these departures from time accuracy and introduces methods to enhance the time accuracy of conventional schemes. Approximations are examined for their effect on time accuracy, subiteration techniques designed to improve time accuracy are presented, and analysis for time accuracy assessment are introduced.

I.   Introduction

Since 1976, when Steger[1] first introduced a practical implicit finite difference scheme for the Euler and Navier-Stokes equations, there have been numerous (too numerous to reference here) modifications and new methods developed which use implicit time approximations with various spatial discretion techniques. Until recently, most of the effort was directed toward steady state or slowly varying unsteady applications. Most of these use either large time steps, spatially variable scalings, or preconditioning techniques to accelerated convergence. The next generation of challenges in CFD, though, will be in the area of unsteady time accurate calculations. At face value one might assume that explicit techniques are the choice for such computations. The demands of adequate boundary layer and complicated geometric resolution make explicit methods too restrictive in terms of time steps and efficiency for most practical cases. The alternative is to develop more efficient and accurate implicit methods. One advantage of implicit methods over explicit is that larger time steps can be used than would be permitted by explicit stability bounds. For instance, a high Reynolds number viscous transonic airfoil computation using fine grid resolution at the surface typically requires CFL numbers in the wall normal direction on the order 10^3. These arguments have been delineated many times in many references and applications with the result that implicit methods of one form or another are widely used today.

This paper will address some of the approximations used to make implicit methods more efficient and practical for the solution of the Euler and Navier-Stokes equations. In particular, approximate factorizations, diagonalizations and linearization approximations will be reviewed and categorized. A subiteration correction scheme commonly used today will be presented, improved, demonstrated and analyzed. This scheme is used to produce a second order accurate, more robust implicit method for unsteady flow computations. Also presented here are 3rd and 4th order time accurate schemes of the ESDIRK class.[4,5]

II.   Euler Equations

The generic form of a system of partial differential equations will be used for demonstration and analysis,

$$\partial_t Q + \mathcal{F}(Q) = 0 \tag{1}$$

This generic system of equations may represent any number of conventional problem definitions, for example: the two-dimensional Euler equations on a Cartesian grid,

$$\partial_t Q + \partial_x E + \partial_y F = 0 \tag{2}$$

so that, $\mathcal{F}(Q) = \partial_x E + \partial_y F$ **with**

$$
Q = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ e \end{bmatrix}, \quad
E = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(e + p) \end{bmatrix}, \quad
F = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(e + p) \end{bmatrix}
\tag{3}
$$

with $p = (\gamma - 1)(e - 0.5\rho(u^2 + v^2))$ **and** $\gamma = 1.4$ **for a perfect gas,**

A.   Test Case

An isolated vortex propagation problem is used as the test case. A two-dimensional (2D) uniform grid is assumed, with periodic boundary conditions in x and simple Dirichlet conditions in y enforced on a rectangular domain. The domain size for the results shown below is 10×10 units where the vortex core size (diameter) C_d is defined as 1.0 unit. The equations for the initial condition are a finite core vortex embedded in a free stream flow $Q_\infty = (\rho_\infty, M_\infty, 0, e_\infty)$ with $\rho_\infty = 1.0, M_\infty = 0.5, p_\infty = \frac{1}{\gamma}$, and $T_\infty = 1.0$. The perturbed (by the vortex) field is

$$
\begin{aligned}
T &= T_\infty - \frac{V_s^2(\gamma - 1)}{16 G_s \gamma \pi^2} e^{2 G_s (1 - r^2)} \\
\rho &= T^{\frac{1}{(\gamma - 1)}} \\
u &= M_\infty - \frac{V_s}{2\pi}(y - y_0) e^{G_s (1 - r^2)} \\
v &= \frac{V_s}{2\pi}(x - x_0) e^{G_s (1 - r^2)}
\end{aligned}
\tag{4}
$$

with the vortex strength $V_s = 5.0$ and the Gaussian width scale $G_s = 0.5$. The vortex is initially centered at $x_0 = 5.0$ and $y_0 = 5.0$ and $r = \sqrt{(x - x_0)^2 + (y - y_0)^2}$. Figure 1 shows density contours of the initial vortex, which has a period of 10 C_d and unless stated all computed results are after 30 C_d or three revolutions of the vortex across the domain.
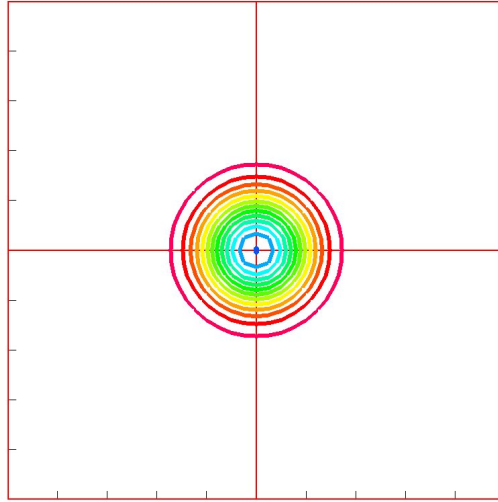

Figure 1.   Density Contours, Initial Vortex

III.   Time Advance Solver: Implicit Forms

The time advance approach in OVERFLOW is to use implicit time integration for either steady state computations or in a time accurate mode. Details can be found in numerous

publications, e.g., the **OVERFLOW** manual[2] or in Pulliam.[3] The options for time accuracy are either 1st or 2nd order using simple time stepping (block tri-diagonal operators) or dual-timestepping. We shall provide the development of the dual-time stepping in two forms given below and also show the preliminary development of 3rd and 4th order accurate implicit Runge-Kutta schemes, **ESDIRK.**[4,5]

An implicit approximation in time for the solution of **Eq. 2** can be written as

$$\Delta Q^n = \frac{\theta\Delta t}{1+\phi}\frac{\partial}{\partial t}(\Delta Q^n) + \frac{\Delta t}{1+\phi}\frac{\partial}{\partial t}Q^n + \frac{\phi}{1+\phi}\Delta Q^{n-1} + O\left[(\theta - \frac{1}{2} - \phi)\Delta t^2\right] + O\left[\Delta t^3\right]$$

with $\Delta Q^n = Q^{n+1} - Q^n$ and $Q^n = Q(n\Delta t)$. The parameters θ and ϕ can be chosen to produce different schemes of either first or second order accuracy in time.

The values $\theta = 1$ and $\phi = 0$, results in the first order Euler implicit scheme, $\theta = 1/2$ and $\phi = 0$ for a trapezoidal implicit or for $\theta = 1$ and $\phi = 1/2$ gives the three point backward second order implicit scheme (typically referred to as **BDF2**).

Using **Eq. 1** to replace $\frac{\partial}{\partial t}Q$ with $-\mathcal{F}(Q)$ we have (neglecting the higher order terms)

$$\Delta Q^n + \frac{\theta\Delta t}{1+\phi}\mathcal{F}(Q^{n+1}) = \frac{(\theta-1)\Delta t}{1+\phi}\mathcal{F}(Q^n) + \frac{\phi}{1+\phi}\Delta Q^{n-1} + \mathcal{O}(\Delta t^3) \tag{5}$$

Equation 5 is nonlinear in terms of Q^{n+1} due to the second term on the left hand side. The nonlinear terms are linearized in time about Q^n by a Taylor series such that

$$\mathcal{F}(Q^{n+1}) = \mathcal{F}(Q^n) + \mathcal{A}(Q^n)\Delta Q^n + \mathcal{O}(\Delta t^2) \tag{6}$$

where $\mathcal{A} = \partial\mathcal{F}(Q)/\partial Q$ is typically called the Jacobian of $\mathcal{F}$ and ΔQ^n is $\mathcal{O}(\Delta t)$. Note that the linearizations are carried out to the $\mathcal{O}(\Delta t^2)$ terms. These terms will be multiplied (see below) by Δt and so if a second order time scheme had been chosen the linearizations would not degrade the time accuracy.

Replacing $\mathcal{F}(Q^{n+1})$ using **Eq. 6** in **Eq. 5** one gets

$$\left[I + \frac{\theta\Delta t}{1+\phi}\mathcal{A}(Q^n)\right]\Delta Q^n = -\frac{\Delta t}{1+\phi}\mathcal{F}(Q^n) + \frac{\phi}{1+\phi}\Delta Q^{n-1} \tag{7}$$

which is second order accurate in time for $\phi = 0, \theta = \frac{1}{2}$ or $\phi = \frac{1}{2}, \theta = 1$. **Equation 7** is kept in what is called "Delta Form," referring to the $\Delta Q^n = Q^{n+1} - Q^n$ term on the left hand side.

Equation 7 is the basis for most implicit time integration schemes. If the original problem of interest was a system of nonlinear partial differential equations, e.g. the Euler equations, then the implicit operator of **Eq. 7** would represent a matrix operator of the order of the system size. In addition, if finite difference operators were used for the spatial derivatives then the implicit operator would represent an even larger matrix system of the order of the system size times the discrete grid dimensions.

A. Implicit Approximations

In general, **Eq. 7** is not too difficult, but prohibitively time consuming, to solve directly. Large sparse matrix systems usually result from conventional finite difference or finite volume schemes for the spatial derivatives. Approximations to the implicit operator in **Eq. 7** are employed to either improve the efficiency, reduce computer storage requirements, enhance the stability, or map the system onto parallel processors.

A more general form is **Eq. 7** is

$$\mathcal{L}(Q^n)\Delta Q^n = \mathcal{R}(Q^n, Q^{n-1}) \tag{8}$$

Approximations to $\mathcal{L}(Q^n)$ are made for a variety of reasons. In steady state computations (where one is just interested in satisfying $\Delta Q^n = 0$), any approximation to $\mathcal{L}(Q^n)$ which produces a stable and convergent solution process is admissible. In fact, $\mathcal{L}(Q^n)$ is typically taken as some $\mathcal{O}(\Delta t)$ approximation. In unsteady computations, one tries to maintain the order of accuracy within any approximations to $\mathcal{L}(Q^n)$, while at the same time making the time integration as efficient as possible. Examples include, Beam-Warming[6] approximate factorization of multidimensional implicit schemes applied to the Euler and Navier-Stokes equations and a diagonal[7] (1st order in time) variant of Beam-Warming.

Beam and Warming[6] developed an approximate factorization of multidimensional implicit schemes applied to the Euler and Navier-Stokes equations. The implicit scheme applied to Eq. 2 yields

$$[I + \alpha \delta_x A^n + \alpha \delta_y B^n] \Delta Q^n = \mathbf{R}(Q^n, Q^{n-1}) = -\beta (\delta_x E(Q^n) + \delta_y F(Q^n)) + \frac{\phi}{1+\phi} \Delta Q^{n-1}$$

with $A^n = \frac{\partial E}{\partial Q}$, $B^n = \frac{\partial F}{\partial Q}$, $\alpha = \frac{\theta \Delta t}{1+\phi}$ and $\beta = \frac{\Delta t}{1+\phi}$.

Second order central differences are typically used for the δ derivatives in Eq. 9, which would lead to a very large sparse matrix representing $\mathcal{L}(Q^n)$.

Factorization of Eq. 9 produces

$$[I + \alpha \delta_x A^n] [I + \alpha \delta_y B^n] \Delta Q^n - \alpha^2 (\delta_x A^n)(\partial_y B^n) \Delta Q^n = \mathbf{R}(Q^n, Q^{n-1}) \tag{9}$$

The α^2 term in Eq. 9 can be neglected since with $\Delta Q^n = \mathcal{O}(\Delta t)$ that term is now $\mathcal{O}(\Delta t^3)$ which maintains the second order accuracy of the implicit scheme. In Eq. 9, each of the separate implicit operators are now banded block tridiagonal matrices and are much easier to invert.

Pulliam and Chaussee[7] introduce a further approximation where they diagonalized the implicit operators in Eq. 9 using the eigensystems of $A^n = \mathbf{X}(Q)\Lambda_A \mathbf{X}^{-1}(Q)$ and $B^n = \mathbf{Y}(Q)\Lambda_B \mathbf{Y}^{-1}(Q)$.

Briefly, the eigenvector matrices for A^n and B^n are factored out of the implicit operators producing

$$\mathbf{X}(Q) [I + \alpha \delta_x \Lambda_A] \mathbf{X}^{-1}(Q) \mathbf{Y}(Q) [I + \alpha \delta_y \Lambda_B] \mathbf{Y}^{-1}(Q) \Delta Q^n + \mathcal{O}(\Delta t^2) = \mathbf{R}(Q^n, Q^{n-1})$$

The advantage of this scheme is reduced operation counts for the inversions. Instead of banded block matrices, one now deals with matrix multiples and banded scalar inversions, since the bracketed terms are diagonal. One disadvantage is that the maximum time accuracy is now first order (the error term in Eq. 10 is $\mathcal{O}(\Delta t^2)$). Another disadvantage is that the unsteady nature of this scheme is nonconservative due to the solution dependence of eigenvector matrices and the errors induced by the diagonalization. On the other hand, the steady state solutions for this scheme would be identical to the nondiagonalized scheme (the approximation only affect the implicit operators which at convergence goes to zero leaving $\mathbf{R}(Q^n, Q^{n-1}) = 0$ for both schemes), see Pulliam.[7]

A number of other common approximations have been used with varying degrees of consequences in terms of accuracy, efficiency, and stability. Time and space would only allow one to address a small subset of the numerous approximations presented in the literature. One common approximation is to use low order accurate spatial difference operators on the implicit side of Eq. 8, $\mathcal{L}(Q^n)$, and high order operators on the explicit side, $\mathbf{R}(Q^n, Q^{n-1})$. This is especially advantageous when high order upwind differences are used in conjunction with flux splitting or flux difference methods, see Steger and Warming[8] or Rai.[9] Other approximations include, modified flux Jacobians, incomplete decompositions of the implicit side matrices, approximate artificial dissipation operators and many more. In general, we consider our system of equations to be of the form

$$\mathcal{L}(Q^n)\Delta Q^n + \mathcal{O}(\Delta t^{r-1})\Delta Q^n = \mathcal{R}(Q^n, Q^{n-1}) \tag{10}$$

where the second term on the left hand side is $\mathcal{O}(\Delta t^r)$ since ΔQ^n is $\mathcal{O}(\Delta t)$. Of concern is the effect of the error term on stability, iterative convergence to a steady state, and in particular for this paper, the resulting time accuracy when approximations are applied. In general, any approximation to the left hand side of Eq. 8 will affect the time accuracy in some manner.

IV. Time Accurate Methods

Various approaches to time accurate methods exist in the literature ranging from explicit linear multi-step schemes (e.g. Lax-Wendroff), multi-stage schemes (e.g. Runge-Kutta), implicit forms (e.g. full implicit, approximate factorization, implicit Runge-Kutta, etc). Sometimes the choice is more a matter of personal preference, but most likely efficiency and accuracy considerations are more often the criteria.

Within the realm of implicit schemes, direct methods for the unfactored implicit operators Eq. 8, (irregardless of the discrete approximation in space, i.e. finite-difference, finite volume, etc), are too expensive especially in three-dimensions (3D). Approximate factorization[1,3,6] is one approach, but time accuracy can be compromised by efficiency approximations and the cross term errors of the approximate factorization. In addition, other aspects of real geometry CFD codes, for example, multi-zone approaches,[11] etc. can degrade time accuracy.

A. Dual Time Step

One way to recover the advantage of the fully implicit form, Eq. 5, is the recast the scheme in terms of a dual-time-stepping approach. Starting with Eq. 1, add the pseudo-time τ derivative of the solution variable Q to get

$$\partial_\tau Q + [\partial_t Q + \mathcal{F}(Q)] = 0 \tag{11}$$

and drive the pseudo time integration of the fixed point ($\partial_\tau Q = 0$), i.e. steady-state in τ. At full convergence in τ, the original nonlinear system is satisfied. This is actually done within the discrete time space, where we used implicit discrete time approximations in both t and τ. The fully implicit approximation, Eq. 5, is recast as

$$\frac{\left((1+\phi)Q^{n+1} - (1+2\phi)Q^n + \phi Q^{n-1}\right)}{\Delta t} + \theta \mathcal{F}(Q^{n+1}) + (1-\theta)\mathcal{F}(Q^n) + \mathcal{O}(\Delta t^2) = 0 \tag{12}$$

In discrete pseudo-time τ (using iteration index s) with 1st order Euler implicit and replacing $n+1$ with $s+1$ in **Eq. 12**, we have

$$\frac{Q^{s+1} - Q^s}{\Delta \tau} + \frac{\left((1+\phi)Q^{s+1} - (1+2\phi)Q^n + \phi Q^{n-1}\right)}{\Delta t} + \theta \mathcal{F}(Q^{s+1}) + (1-\theta)\mathcal{F}(Q^n) + \mathcal{O}(\Delta t^2) + \mathcal{O}(\Delta \tau) = 0 \tag{13}$$

Linearizing $\mathcal{F}(Q^{s+1})$ as in **Eq. 6**, collecting $\Delta Q^s = Q^{s+1} - Q^s$ on the left, we have

$$\left[\left(\frac{1}{\Delta \tau} + \frac{1+\phi}{\Delta t}\right)I + \theta \mathcal{A}(Q^s)\right](Q^{s+1} - Q^s) + \mathcal{O}(\Delta \tau) =$$

$$-\left[\frac{\left((1+\phi)Q^s - (1+2\phi)Q^n + \phi Q^{n-1}\right)}{\Delta t} + \theta \mathcal{F}(Q^s) + (1-\theta)\mathcal{F}(Q^n) + \mathcal{O}(\Delta t^2)\right] \tag{14}$$

This is the dual-time-stepping scheme, which can be found in numerous reference, e.g., Rai,[9] Venkateswaren, et al.,[12].[13] The basic approach starting at physical time step n (setting $Q^{s=0} = Q^n$) to $n+1$, is to integrate the left hand side of Eq. 14 with your favorite implicit scheme, e.g. approximation factorization, diagonal scheme, direct method, etc., until convergence in s. The n and $n-1$ terms are evaluated from previous time levels and after iterating s times, the solution at time level $n+1$ will be taken from the most recent Q^{s+1}. Choosing $\phi = \frac{1}{2}$ and $\theta = 1$ and in the limit (assuming the iterative process converges) $Q^{s+1} = Q^s$, setting $Q^{n+1} = Q^{s+1}$ we have

$$\frac{3Q^{n+1} - 4Q^n + Q^{n-1}}{2\Delta t} + \mathcal{F}(Q^{n+1}) + \mathcal{O}(\Delta t^2) = 0 \tag{15}$$

which is a second order in time fully implicit approximation to Eq. 1.

Equation 14 can be written in a form similar to Eq. 8

$$\mathcal{L}(Q^s)(Q^{s+1} - Q^s) + \mathcal{O}(\Delta t^2) + \mathcal{O}(\Delta \tau) = \mathcal{R}(Q^s, Q^n, Q^{n-1}) \tag{16}$$

We can now require that the first term on the left hand side be approximated with an error term of order $\mathcal{O}(\Delta t^r)$ with $r \geq 2$.

Analysis of Eq. 16 will determine conditions on the approximate implicit operator $\mathcal{L}(Q^s)$. The first criteria on $\mathcal{L}(Q^s)$ is that the resulting iterative scheme does converge. The additional requirement is that the local subiteration process be performed until the error is second order in time. The convergence of the iterative process and the accuracy requirement are directly linked, if the iterative scheme fails to converge the error term will be large.

The requirement for accuracy is easily defined in terms of Eq. 16 as having

$$\mathcal{L}(Q^s)(Q^{s+1} - Q^s) \approx \mathcal{O}(\Delta t^3) \tag{17}$$

fall within the truncation error of a second order accurate scheme. This term is exactly the residual of the integration operator which most numerical codes use as a measure of convergence. Error estimates should be obtainable from measures of the subiteration residual and/or its rate of convergence.

Results for the vortex propagation using an **80 grid point mesh, the 2nd order implicit dual time stepping schemes and the 5th order accurate central difference scheme (made up of the 6th order flux differences and 5th order artificial dissipation) are shown in Fig. 2 after $100C_r$ for increasing time steps $\Delta t = 0.025, 0.05, 0.1, 0.2$**
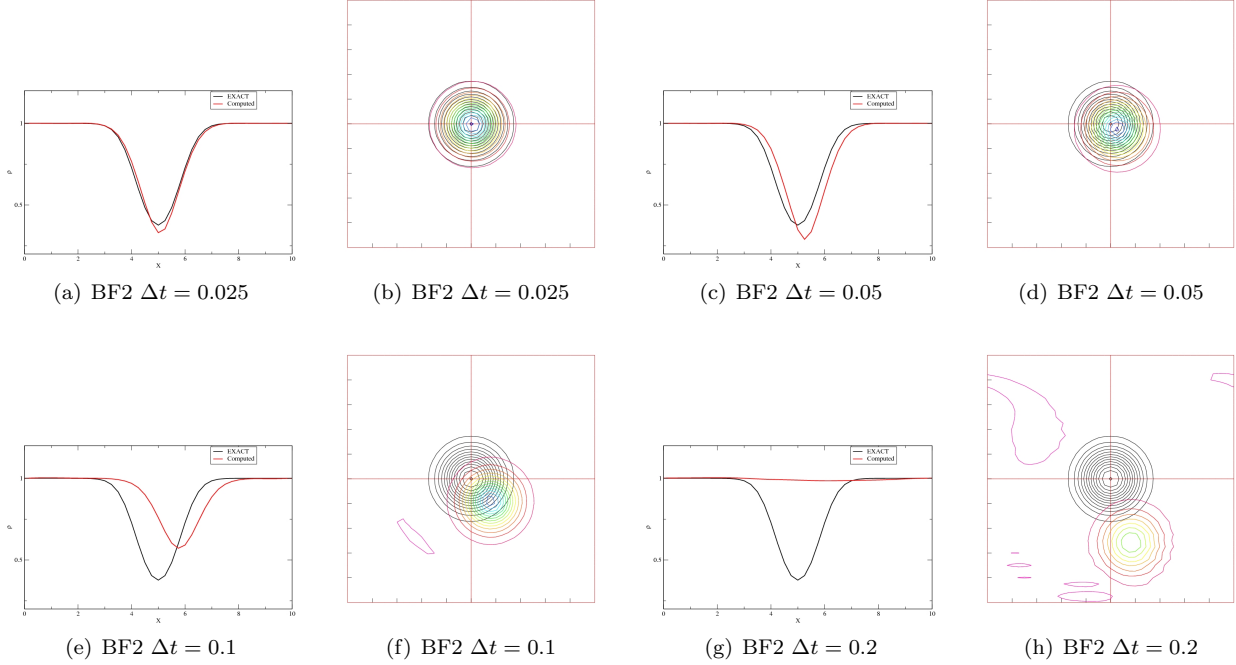

(a) BF2 $\Delta t = 0.025$ (b) BF2 $\Delta t = 0.025$ (c) BF2 $\Delta t = 0.05$ (d) BF2 $\Delta t = 0.05$

(e) BF2 $\Delta t = 0.1$ (f) BF2 $\Delta t = 0.1$ (g) BF2 $\Delta t = 0.2$ (h) BF2 $\Delta t = 0.2$

Figure 2.   Centerline Density and Contours, Computed Vortex: 100 C_r.

B.   Newton-Scheme for Time Accuracy

Following Eq. 8, a second order in time difference approximation to Eq. 1 can be written as

$$\Delta t \left[\frac{\partial \mathbf{Q}}{\partial t} + \mathcal{F}(\mathbf{Q}) \right] \approx \Delta Q^n + \frac{\theta \Delta t}{1 + \phi} \mathcal{F}(Q^{n+1}) = \frac{(\theta - 1)\Delta t}{1 + \phi} \mathcal{F}(Q^n) + \frac{\phi}{1 + \phi} \Delta Q^{n-1} + \mathcal{O}(\Delta t^3) \tag{18}$$

for proper choices of ϕ and θ.

Neglecting for now the error term, rewrite Eq. 18 substituting a new iterative index $p + 1$ for the $n + 1$ terms and add Q^p to both sides of the equality, giving

$$\Delta Q^p + \frac{\theta \Delta t}{1 + \phi} \mathcal{F}(Q^{p+1}) = -(Q^p - Q^n) + \frac{(\theta - 1)\Delta t}{1 + \phi} \mathcal{F}(Q^n) + \frac{\phi}{1 + \phi} \Delta Q^{n-1} \tag{19}$$

Linearizing about Q^p and writing in "Delta Form" we have

$$\left[I + \frac{\theta \Delta t}{1 + \phi} \mathcal{F}'(Q^p) \right] (Q^{p+1} - Q^p) = \frac{(\theta - 1)\Delta t}{1 + \phi} \mathcal{F}(Q^n) - \frac{\Delta t}{1 + \phi} \mathcal{F}(Q^p) - \left[Q^p - \frac{1 + 2\phi}{1 + \phi} Q^n + \frac{\phi}{1 + \phi} Q^{n-1} \right] \tag{20}$$

with $\mathcal{F}'(Q^p) = \partial \mathcal{F}/\partial \mathbf{Q}$.

Equation 20 is the basic newton-iteration time advance scheme which will yield second order time accuracy independent of the choice of $\mathcal{F}'(Q^p)$ if the subiteration process converges. The n and $n - 1$ terms are evaluated from previous time levels and after iterating

p times the solution at time level $n+1$ will be taken from the most recent Q^{p+1}. Choosing $\phi = \frac{1}{2}$ and $\theta = 1$, in the limit (assuming the iterative process converges) $Q^{p+1} = Q^p$, setting $Q^{n+1} = Q^{p+1}$ we have

$$\frac{3Q^{n+1} - 4Q^n + Q^{n-1}}{2\Delta t} + \mathcal{F}(Q^{n+1}) = 0 \tag{21}$$

which is a second order in time fully implicit approximation to Eq. 1.

Equation 20 can be written in a form similar to Eq. 8

$$\mathcal{L}(Q^p)\Delta Q^p = \mathcal{R}(Q^p, Q^n, Q^{n-1}) \tag{22}$$

where $\mathcal{R}(Q^p, Q^n, Q^{n-1})$ represents the fully implicit approximation to Eq. 1. We can now consider the left-hand-side as the error term of order $\mathcal{O}(\Delta t^r)$ with $r \geq 2$.

Analysis of Eq. 22 will determine conditions on the approximate implicit operator $\mathcal{L}(Q)$. The first criteria on $\mathcal{L}(Q)$ is that the resulting iterative scheme does converge. The additional requirement is that the local subiteration process be performed until the error is second order in time. The convergence of the iterative process and the accuracy requirement are directly linked, if the iterative scheme fails to converge the error term will be large.

The solution procedure when employing Eq. 22 is as follows. Given a choice of the approximate operator $\mathcal{L}(Q)$ and an initial solution Q^n (probably from one iteration of a first order scheme) a given number of subiterations $p = 0, 1, \ldots, np$ are performed (typically 3, but some analysis below will provide guidelines for a proper choice). Comparing Eq. 20 and Eq. 7, we see a change of variable (n to p) and the addition of a few terms to the right-hand-side. If one assumes that a given numerical code represents the basic operators $\mathcal{L}$ and $\mathcal{R}$, then the addition of the subiteration process is a trivial source term included in the numerical code. The amount of extra computational work is directly proportional to the number of subiterations required. Note that all operations of the original numerical code, including boundary operators are applied to Q^p. After the subiteration process is adequately converged the solution Q^{n+1} is updated.

C.   ESDIRK Schemes

A class of Implicit Runge-Kutta schemes[4, 5, 14] can be employed which are higher order accurate and have L-stability performance. Starting with a Eq. 1 again, $\partial_t Q + \mathcal{F}(Q, t) = 0$ [a], we define a class of ESDIRK(m) schemes as

$$Q^k = Q^n - \Delta t \sum_{j=1}^{k} a_{k,j} \mathcal{F}(Q^j, t^n + c_j \Delta t) \quad k = 1, \ldots, S \tag{23a}$$

$$Q^{n+1} = Q^n - \Delta t \sum_{j=1}^{S} b_j \mathcal{F}(Q^j, t^n + c_j \Delta t) \tag{23b}$$

$$\widehat{Q}^{n+1} = Q^n - \Delta t \sum_{j=1}^{S} \widehat{b}_j \mathcal{F}(Q^j, t^n + c_j \Delta t) \tag{23c}$$

where S is the number of stages and the coefficients are presented in the Butcher[15] tables given below. The coefficients, $a_{k,j}$ are the stage weights, b_j and $\widehat{b}_j$ are the main and embedded scheme weights, the solutions Q^{n+1} and $\widehat{Q}^{n+1}$, are p^{th} and $(p-1)^{th}$ - order accurate, and the c_i are the weights for the explicit time evaluations (i.e., $t + c_j \Delta t$) for each stage. The stiffly accurate assumption,[16] $a_{k,j} = b_j$, makes the solution, Q^{n+1} independent of any explicit process within the integration step.

The general form of a Butcher Table (Table 1) is given below and represents a wide variety of explicit/implicit multi-stage schemes of various order.

The specific form of a Butcher Table (Table 2) for the class of ESDIRK(m) schemes is given below. Note, the first stage is trivial ($Q^1 = Q^n$) and that the diagonals are constant and also that the last stage is the same as the final residual accumulation and therefore is typically redundant.

---

[a] An explicit dependency on time t has been added so that we can be specific as to time at which various stages are evaluated.

c_1	a_{11}	0	0	0	0	0
c_2	a_{21}	a_{22}	0	0	0	0
c_3	a_{31}	a_{32}	a_{33}	0	0	0
c_4	a_{41}	a_{42}	a_{43}	a_{44}	0	0
c_5	a_{51}	a_{52}	a_{53}	a_{54}	a_{55}	0
1	a_{61}	a_{62}	a_{63}	a_{64}	a_{65}	a_{66}
	b_1	b_2	b_3	b_4	b_5	b_6
	$\widehat{b}_1$	$\widehat{b}_2$	$\widehat{b}_3$	$\widehat{b}_4$	$\widehat{b}_5$	$\widehat{b}_6$

Table 1. General Butcher Table

0	0	0	0	0	0	0
c_2	a_{21}	α	0	0	0	0
c_3	a_{31}	a_{32}	α	0	0	0
c_4	a_{41}	a_{42}	a_{43}	α	0	0
c_5	a_{51}	a_{52}	a_{53}	a_{54}	α	0
1	a_{61}	a_{62}	a_{63}	a_{64}	a_{65}	α
	a_{61}	a_{62}	a_{63}	a_{64}	a_{65}	α
	$\widehat{b}_1$	$\widehat{b}_2$	$\widehat{b}_3$	$\widehat{b}_4$	$\widehat{b}_5$	$\widehat{b}_6$

Table 2. Butcher Table For ESDIRK(m) Schemes

Each stage of the **ESDIRK(m)** scheme looks like an implicit 1st order accurate method, e.g. **ESDIRK3** with Q^k to stage solution variable

$$Q^{k=1} = Q^n - \Delta t \, a_{11} \mathcal{F}(Q^{k=1}, t^n + c_1 \Delta t) \tag{24}$$
$$Q^{k=2} = Q^n - \Delta t \left(a_{21} \mathcal{F}(Q^{k=1}, t^n + c_1 \Delta t) + a_{22} \mathcal{F}(Q^{k=2}, t^n + c_2 \Delta t) \right)$$
$$Q^{k=3} = Q^n - \Delta t \left(a_{31} \mathcal{F}(Q^{k=1}, t^n + c_1 \Delta t) + a_{32} \mathcal{F}(Q^{k=2}, t^n + c_2 \Delta t) + a_{33} \mathcal{F}(Q^{k=3}, t^n + c_3 \Delta t) \right)$$
$$Q^{k=4} = Q^n - \Delta t \left(a_{41} \mathcal{F}(Q^{k=1}, t^n + c_1 \Delta t) + a_{42} \mathcal{F}(Q^{k=2}, t^n + c_2 \Delta t) + a_{43} \mathcal{F}(Q^{k=3}, t^n + c_3 \Delta t) + a_{44} \mathcal{F}(Q^{k=4}, t^n + c_4 \Delta t) \right)$$

For $k = 1$, implicitly solve for $Q^{k=1}$. At stage $k = 2$, implicitly solve for $Q^{k=2}$ with the residual from stage $k = 1$, $(a_{21} \mathcal{F}(Q^{k=1}), t^n + c_1 \Delta t)$, a source term. Each stage can be solved with either of the two 1st order schemes described above, dual-time stepping or the Newton scheme. For example, stage $k = 2$, replacing $k = 2$ with $p + 1$

$$Q^{p+1} = Q^n - \Delta t \left(a_{21} \mathcal{F}(Q^{k=1}, t^n + c_1 \Delta t) + a_{22} \mathcal{F}(Q^{p+1}, t^n + c_2 \Delta t) \right) \rightarrow$$
$$\left(Q^{p+1} - Q^p \right) + \Delta t \, a_{22} \mathcal{F}(Q^{p+1}, t^n + c_2 \Delta t) = - (Q^p - Q^n) - \Delta t \, a_{21} \mathcal{F}(Q^{k=1} t^n + c_1 \Delta t)$$

linearizing in Q^{p+1} about Q^p and collecting terms we have,

$$\left[I + \Delta t \, a_{22} \frac{\partial \mathcal{F}(Q^p, t^n + c_2 \Delta t)}{\partial Q^p} \right] (Q^{p+1} - Q^p) = -\Delta t \left(\frac{Q^p - Q^n}{\Delta t} + a_{21} \mathcal{F}(Q^{k=1}, t^n + c_1 \Delta t) + a_{22} \mathcal{F}(Q^p, t^n + c_2 \Delta t) \right) \tag{25}$$

Except for the weighting on the left hand side Jacobian term (a_{22}) and the accumulated fluxes with coefficient weightings (a_{21}, a_{22}), each stage can be attacked with the 1st order dual or Newton subiteration scheme. Subiteration convergence requirements are similar to the original dual/Newton approaches.

A dual timestepping approach to solving each stage can also be used. Adding a pseudo-time derivative and rewriting at stage k of Eq. 23a,

$$\partial_\tau Q + \frac{Q^k - Q^n}{\Delta t} + \sum_{j=1}^{k-1} a_{k,j} \mathcal{F}(Q^j, t^n + c_j \Delta t) + a_{k,k} \mathcal{F}(Q^k, t^n + c_k \Delta t) = 0 \tag{26}$$

In discrete pseudo-time τ (using iteration index s) with 1st order Euler implicit and replacing Q^k with $Q^{k,s+1}$ in Eq. 26 [b],

$$\frac{Q^{k,s+1} - Q^{k,s}}{\Delta\tau} + \frac{Q^{k,s+1} - Q^n}{\Delta t} + \sum_{j=1}^{k-1} a_{k,j}\mathcal{F}(Q^j, t^n + c_j\Delta t) + a_{k,k}\mathcal{F}(Q^{k,s+1}, t^n + c_k\Delta t) = 0 \qquad (27)$$

Linearizing $\mathcal{F}(Q^{k,s+1})$ and writing in "Delta" form

$$\left[\left(1 + \frac{\Delta\tau}{\Delta t}\right) I + a_{k,k}\Delta\tau\mathcal{A}(Q^{k,s})\right]\left(Q^{k,s+1} - Q^{k,s}\right) =$$

$$-\Delta\tau\left(\frac{Q^{k,s} - Q^n}{\Delta t} + \sum_{j=1}^{k-1} a_{k,j}\mathcal{F}(Q^j, t^n + c_j\Delta t) + a_{k,k}\mathcal{F}(Q^{k,s}, t^n + c_k\Delta t)\right) \qquad (28)$$

Now the dual or Newton timestepping subiteration process can be applied at each stage, where after subiteration convergence $Q^{k,s} \rightarrow Q^k$

Results for the vortex propagation using an **80 grid point mesh**, the 3rd order and 4th order **ESDIRK** schemes and the 5th order accurate central difference scheme (made up of the 6th order flux differences and 5th order artificial dissipation) are shown in **Fig. 3** after $100C_r$ for increasing time steps $\Delta t = 0.1, 0.2, 0.5, 1.0$

0	0	0	0	0
$\frac{1767732205903}{2027836641118}$	$\frac{1767732205903}{4055673282236}$	$\frac{1767732205903}{4055673282236}$	0	0
$\frac{3}{5}$	$\frac{2746238789719}{10658868560708}$	$\frac{-640167445237}{6845629431997}$	$\frac{1767732205903}{4055673282236}$	0
1.0	$\frac{1471266399579}{7840856788654}$	$\frac{-4482444167858}{7529755066697}$	$\frac{11266239266428}{11593286722821}$	$\frac{1767732205903}{4055673282236}$
b_j	$\frac{1471266399579}{7840856788654}$	$\frac{-4482444167858}{7529755066697}$	$\frac{11266239266428}{11593286722821}$	$\frac{1767732205903}{4055673282236}$
$\widehat{b}_j$	$\frac{2756255671327}{12835298489170}$	$\frac{-10771552573575}{22201958757719}$	$\frac{9247589265047}{10645013368117}$	$\frac{2193209047091}{5459859503100}$

Table 3. Butcher Table for ESDIRK3

D. Comments on Dual Time Stepping

One of the advantages of using a dual-time-stepping approach, is that upon adequate convergence of the pseudo time step, (tau), the errors associated with approximations , such as, factorization, fringe stencil interfaces (especially at split boundaries where the interface is point to point), decoupled turbulence models, implicit linearization simplifications, etc. are deduced below the level of the chosen time accuracy. As noted above, what level of convergence is needed is still an open research area. But, for example in the **OVERFLOW-D** mode the effect of parallization split boundaries are iterated out.

References

[1]Steger, J. L., "Implicit Finite Difference Simulation of Flow About Arbitrary Geometries with Application to Airfoils," AIAA Paper **77-665, 1977.**

[2]Nichols, R.H., Tramel, R.W., Buning, P.G. "Users Manual for OVERFLOW 2.1, Version 2.1t Aug 2008"

[3]Pulliam, T.H., "Efficient Solution Methods for the Navier-Stokes Equations," Lecture Notes for the von Kármán Institute For Fluid Dynamics Lecture Series: Numerical Techniques for Viscous Flow Computation In Turbomachinery Bladings, von Kármán Institute, Rhode-St-Genese, Belgium, 1985.

---

[b]The notation $Q^{k,s}$ is used to designate that the subiteration process is performed for each stage.

0	0	0	0	0	0	0
$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$	0	0	0	0
$\frac{83}{250}$	$\frac{8611}{62500}$	$\frac{-1743}{31250}$	$\frac{1}{4}$	0	0	0
$\frac{31}{50}$	$\frac{5012029}{34652500}$	$\frac{-654441}{2922500}$	$\frac{174375}{388108}$	$\frac{1}{4}$	0	0
$\frac{17}{20}$	$\frac{15267082809}{155376265600}$	$\frac{-71443401}{120774400}$	$\frac{730878875}{902184768}$	$\frac{2285395}{8070912}$	$\frac{1}{4}$	0
1	$\frac{82889}{524892}$	0	$\frac{15625}{83664}$	$\frac{69875}{102672}$	$\frac{-2260}{8211}$	$\frac{1}{4}$
b_j	$\frac{82889}{524892}$	0	$\frac{15625}{83664}$	$\frac{69875}{102672}$	$\frac{-2260}{8211}$	$\frac{1}{4}$
$\widehat{b}_j$	$\frac{4586570599}{29645900160}$	0	$\frac{178811875}{945068544}$	$\frac{814220225}{1159782912}$	$\frac{-3700637}{11593932}$	$\frac{61727}{225920}$

Table 4. Butcher Table for ESDIRK4

[4]Bijl, H., Carpenter, M.H., Vatsa, V.N., Kennedy CA. "Implicit Time integration schemes for the unsteady compressible Navier-Stokes equations: laminar flow.", J Comput Phys Vol. 179, p.1-17, 2002

[5]van Zuijlen, A.H. Bijl, H., "Implicit and explicit higher order time integration schemes for structural dynamics and fluid-structure interaction computations", Computer and Structures 83, p. 93-105, 2005.

[6]Beam, R. and Warming, R. F., "An Implicit Finite-Difference Algorithm for Hyperbolic Systems in Conservation Law Form," J. Comp. Phys., Vol. 22, 1976, pp. 87-110.

[7]Pulliam, T. H. and Chaussee, D. S., "A Diagonal Form of an Implicit Approximate Factorization Algorithm," J. Comp. Phys.,Vol. 39, p. 347, 1981.

[8]Steger, J. L. and Warming, R. F., "Flux Vector Splitting of the Inviscid Gas Dynamic Equations with Applications to Finite Difference Methods," J. Comp. Phys., Vol. 40, pp. 263-293, 1981.

[9]Rai, M.M., "Navier-Stokes Simulations of Blade-Vortex Interaction Using High-Order Accurate Upwind Schemes," AIAA Paper No. 87-0543, 1987.

[10]Pulliam T. H. and Steger J. L., "On Implicit Finite-Difference Simulations of Three- Dimensional Flow,"AIAA Journal Vol. 18, 1980 p. 159.

[11]OVERFLOW: http://aaac.larc.nasa.gov/ buning/codes.html#overflow

[12]Venkateswaraen, S., Merkle, C., "Dual Time Stepping and Preconditioning for Unsteady Computation", AIAA Paper 95-0078, AIAA 33rd Aerospace Sciences Meeting, Reno, NV, 1995

[13]Pandya, S.A., Venkateswaren, S., Pulliam, T.H., "Implementation of Preconditioned Dual-Time Procedures in OVERFLOW", AIAA Paper 2003-0072, AIAA 41st Aerospace Sciences Meeting, Reno, NV, Jan 2003.

[14]Christopher K.A., Carpenter,M.H., Lewis, R.M. "Low-storage, explicit Runge-Kutta schemes for the compressible Navier-Stokes equations", Applied Numerical Mathematics, 35, P 177-219, 2000.

[15]Butcher, J.C., "Coefficients for the Study of Runge-Kutta Integration Processes", J. Austral. Math. Soc. Ser. B3 (2) p 185-201, 1964

[16]carpenter, M.H., Viken, S.A., Nielsen, E.J., "The Efficiency of High-Order Temporal Schemes", AIAA Paper 2003-0086, AIAA 41st Aerospace Sciences Meeting, Reno, NV, Jan 2003.

[17]Pulliam, T.H., "Artificial Dissipation Models for the Euler Equations", AIAA J., Vol 24, No 12 p. 1931, 1986.
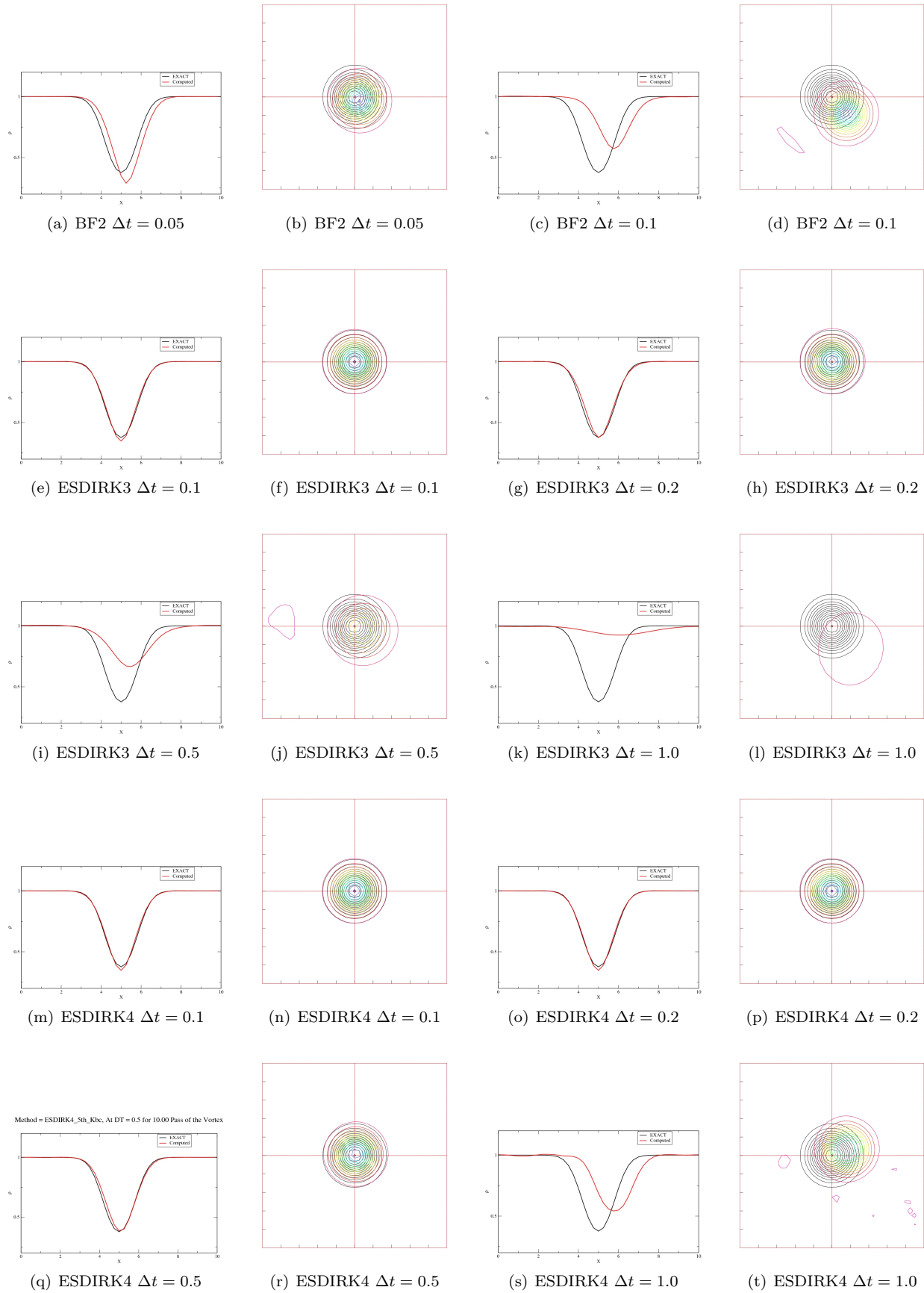
(a) BF2 $\Delta t = 0.05$

(b) BF2 $\Delta t = 0.05$

(c) BF2 $\Delta t = 0.1$

(d) BF2 $\Delta t = 0.1$

(e) ESDIRK3 $\Delta t = 0.1$

(f) ESDIRK3 $\Delta t = 0.1$

(g) ESDIRK3 $\Delta t = 0.2$

(h) ESDIRK3 $\Delta t = 0.2$

(i) ESDIRK3 $\Delta t = 0.5$

(j) ESDIRK3 $\Delta t = 0.5$

(k) ESDIRK3 $\Delta t = 1.0$

(l) ESDIRK3 $\Delta t = 1.0$

(m) ESDIRK4 $\Delta t = 0.1$

(n) ESDIRK4 $\Delta t = 0.1$

(o) ESDIRK4 $\Delta t = 0.2$

(p) ESDIRK4 $\Delta t = 0.2$

(q) ESDIRK4 $\Delta t = 0.5$

(r) ESDIRK4 $\Delta t = 0.5$

(s) ESDIRK4 $\Delta t = 1.0$

(t) ESDIRK4 $\Delta t = 1.0$

Figure 3. **Centerline Density and Contours, Computed Vortex: 100 C_r.**