



AIAA-2001-2612

**Time Integration Schemes for the Unsteady
Navier-Stokes Equations**

Hester Bijl

Delft University of Technology
Delft, The Netherlands

Mark H. Carpenter

NASA Langley Research Center
Hampton, VA

Veer N. Vatsa

NASA Langley Research Center
Hampton, VA

15th Computational Fluid Dynamics Conference

June 11-14, 2001 / Anaheim, CA

TIME INTEGRATION SCHEMES FOR THE UNSTEADY NAVIER-STOKES EQUATIONS

Hester Bijl,^{*} Mark H. Carpenter[†] and Veer N. Vatsa[†]

Abstract

The efficiency and accuracy of several time integration schemes are investigated for the unsteady Navier-Stokes equations. This study focuses on the efficiency of higher-order Runge-Kutta schemes in comparison with the popular Backward Differencing Formulations. For this comparison an unsteady two-dimensional laminar flow problem is chosen, i.e. flow around a circular cylinder at $Re=1200$. It is concluded that for realistic error tolerances (smaller than 10^{-1}) fourth- and fifth-order Runge Kutta schemes are the most efficient. For reasons of robustness and computer storage, the fourth-order Runge-Kutta method is recommended. The efficiency of the fourth-order Runge-Kutta scheme exceeds that of second-order Backward Difference Formula (BDF2) by a factor of 2.5 at engineering error tolerance levels (10^{-1} - 10^{-2}). Efficiency gains are more dramatic at smaller tolerances.

Introduction

Due to constraints of computing costs, the development of numerical techniques for fluid flow simulations in the past has focused mainly on steady state calculations. However, many physical phenomena of interest are inherently unsteady; a few examples being separated flows, wake flows and buffet, fluid actuators and maneuvering. With the continuous reduction of computer costs recently more attention is devoted to the simulation of these unsteady flows^{1,4}. However, the need for further reduction of computer time for unsteady flow computations is still apparent. Therefore, in this paper we investigate possible reductions in computer time due to the choice of an efficient time integration scheme from a series of

schemes ranging from first to fifth order. A similar study was performed for only first and second order time integration methods by Marx¹¹.

Implicit in any comparison of efficiency is a precise error tolerance requirement. Unfortunately, seldom does a single scheme prove to be optimal over a wide range of solution error tolerances. It is well known that high-order schemes (fourth-, fifth-, etc.) outperform low order schemes for error tolerances that are small ($\approx 10^{-7}$). For example, Kennedy⁸ compares explicit third-, fourth- and fifth-order Runge-Kutta (RK) schemes on Direct Navier-Stokes simulations (DNS), and determines the optimal order for a given temporal error tolerance. For DNS it was found, the fourth-order methods are optimal over a surprisingly broad range of error tolerances, and are competitive at large error tolerances as well.

The hallmark of large-scale aerodynamics calculations is that they seldom require small error tolerances. Calculations that are accurate to one or two significant digits, which translates into an error tolerance of (10^{-1} - 10^{-2}), are frequently sufficient. For these aerodynamics calculations, the second-order accurate BDF2 scheme is currently the method of choice. There is little question that calculations requiring low error tolerances (10^{-4} - 10^{-5}) will be well suited for fourth-order RK formulations. The central question of this study, however, is the feasibility of using fourth-order RK formulations for simulations requiring error tolerances of (10^{-1} - 10^{-2}).

A production aerodynamics solver is needed for this study. For this, the extensively tested and well documented solver of Vatsa (TLNS3D)¹⁸ is chosen. This multi-block structured grid solver is representative of a broad class of commonly used solvers. The TLNS3D (Thin-Layer Navier-Stokes 3-Dimensions) code utilizes a special form of the unsteady thin-layer Navier-Stokes equations. The spatial terms are discretized using a conventional cell-centered finite volume scheme with artificial dissipation added for stability. Time is discretized in a fully implicit sense using both multistep BDF and multistage RK schemes. The resultant nonlinear algebraic equations are solved iteratively in pseudo-time with a multi-grid acceleration used to speed up the conver-

^{*}Professor, Department of Aerospace Engineering, Delft University of Technology, Delft, The Netherlands

[†]Senior Research Scientist, Computational Methods and Simulation Branch, NASA Langley Research Center, Hampton, VA 23681-0001

Copyright ©2001 by the American Institute of Aeronautics and Astronautics, Inc. No copyright is asserted in the United States under Title 17, U.S. Code. The U.S. Government has a royalty-free license to exercise all rights under the copyright claimed herein for government purposes. All other rights are reserved by the copyright owner.

gence to (pseudo-time) steady-state.

In this paper, first, the governing flow equations and the space discretization are given. Thereafter, the time discretization techniques employed, i.e. several multi step Backward Differencing Formulations and multistage Runge-Kutta schemes, are extensively discussed. This is followed by a description of the solution algorithm. That is, the implicit time integration of the flow equations and the iterative algorithm for solving the resulting non-linear implicit equation. Through a pseudo-time stability analysis the exact nature of the pseudo-time sub-iterations is analyzed. After validation of the space-time method for unsteady laminar flow around a circular cylinder, accuracy and efficiency of the time integration schemes is discussed.

Governing Equations

In the present work, a modified version of the thin-layer Navier-Stokes equations is used to model the flow. The equation set is obtained from the complete Navier-Stokes equations by retaining only the viscous diffusion terms normal to the solid surfaces. For a body-fitted coordinate system (ξ, η, ζ) fixed in time, these equations can be written in the conservative form as:

$$\frac{\partial(\mathbf{U})}{\partial t} + \frac{\partial(\mathbf{F} - \mathbf{F}_v)}{\partial \xi} + \frac{\partial(\mathbf{G} - \mathbf{G}_v)}{\partial \eta} + \frac{\partial(\mathbf{H} - \mathbf{H}_v)}{\partial \zeta} = 0, \quad (1)$$

where \mathbf{U} represents a combination of the transformation Jacobian \mathbf{J} and the conserved variable vector. The vectors \mathbf{F} , \mathbf{G} , \mathbf{H} , and \mathbf{F}_v , \mathbf{G}_v , \mathbf{H}_v represent the convective and diffusive fluxes in the three transformed coordinate directions, respectively. These equations represent a generalized form of the classical thin-Layer Navier-Stokes equations and include all normal components of the viscous stress terms. The TLNS3D computer code, is used in this study to solve equation (1). Many references exist detailing the discretization and implementational issues of TLNS3D. We include only a brief summary of the general features, and refer to the work of Vatsa ¹⁸ for further details.

Space Discretization

The spatial terms in Equation (1) are discretized using a standard cell-centered finite volume scheme. The convection terms are discretized with second-order central differences with scalar/matrix artificial dissipation (second- and fourth- difference dissipation) added to suppress the odd-even decoupling

and oscillations in the vicinity of shock waves and stagnation points ¹⁸. The viscous terms are central differenced with second-order formulas. The turbulence models used are Baldwin-Lomax ², and Spalart-Allmaras ¹⁵.

Time Discretization

Consider the integration of the system of ordinary differential equations (ODE's) represented by the equation

$$\frac{d\mathbf{U}}{dt} = \mathbf{S}(t, \mathbf{U}(t)) .$$

The vector \mathbf{S} in our case, results from the semi-discretization of the equations of fluid mechanics plus a suitable turbulence model. Inclusion of the turbulence model enables the simulation of high Reynolds number flows in excess of 10^7 , but indirectly introduces extremely fine length-scales in the near-wall regions. Near wall stiffness in the range of $10^3 - 10^4$ is not uncommon in practical engineering problems, and increases with Reynolds number. It is imperative that any efficient solver of these equations be able to maintain stability at arbitrarily large time steps, thus allowing the potential to “step over” unimportant boundary layer time scales.

The term “stiffness” is difficult to define. A practical definition for the purpose of this work, compares an implicit timestep Δt^I with a baseline explicit timestep Δt^E . The implicit timestep is the maximum allowed by accuracy considerations, and the baseline explicit timestep is obtained from stability consideration. All other variables are identical in the comparison. The stiffness is the ratio of the two timesteps. The implicit scheme is capable of stepping over large negative eigenvalues that are not important for solution accuracy, while the explicit scheme must bound them in the finite stability envelop.

There are two mathematical properties that all candidate numerical integrators should possess. The first (and most important) is the “A-stability” property which guarantees that all eigenvalues lying in the left half of the complex plane (LHP) will have an amplification of no more than 1, independent of the chosen step size. The only restriction on the time-step with an A-stable scheme is the consideration of solution accuracy. The second is the “L-stability” property which guarantees that eigenvalues approaching $-\infty$ are damped in one time-step. These spurious eigenvalues are generated by high frequency information in the spatial discretization, and by incomplete solution of the non-linear system at each time-step. (The nonlinear system is never con-

verged to the levels of machine precision). If these spurious modes are not strongly damped they can build up and cause instability.

Most numerical methods suitable for these computations can be categorized into two broad classes: 1) multistep, and 2) multistage, each having its advantages and disadvantages. The current “methods of choice” in the computation of large scale engineering flows are the multistep BDF formulas, and in particular the BDF2 scheme. These schemes achieve great efficiency because they solve only one nonlinear set of equations per time-step. They suffer, however, from not being self-starting, are difficult to use with variable time steps, and are not A-stable beyond second-order temporal accuracy. Multi-stage Runge-Kutta schemes require multiple nonlinear solves per time-step, but are self starting, are easily implemented in variable time-stepping mode, and can be designed with A- and L-stability properties for any temporal order.

Practical experience indicates that large scale engineering computations are seldom stable if run with BDF4 ¹². The BDF3 scheme is often stable, but diverges for certain problems and some spatial operators. These failures result from portions of the Left Half Plane that are not stable for arbitrarily large timesteps. Thus, a reasonable practitioner uses the BDF2 scheme exclusively for large scale computations. The essential question this paper seeks to address is whether high-order RK schemes can be designed which are more efficient than the BDF2 schemes, and if so, what is the optimal order of the RK scheme.

The general formula for a k -step BDF scheme can be written as

$$\mathbf{U}^{(n+k)} = -\sum_{i=0}^{k-1} \alpha_i \mathbf{U}^{(n+i)} + (\Delta t) \beta_k \mathbf{S}^{(n+k)} \quad (2)$$

The BDF formulas involve at each time-step the storage of $k + 1$ levels of the solution vector \mathbf{U} , and the implicit solution of one set of nonlinear equations. The BDF schemes are subject to the famous Dahlquist barrier, which proves that A-stability is impossible for linear multistep schemes beyond second order. Table (1) presents the stability properties and diagonal contribution β_k of several BDF schemes.

Method	Steps	Order	$L(\alpha)$	β_k
BDF1	1	1	$L(90^\circ)$	1
BDF2	2	2	$L(90^\circ)$	$\frac{2}{3}$
BDF3	3	3	$L(86.03^\circ)$	$\frac{6}{11}$
BDF4	4	4	$L(73.35^\circ)$	$\frac{12}{25}$
BDF5	5	5	$L(51.84^\circ)$	$\frac{60}{137}$
BDF6	6	6	$L(17.84^\circ)$	$\frac{60}{147}$

Table 1. *Properties of the BDF methods.*

The BDF1 and BDF2 schemes are L-stable for eigenvalues anywhere in the left half of the complex plane. Beyond second order the BDF schemes are $L(\alpha)$ -stable, with the α being defined in Table (1). This definition of stability implies that the schemes will be stable for eigenvalues lying in the wedge bounded above and below by the complex lines $\pm\alpha$. For further details see the work of Hairer ⁶.

The Runge-Kutta methods are multistage schemes and are implemented as

$$\begin{aligned} \mathbf{U}^k &= \mathbf{U}^n + (\Delta t) \sum_{j=1}^k a_{ij} \mathbf{S}(\mathbf{U}^j), \quad k = 1, s \\ \mathbf{U}^{n+1} &= \mathbf{U}^n + (\Delta t) \sum_{j=1}^s b_j \mathbf{S}(\mathbf{U}^j) \\ \hat{\mathbf{U}}^{n+1} &= \mathbf{U}^n + (\Delta t) \sum_{j=1}^s \hat{b}_j \mathbf{S}(\mathbf{U}^j), \end{aligned} \quad (3)$$

where s is the number of stages and a_{ij} and b_j are the Butcher coefficients of the scheme. The vectors \mathbf{U} , and $\hat{\mathbf{U}}$ are the p^{th} -order and $(p-1)^{th}$ -order solutions at time level $n+1$. The vector $\hat{\mathbf{U}}$ is used solely for estimating error and is obtained with little additional computational overhead. In this work we choose to focus on the ESDIRK class of RK schemes, which refers to the *Explicit* first stage, *Single* diagonal coefficient, *Diagonally Implicit Runge-Kutta*. The Butcher tableau for these schemes (here represented with $s = 5$) takes the form

0	0	0	0	0	0
c_2	a_{21}	a_{22}	0	0	0
c_3	a_{31}	a_{32}	a_{33}	0	0
c_4	a_{41}	a_{42}	a_{43}	a_{44}	0
1	b_1	b_2	b_3	b_4	a_{kk}
	\hat{b}_1	\hat{b}_2	\hat{b}_3	\hat{b}_4	a_{kk}
	\hat{b}_1	\hat{b}_2	\hat{b}_3	\hat{b}_4	\hat{b}_5

where c_i denotes the point in the time interval $t \rightarrow t + \Delta t$ where the intermediate stage is evaluated, and \hat{b}_j are the coefficients used for the embedded error predictor. The fully implicit RK schemes are not

pursued owing to the complexity of their implementation in our general aerodynamics solver TLNS3D. The ESDIRK schemes are used rather than the standard SDIRK schemes ($a_{i1} = 0$, see Hairer ⁷) because the ESDIRK schemes can achieve stage order 2 (each stage is at least second-order accurate), and the extra column of nonzero Butcher coefficients (a_{11}) allows more flexibility when designing new schemes. For all schemes, the “stiffly-accurate” assumption ($a_{sj} = b_j$) is enforced, which automatically extends A-stability into L-stability. In addition, it eliminates the potentially damaging explicit update $U^{n+1} = U^n + \Delta t \sum_{j=1}^s b_j S^j$ from the algorithm, and replaces it with the condition $U^{n+1} = U^s$.

Numerous ESDIRK schemes were developed and tested in this study. Three schemes, one each of third-, fourth-, and fifth-order accuracy were chosen as being representative of each of these orders. No claim is made as to their optimality. Henceforth, they are referred to as ESDIRK3, ESDIRK4, and ESDIRK5, respectively. All three schemes are presented in Kennedy ⁹. The coefficients for the recommended ESDIRK4 scheme are included in the Appendix.

Solution Algorithm

Discretizing equation (1) with an s -stage ESDIRK scheme represented in equation (4) yields for stage k , the expression

$$\frac{\mathbf{U}^k - \mathbf{U}^n}{\Delta t} + \sum_{j=1}^k a_{kj} \mathbf{S}^j = 0, \quad k = 1, s \quad (4)$$

with

$$\mathbf{S}^j = \frac{\partial(-\mathbf{F} + \mathbf{F}_v)}{\partial \xi} + \frac{\partial(-\mathbf{G} + \mathbf{G}_v)}{\partial \eta} + \frac{\partial(-\mathbf{H} + \mathbf{H}_v)}{\partial \zeta}.$$

The vector \mathbf{U}^k is the solution at stage k , \mathbf{U}^n is the solution at the previous time level n , and a_{kj} are the Butcher parameters for the RK-method used. Again note that in our case the last stage gives the solution at the new time level, that is $\mathbf{U}^{n+1} = \mathbf{U}^s$. Advancing the solution vector \mathbf{U}^k from time level $n \rightarrow n + 1$, requires solving the sequential set of s nonlinear algebraic equations defined in equation (4).

Pseudo-Time Iterative Algorithm

We follow here the work of Melson et al.¹² which was originally developed for a BDF algorithm. Equation (4) can be difficult to solve in its present

form. Thus, the pseudo-time term $\frac{\partial \mathbf{U}^k}{\partial \tau}$ is added to each stage k , yielding the expression

$$\frac{\partial \mathbf{U}^k}{\partial \tau} + \frac{\mathbf{U}^k - \mathbf{U}^n}{\Delta t} + \sum_{j=1}^k a_{kj} \mathbf{S}^j = 0, \quad k = 1, s. \quad (5)$$

In this form, equation (5) is amenable to all the nonlinear solving machinery available in TLNS3D. Each nonlinear equation is marched in pseudo-time with multi-grid acceleration until a predetermined convergence criterion is satisfied.

Equation (5) can be rewritten in the form

$$\frac{\partial \mathbf{U}^k}{\partial \tau} + \frac{\mathbf{U}^k}{\Delta t} + a_{kk} \mathbf{S}^k + E = 0, \quad k = 1, s \quad (6)$$

with

$$E = \frac{-\mathbf{U}^n}{\Delta t} + \sum_{j=1}^{k-1} a_{kj} \mathbf{S}^j,$$

where E includes all the iteration independent information at stage k of the RK scheme. Discretizing equation (6) in the variable τ using an Implicit-Explicit Runge-Kutta (RK_τ) scheme, yields

$$\frac{\mathbf{U}^{kp} - \mathbf{U}^{k0}}{\Delta \tau} + \alpha_p \left(\frac{\mathbf{U}^{kp}}{\Delta t} + a_{kk} \mathbf{S}^{k(p-1)} + E \right) = 0 \quad (7)$$

where the p is the stage value of the RK_τ scheme. Note that the contribution from the time term is treated implicitly, the inviscid and viscous flux terms are treated explicitly, and that first-order temporal accuracy is sufficient for this scheme. Adding, subtracting, rearranging terms and accounting for residual smoothing in equation (7) yields

$$\left(1 + \frac{\alpha_p \Delta \tau}{\Delta t} \right) \mathbf{U}^{kp} = \mathbf{U}^{k0} + \frac{\alpha_p \Delta \tau}{\Delta t} \mathbf{U}^{k(p-1)} - \alpha_p \Delta \tau \mathbf{L}_{irs}^{-1} \left[\frac{\mathbf{U}^{k(p-1)}}{\Delta t} + a_{kk} \mathbf{S}^{k(p-1)} + E \right], \quad (8)$$

where \mathbf{L}_{irs}^{-1} is the implicit residual smoothing matrix. Note that the bracketed term in equation (8) is still the residual of the physical time RK scheme.

Pseudo-Time Stability Analysis

Each grid level of the multi-grid process yields an equation similar to equation (8), with minor differences in the variable E accounting for grid transfer contributions. The pseudo-time sub-iteration should converge rapidly, and any remaining residual at the end of the iteration should be devoid of high-frequency content. A Fourier stability analysis is used to analyze the exact nature of the pseudo-time

sub-iteration and to optimize the residual smoothing parameters.

A necessary first test for the sub-iteration algorithm is good behavior for the 1-D wave equation $U_t + aU_x = 0$. Convergence and stability are strong functions of the discrete spatial operator U_x . TLNS3D uses spatial discretizations that closely resemble an approximation of the form

$$\frac{\partial U}{\partial x} = \frac{U_{i+1} - U_{i-1}}{2\Delta x} + \frac{U_{i+2} - 4U_{i+1} + 6U_i - 4U_{i-1} + U_{i-2}}{32\Delta x}.$$

The analysis begins by substituting this spatial operator into the 1-D wave equation on a periodic domain, discretizing time using the ESDIRK scheme, and assuming solutions of the form $U(x, t) = \hat{u}(t)e^{i(j\theta)}$. Implementing the pseudo-time sub-iteration algorithm described in equation (8) then yields for the p^{th} stage of the RK_τ operator, the expression

$$(1 + \alpha_p R)g^p = g^0 + \alpha_p Rg^{p-1} - \alpha_p L_{irs}^{-1}[Rg^{p-1} + a_{kk}\lambda\phi g^{p-1}], \quad (9)$$

with

$$\phi = i \sin(\theta) + \frac{1}{2} \sin(\theta/2)^4$$

$$L_{irs}^{-1} = (1 + 4\gamma \sin(\theta/2)^2)^{-1},$$

and

$$g^p = \frac{\hat{u}^p}{\hat{u}^0}; \quad R = \frac{\Delta\tau}{\Delta t}; \quad \hat{\lambda} = a_{kk} \frac{a\Delta\tau}{\Delta x};$$

The variable E is eliminated from consideration in the stability analysis because it is constant during the pseudo-time sub-iteration and therefore has no influence on the stability. The diagonal coefficient a_{kk} is the same on each stage of an ESDIRK, which yields an identical stability analysis for each stage. (A stability analysis for the BDF scheme given by equation (2) yields a similar result, with the variable a_{kk} replaced by β_k .)

Several observations about equation (9) are instructive. The independent parameters are γ : the level of implicit residual smoothing, R : the ratio of time-steps, and $\hat{\lambda}$: the CFL condition for the pseudo-time-stepping scheme multiplied by the diagonal coefficient from the ESDIRK scheme. In general the parameter R satisfies the condition $0 \leq R \leq 1$. The limit $R \rightarrow 0$ or $\Delta t \rightarrow \infty$ reduces to the steady state formulation. The limit $R \rightarrow 1$ corresponds to a physical time-step for which an explicit method would be stable, a situation that is unlikely to occur

in high Reynolds number calculations.

An exhaustive study in the parameters $\gamma, R, \hat{\lambda}$ yields the following general conclusions. The residual smoothing parameter γ is most productive when the effective stability limit of the sub-iteration is increased by a factor 2 – 3. Parameter values in the range $1.0 \leq \gamma \leq 1.5$ produce this increase. Figures (1)-(3) show a comparison of the damping characteristics as a function of $\hat{\lambda}$, with the value $\gamma = 1.5$. Shown are the cases $R = 0, R = \frac{1}{10}$, and $R = 1$, with $1 \leq \hat{\lambda} \leq 7$ varying on each plot. Plots of $R < \frac{1}{10}$ are virtually indistinguishable from the case $R = 0$. The three plots reveal that the pseudo-time sub-iteration is stable for $\hat{\lambda} \approx 7$, and has the best damping characteristics in the $\frac{\pi}{2} \leq \theta \leq \pi$ range for values of $\hat{\lambda}$ near the stability limit.

The parametric study yields the following algo-

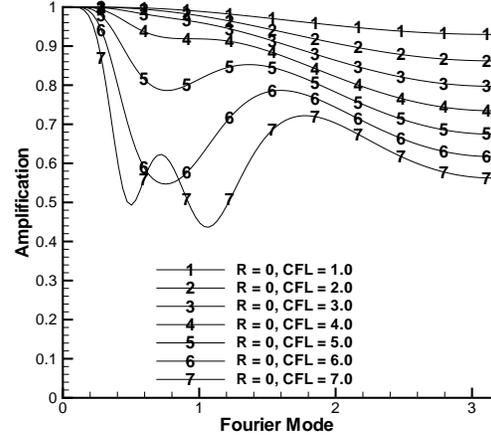


Figure 1. *Damping characteristics for the case $R = \frac{\Delta\tau}{\Delta t} = 0$, with “CFL” = $\hat{\lambda}$.*

rithm for the pseudo-time sub-iteration. The RK_τ scheme is implemented in five stages with the coefficients being defined by $\vec{\alpha} = [1/4, 1/6, 3/8, 1/2, 1]$. Three evaluations of artificial dissipation terms (computed at the odd stages) are used to obtain a larger stability bound, which allows a higher CFL number in the presence of physical diffusion terms. The stability limit of the numerical method is further increased with the use of the implicit residual smoothing technique that employs grid aspect-ratio-dependent coefficients¹⁸ and local time-stepping is used in each cell. The efficiency of the solution process is significantly enhanced through the use of a multigrid acceleration technique. For a detailed description of the resulting iterative method see Vatsa et al.¹⁸ The same concept for the computation of

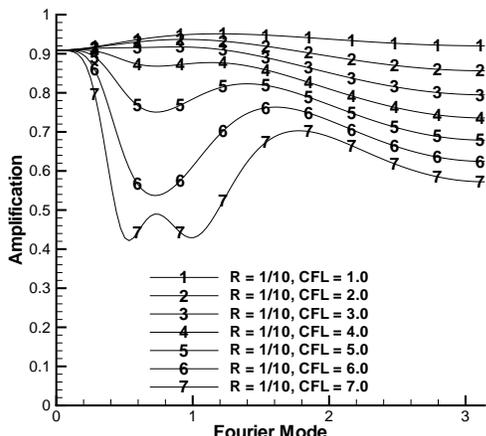


Figure 2. Damping characteristics for the case $R = \frac{\Delta\tau}{\Delta t} = \frac{1}{10}$, with “CFL” = $\hat{\lambda}$.

unsteady flows is used by other researchers, see for example Martinelli¹⁰ and Swanson and Turkel¹⁶.

Diagonal Coefficients

The pseudo-time sub-iteration is strongly influenced by the diagonal coefficients a_{kk} and β_k . In TLNS3D the pseudo-time sub-iteration is always advanced with the maximum allowable *scaled* pseudo-time-step $\hat{\lambda} = a_{kk}\lambda \approx 7$. The rate of relaxation in *non-scaled* pseudo-time is therefore, inversely proportional to the diagonal coefficient a_{kk} : the smaller the value of a_{kk} the more rapidly the pseudo-time sub-iteration progresses. The values a_{kk} and β_k vary significantly for different physical time-advancement schemes. Table (1) presents the diagonal coefficients β_k for the BDF schemes, where the coefficients vary by approximately a factor of 2. Table (2) presents the diagonal contribution of the ESDIRK schemes, as well as the number of stages, the implicit stages and the order. The a_{kk} coefficients vary by approximately a factor of two, and are generally much smaller than the BDF β_k values.

Unlike the BDF schemes, the ESDIRK schemes can be optimized to improve their efficiency. The important parameters are the diagonal coefficient a_{kk} and the number of stages s . A fortuitous trend observed with the ESDIRK schemes is a general decrease in the value of a_{kk} with increasing stage number, at a given accuracy and L -stability property. Increasing the number of implicit stages does not always decrease the efficiency of the scheme, and sometimes yields *greater* efficiency. Fifteen ESDIRK schemes ranging from 3 to 8 stages were investigated to determine good choices of third-, fourth-, and

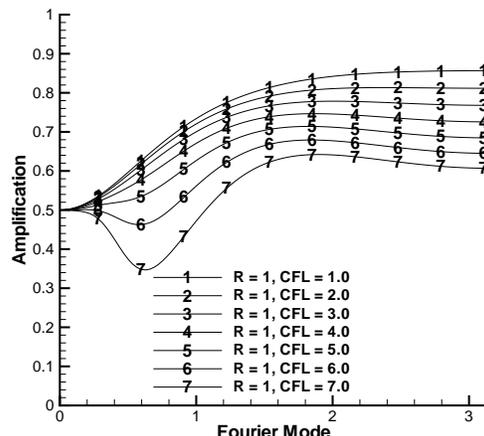


Figure 3. Damping characteristics for the case $R = \frac{\Delta\tau}{\Delta t} = 1$, with “CFL” = $\hat{\lambda}$.

fifth-order accuracy. The ESDIRK3, ESDIRK4, and ESDIRK5, used in this study (as well as the work of Kennedy⁹) are representative candidates from this study of ESDIRK schemes. The value $a_{kk} = \frac{1}{4}$ in the ESDIRK4 scheme is an example of five implicit stages producing a more efficient fourth-order scheme than do four. This is consistent with the findings of other investigators (see Hairer⁷).

Method	Stgs	Impl Stgs	Order	a_{kk}
ESDIRK3	4	3	3	0.435
ESDIRK4	6	5	4	0.250
ESDIRK5	7	6	5	0.184

Table 2. Properties of the ESDIRK methods.

Validation of the Space-Time Method

The accuracy of the space-time integration methods is investigated for an unsteady laminar flow test case. The test problem is laminar flow around a two-dimensional circular cylinder at a Reynolds number of 1200 and a Mach number of 0.3. The initial flow is symmetric with zero lift. As the wake behind the cylinder starts to grow, it becomes unstable and begins to shed from alternate sides of the cylinder. Detailed numerical and experimental investigations of this flow have been performed by several authors^{3, 5, 14, 16}. The computational grid of 97×65 is shown in Figure 4. The boundary is a distance of 20 times the diameter of the cylinder away from the wall, while the distance between the wall and the first grid point is 0.001 times the diameter of the cylinder. Grid points are clustered in the wake. A density contour plot is shown in Figure 5 as calculated on the 97×65 grid. Note that the near wall

vortical structures appear to be sufficiently resolved, but that resolution is lost as the grid expands in the far-field. In these preliminary calculation, a small time-step was chosen, so that the dominant component of error is the spatial contribution.

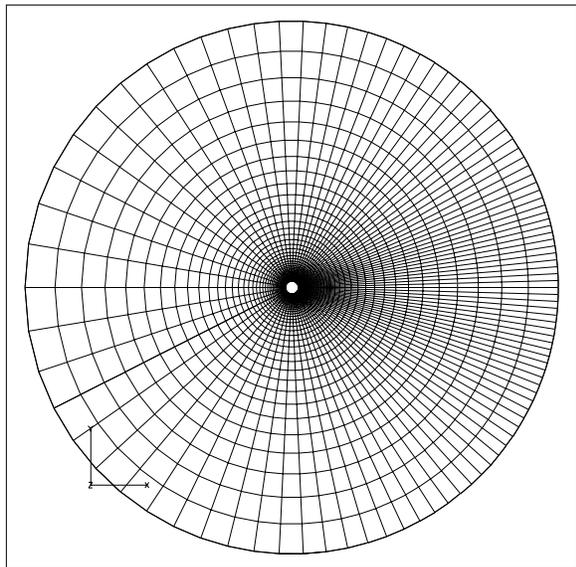


Figure 4. The 97×65 O-Grid used in the circular cylinder study.

Figure 6 shows a comparison of the lift history for one shedding cycle as calculated on the 97×65 and 193×129 grids. The 193×129 data is shifted so that the first zero in lift coincides on both curves. The Strouhal number in each case was 0.2489, and 0.2459 as calculated on the 97×65 and 193×129 grids, respectively. Small differences in the lift are seen near the peaks of the cycle. These values are larger than 0.21, reported from experiments^{4, 14}. As Mittal and Balachander report¹³ this might be caused by the onset of three-dimensional effects, not captured in our two-dimensional computations. Two-dimensional computations performed by other researchers^{5, 17} resulted in larger Strouhal numbers too, in the range of 0.23-0.24. This study and a more exhaustive study of inviscid vortex propagation support the conclusion that the space-time operator is working properly, and that the spatial operator converges at the design spatial accuracy. The coarse grid (97×65) provides adequate spatial resolution to capture the relevant large scales features in the shedding process. As such, the 97×65 grid is chosen as the basis of most of the temporal refinement studies presented in this work.

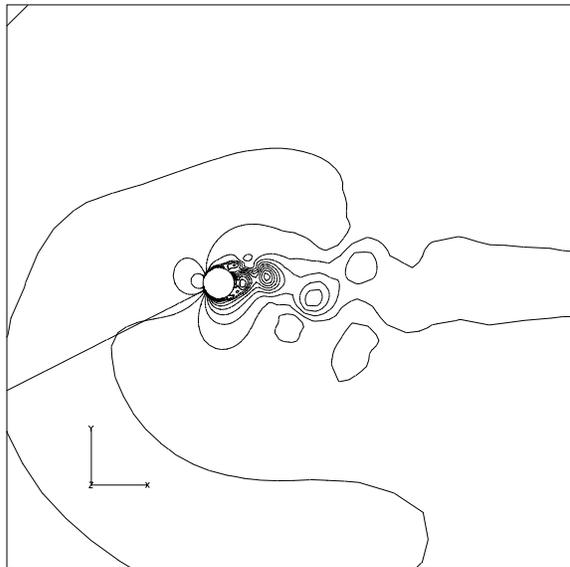


Figure 5. The density contours as calculated on the 97×65 O-Grid.

Temporal Accuracy

A temporal refinement study is performed to assess the accuracy of various ESDIRK and BDF schemes. The initial condition for the study was obtained by simulating the limit cycle behavior of the flow for approximately 20 shedding cycles, with a relatively small time-step ($\Delta t = \frac{1}{2}$). After 20 cycles, the solution was stored in a restart file for use as the initial condition in the subsequent studies. A classical temporal study was then performed from this initial condition. A typical practitioner is interested in lift, drag, pitching moment, skin friction, and the frequency spectrum over several cycles. Thus, the time interval of the study includes approximately $1\frac{1}{4}$ shedding cycles. This interval is sufficient in length to allow accumulation of temporal error during the shedding cycle. No exact solution is known for this problem, so a “numerical exact” solution was obtained using a small time-step and a small iteration tolerance on the non-linear solves. The “exact” time-step used was $\Delta t = 0.05$, and a stopping criterion for the iterative solve of the implicit equations, was $\max(\text{residual}) < 10^{-6}$. The “exact” solution is accurate to approximately 6 significant digits in the lift.

The lift on the body is used as the representative measure of error in all calculations. Other integral measures including drag, skin friction, total drag, pitching moment, as well as L_2 and L_∞ norms over the domain were studied. All integral measures yield nearly the same quantitative conclusions, although

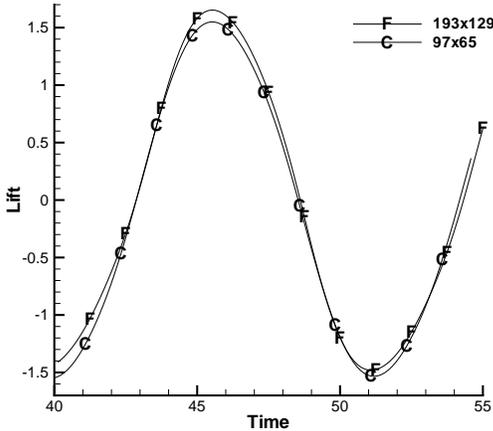


Figure 6. Comparison of lift over one shedding cycle between coarse and fine grids

the qualitative errors are different for each case. A detailed investigation of the location of the maximum temporal error revealed that the vortex generation process in the near-wall regions is the most temporal demanding portion of the cycle. It is not surprising, therefore, that the lift integral is a good measure of total error in the calculation.

Detailed results from the study are now presented. Figure 7 shows a detailed refinement study with an ESDIRK4 scheme. Shown are the solution errors in the lift, viscous drag, total drag, and pitching moment, as a function of logarithm of time-step. In all cases, the nonlinear system is solved to strict tolerances to eliminate “iteration” error as a contaminating variable in the study. At coarse time-steps the solution accuracy deteriorates away from design accuracy. For sufficiently small time-steps, design accuracy is clearly demonstrated in all variables. For example, a least-squares fit of the finest four data points on each curve reveals that the convergence rates for [lift, drag V, drag T, pitch] are [3.9628, 4.0257, 4.0478, 4.0251]. The coarsest data point corresponds to a time-step of $\Delta t = 2$, for which twelve points resolve the shedding cycle. Note that 15 – 20 points are needed to resolve the cycle before the fourth-order scheme converges with design accuracy. This resolution is consistent with conventional estimates of “points-per-wavelength” needed to resolve a periodic wave.

Figure 8 presents the error in lift versus the time step (log-log) for three BDF schemes and three ESDIRK schemes. The ESDIRK schemes presented in this figure are summarized in table (2). Again, the

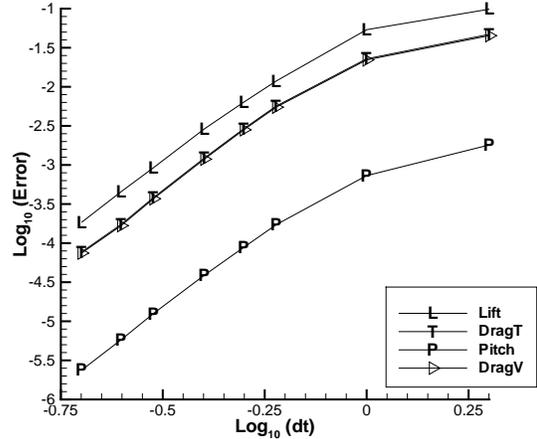


Figure 7. Convergence behavior for lift, drag and pitching moment as calculated with a fourth-order ESDIRK scheme.

nonlinear equations at each stage (step) are solved to tolerances which are small compared with the absolute error in the calculation. There is a dramatic increase in accuracy in going from BDF1 to ESDIRK5. For example, an error tolerance of 10^{-1} is achieved with a time-step of $\Delta t = 1$ for the ESDIRK4 and ESDIRK5 schemes, while the BDF1 and BDF2 require $\Delta t = 10^{-2}$ and $\Delta t = 10^{-1}$, respectively. The BDF3 and the ESDIRK3 schemes have nearly the same absolute level of error, although the convergence behavior of the BDF3 scheme is sporadic. No explanation of this behavior was identified, although a possible explanation is the schemes lack of A-stability.

As mentioned previously in this work, the BDF2 scheme is consistently used by practitioners because of its robustness, simplicity and efficiency. The results in Figure 8 clearly show that the ESDIRK4 scheme can be used at time-steps which are a factor of ten larger than those used in the BDF2, while achieving similar accuracy. At fine tolerances, this difference becomes even greater. Figure 8 still can not be used to conclude that the ESDIRK4 scheme is more *efficient* than the BDF2 scheme. To do so requires a detailed accounting of the work involved in each algorithm, and is not a simple task.

Temporal efficiency

For large computations, the work involved in advancing the solution from $t = T_0$ to $t = T_f$ is proportional to the number of non-linear solves required over that interval, but also depends strongly

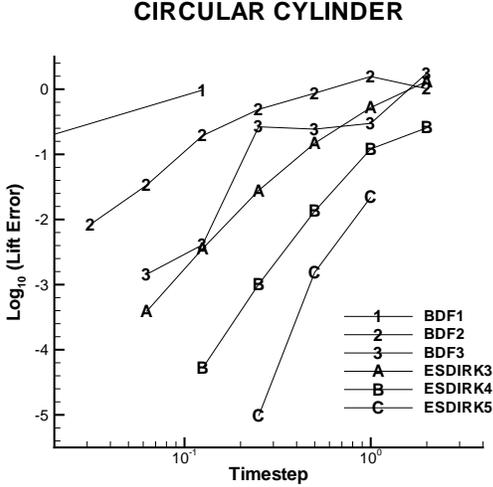


Figure 8. Convergence as a function of time-step for BDF and ESDIRK schemes.

on how quickly each non-linear solve converges. The number of non-linear solves is simply $N_L = \frac{T_f - T_0}{\Delta t} I_s$, where I_s is the number of implicit stages per time-step. (An explicit stage is virtually “free”: hence the use of the ESDIRK instead of the SDIRK schemes.) For the BDF schemes $I_s = 1$. A much more difficult aspect to predict, however, is how rapidly a nonlinear solve will converge. Smaller physical time-steps provide a much better initial guess for the non-linear iteration, which implies an advantage for the BDF schemes. The ESDIRK schemes, however, have a much smaller diagonal coefficient a_{kk} which increases the asymptotic convergence rate of the multi-grid process.

Figure 9 presents the convergence of the six schemes as a function of the required work. Three accuracy levels are chosen, $[10^{-1}, 10^{-2}, 10^{-3}]$ as representative of desired engineering accuracy levels. The appropriate values of Δt needed for each method are obtained from Figure 8. An iteration tolerance a factor of 200 times smaller than the desired error level, is used for the nonlinear iteration, with the error based on the L_∞ norm of the residual. For example, if the desired error is 10^{-2} then at each stage (step) the nonlinear system is solved until the maximum residual is $\frac{10^{-2}}{200}$. The work from each method is measured as the total number of multi-grid cycles used in the entire time interval.

An obvious conclusion from the study presented in Figure 9 is that the BDF1 scheme (Euler Implicit) will never compete with the higher-order schemes in terms of efficiency. Similarly, the BDF2 scheme is only competitive with the third-, fourth- and fifth-

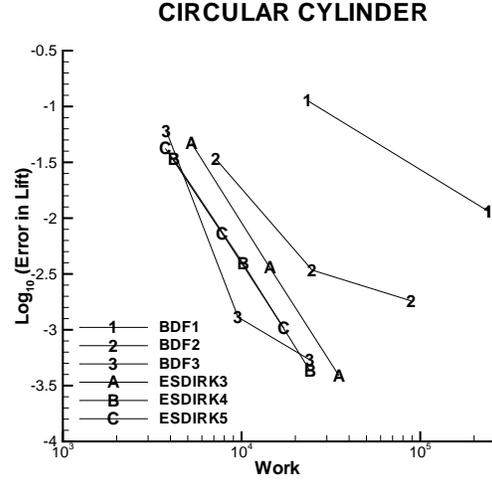


Figure 9. Convergence as a function of work for BDF and ESDIRK schemes.

order scheme at extremely coarse tolerances. For example, the BDF2 is 2.5 times less efficient than the ESDIRK4 scheme at an error tolerance of 10% (one significant digit in the solution accuracy). As the error tolerance becomes more strict, the high-order schemes easily outperform the BDF1 and the BDF2 scheme. For a desired temporal error of 1% in lift the fourth-order integration method ESDIRK4, only requires 1.5% of the work required by BDF1, an efficiency increase with a factor 70. For a temporal error of 0.1% solution ESDIRK4 requires 10% of the work of BDF2, an efficiency increase with a factor 10. Note again that the BDF3 scheme shows an irregular behavior. Perhaps this is due to the fact that the BDF3 solution oscillates around the exact solution for different time-steps, and sometimes by coincidence yields unusually low levels of error.

It is interesting to note that the fourth- and fifth-order ESDIRK schemes have exactly the same accuracy-work ratio, and that their convergence behavior appears to be logarithmic in nature. Both have different time-steps, stages, and diagonal coefficients a_{kk} , and there is no reason why they should lie on the same line, or have logarithmic convergence behavior. Though both have the same efficiency, the ESDIRK5 scheme requires more storage and is less robust than the ESDIRK4 scheme (internal stability problems at huge time-steps). Therefore, for an efficient and robust solution in time the ESDIRK4 scheme is recommended.

A major contributor to the inefficiency of implicit methods is solving the nonlinear systems at each stage (step) to inappropriate sub-iteration tolerance

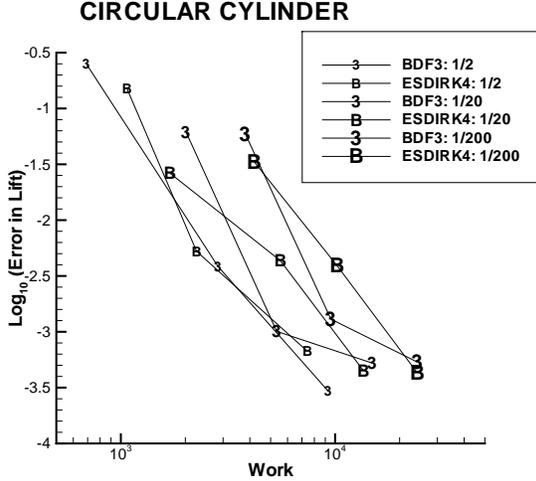


Figure 10. The effects of sub-iteration tolerance on solution accuracy for the BDF3 and ESDIRK4 schemes.

levels. If the nonlinear solve is iterated too many times, the additional work does not increase the solution accuracy or robustness, but does increase the cost. If the nonlinear solve is not iterated enough, then the solution error will be dominated by the errors in the nonlinear solve, robustness will suffer, and the entire solution will be in jeopardy. The solution **cannot** be more accurate than the errors in the nonlinear equations. Unfortunately, it is not known a-priori to what levels the nonlinear equations should be solved. A final study is performed, in which the nonlinear equations are solved to sub-iteration levels which were $[\frac{1}{200}, \frac{1}{20}, \frac{1}{2}]$ of the desired error level. The error tolerances used in the previous study $[10^{-1}, 10^{-2}, 10^{-3}]$ are again used.

Figure 10 shows a plot of the solution accuracies for the BDF3 and the ESDIRK4 schemes. Ideally, increasing the sub-iteration tolerance level, should move the curves uniformly to smaller values of work, until at a critical tolerance level the solution accuracy begins to deteriorate. The sub-iteration tolerances of $\frac{1}{20}$ and $\frac{1}{200}$ yield essentially the same accuracy levels, but significantly differ in the amount of work required. For a sub-iteration tolerance of $\frac{1}{2}$ the solution accuracy begins to degrade for both schemes. Similar results are exhibited with the other ESDIRK schemes. It is concluded that the nonlinear sub-iteration in TLNS3D should be converged to a level which is at least $\frac{1}{10}$ of the desired solution accuracy, independent of the temporal integration method used.

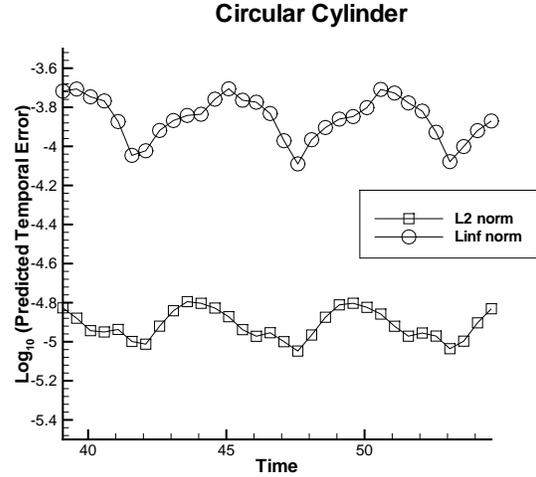


Figure 11. Cycle variation of predicted temporal error as calculated with ESDIRK4

Temporal error predictor

An efficient time advancement scheme is capable of monitoring the solution error, and adjusting the time-step when needed. The ESDIRK schemes excel in the variable time-stepping environment because they are self starting. At each time-step a reliable measure of solution error is needed, however. All the ESDIRK schemes used in this study have embedded schemes available to monitor the solution error. Some norm of the solution error is obtained at each time-step by comparing the main and embedded solutions \mathbf{U} and $\hat{\mathbf{U}}$. The main and embedded solution differ in accuracy by one order. Thus, the difference between the solutions is the leading order error term in the temporal Taylor series expansion, and is proportional to the time-step error. Knowing the solution error at each time-step, the subsequent time-step is adjusted to reflect a desired accuracy tolerance for the calculation. An additional benefit of the error estimate is that a precise stopping criterion for the nonlinear sub-iteration can be implemented.

Figure 11 shows the predicted temporal error as calculated by the ESDIRK4 scheme. The L_2 and L_∞ are presented for the coarse grid case shown in Figure 6. A fixed time-step of $\Delta t = \frac{1}{2}$ was used with a nonlinear sub-iteration tolerance of 0.5×10^{-5} . The temporal error correlates highly with the maximum and minimum lift. Note that about $\frac{1}{2}$ an order variation in temporal error is observed over the shedding cycle. Calculations were successfully performed on this case in variable time-stepping mode using a controller (see Kennedy ⁹ for details).

Work continues into automating the time-step controller. The objective is automation such that the only temporal input is the time-step error. Many additional cases must be run in the future, including fully turbulent flows, to calibrate the controller.

Conclusions

The accuracy and efficiency of several time integration schemes has been investigated for the unsteady Navier-Stokes equations. Time is discretized implicitly, while the spatial discretization is a conventional cell-centered finite volume scheme with artificial dissipation added for stability. The nonlinear equations are solved at each step with a multigrid algorithm. This study focuses on the efficiency of higher-order Runge-Kutta schemes in comparison with the popular Backward Differencing Formulations. For this comparison an unsteady two-dimensional laminar flow problem was chosen, i.e. flow around a circular cylinder at $Re=1200$. It is concluded that for all realistic error levels (smaller than 10^{-1}) fourth- and fifth-order Runge Kutta schemes are the most efficient. For reasons of robustness and computer storage, the fourth-order Runge-Kutta method is recommended. The efficiency of the fourth-order Runge-Kutta scheme exceeds that of second-order Backward Difference Formula (BDF2) by a factor of 2.5 at engineering error tolerance levels (10^{-1} - 10^{-2}). Efficiency gains are more dramatic at smaller tolerances.

Acknowledgment

This work was performed while the first author was in residence at NASA Langley Research Center, Hampton, VA, 23681-0001. Special thanks are extended to Dr. Chris A. Kennedy, Sandia National Laboratories, for his help in fine-tuning the ESDIRK schemes used in this study.

References

1. K.J. Badcock, F. Cantariti, I. Hawkins, M. Woodgate, L. Dubuc and B.E. Richards, "Simulation of unsteady turbulent flows around moving aerofoils using the pseudo-time method", *Int. J. Num. Meth. Fl.*, 32: 585-604, 2000.
2. Baldwin and Lomax, "Thin layer approximation and algebraic model for separated turbulent flows", AIAA Paper 78-257, 1978.
3. G.Bosch and W. Rodi, "Simulation of vortex shedding past a square cylinder with different turbulence models", *Int. J. Num Meth. Fl.*, 28: 601-616, 1998.
4. R. Bouard and M. Coutanceau, "The early stage of the wake behind an impulsively started cylinder for $40 < Re < 104$ ", *J. Fluid Mech.*, 101: 583-607, 1980.
5. J.R. Cox, K.S. Brentner and C.L Rumsey, "Computation of vortex shedding and radiated sound for a circular cylinder: sub-critical to trans-critical Reynolds number", *Theoret. Comp. Fluid Dynamics*, 12:233-253, 1998.
6. E. Hairer S.P. Norsett, and G. Wanner, "Solving ordinary differential equations I: nonstiff problems", Springer-Verlag, 1991.
7. E. Hairer and G. Wanner, "Solving ordinary differential equations II: stiff and differential-algebraic problems", Springer-Verlag, 1991.
8. C.A. Kennedy, M.H. Carpenter, and R.M. Lewis, "Low-Storage, explicit Runge-Kutta Schemes for the compressible Navier-Stokes equations", *Appl. Num. Math*, Vol 35, (2000), pp. 177-219.
9. C.A. Kennedy and M.H. Carpenter, "Additive Runge-Kutta Schemes for Convection-Diffusion-Reaction Equations", Submitted to *Appl. Num. Math.*
10. L. Martinelli and A. Jameson, "Validation of a multigrid method for the Reynolds averaged equations", AIAA Paper 88-0414, 1988.
11. Y.P Marx, "Time integration for the unsteady incompressible Navier-Stokes equations", *J. Comp. Phys.*, 112:182-209, 1994.
12. N.D. Melson, M.D. Sanetrik, H.L. Atkins, Time-accurate calculations with multigrid acceleration, in: N.D. Melson, T.A. Manteuffel, S.F. McCormick. (eds.), "Proceedings of the Sixth Copper Mountain conference on multigrid methods", 1993.
13. R. Mittal and S. Balachander, "Effect of three-dimensionality on the lift and drag of nominally two-dimensional cylinders", *Phys. Fluids*, 7:1841-1865, 1995.
14. A.Roshko, "The flow past a circular cylinder at low speeds", NACA Report TN 2913, 1953.
15. P.R. Spalart and S.R. Allmaras, "A one-equation turbulence model for aerodynamic flows", AIAA Paper 92-439, Reno, NV.

16. R.C. Swanson and E. Turkel, "A multistage time-stepping scheme for the Navier-Stokes equations", AIM Paper 85-0035, 1985.
17. Tang and N. Aubry, "On the symmetry breaking instability leading to vortex shedding", Phys. Fluids,9: 2550-2561,1997.
18. V.N. Vatsa and B. W. Wedan, "Development of a multigrid code for 3-d Navier-Stokes equations and its application to a grid-refinement study", Computers and Fluids. 18:391-403. 1990.

Appendix

The Butcher variables A , \hat{B} , and C for the ES-
DIRK4 scheme are (with $a_{6,j} = b_j$)

a_{21}	a_{22}	a_{31}	a_{32}
a_{33}	a_{41}	a_{42}	a_{43}
a_{44}	a_{51}	a_{52}	a_{53}
a_{54}	a_{55}	a_{61}	a_{62}
a_{63}	a_{64}	a_{65}	a_{66}
\hat{b}_1	\hat{b}_2	\hat{b}_3	\hat{b}_4
\hat{b}_5	\hat{b}_6	c_2	c_3
c_4	c_5	c_6	

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