

DUAL TIME STEPPING WITH MULTI-STAGE SCHEMES IN PHYSICAL TIME FOR UNSTEADY LOW MACH NUMBER COMPRESSIBLE FLOWS

Leonardo Santos de Brito Alves, leonardo_alves@ime.eb.br

Laboratório de Motores e Propulsão - LMP
Departamento de Engenharia Mecânica e de Materiais - SE/4,
Instituto Militar de Engenharia - IME,
Praça General Tibúrcio nº80, Praia Vermelha, Rio de Janeiro, RJ 22290-270, Brazil

Abstract. All unsteady simulations of low Mach number ($M \ll 0.3$) compressible flows performed with preconditioned density-based methods utilize multi-step schemes in physical-time. This restricts physical-time accurate simulations of even moderately stiff problems to second-order at most because higher order versions of these schemes are conditionally stable. The present paper describes how the traditional dual time stepping formulation can be used to introduce a preconditioned pseudo-time derivative into the intermediate stages of implicit Runge-Kutta (IRK) methods. In doing so, all the well known preconditioning techniques for low speed flows become available to multi-stage schemes in physical-time as well. Hence, higher order simulations with stronger numerical stability become possible. Furthermore, this comes at essentially no additional cost because IRK methods already require iterative solution techniques. A series of tests are performed to demonstrate the capabilities of this novel formulation. A-stable and strongly S-stable schemes with second, third and fourth-orders of accuracy are used to simulate periodic entropy perturbations in a flow field with Mach numbers ranging from 10^{-1} to 10^{-5} . They outperform their multi-step competitors when error tolerances imposed are smaller than required by engineering applications (approximately 1 digit accurate).

Keywords: Unsteady compressible flows, Low Mach numbers, Preconditioned density-based methods, Dual time stepping, Implicit Runge-Kutta schemes

1. INTRODUCTION

The ever increasing computer power reduced total calculation times and memory costs to such an extent that numerical simulations of unsteady three-dimensional equations are now feasible. Selecting a time-marching scheme for unsteady simulations is a compromise between efficiency and accuracy. However, stiffness plays a major role in this decision as well. It is caused by disparities in characteristic time and/or length scales, usually due to differences between convective and acoustic velocities, near-wall grid refinement and chemical reactions, among others. As a result, time-steps dictated by numerical stability are much smaller than required by accuracy considerations, drastically increasing computer time for conditionally stable time-marching schemes. Simulations with realistic computational costs are only possible in these cases with numerical schemes that are at least A-stable.

Many techniques have already been developed to deal this issue and improve solution accuracy and convergence towards steady-state [Blazek, 2006]. However, most of them modify the original transient evolution of the governing equations. A popular way of removing this restriction is known as dual time stepping (or DTS). It creates a pseudo-time within which the above acceleration techniques can be used without affecting the original physical-time. This is done by adding to the governing equations a pseudo-time derivative that emulates the original physical-time derivative. A physical-time accurate solution is then generated upon convergence towards pseudo-time steady-state in each physical-time step.

This concept of an additional time-like independent variable appears to have been first introduced to allow simulations of the unsteady equations of motion for incompressible flows [Merkle and Athavale, 1987]. They used the artificial compressibility method [Chorin, 1967] to render the governing equations fully hyperbolic and, hence, suitable for all time-marching algorithms already well-developed for the compressible flow equations. A generalized Crank-Nicholson ($CN\theta$) scheme was employed in physical-time with either first ($\theta = 1.0$) or second ($\theta = 0.5$) accuracy order. This time-like variable idea was independently introduced for the unsteady simulations of compressible flows as well [Rai, 1987], also motivated by the Newton iterative procedure commonly used at the time to speedup convergence towards steady-state [Rai and Chakravarthy, 1986]. Backwards differentiation formulas with first (BDF1) and second (BDF2) accuracy orders in physical-time were utilized.

Preconditioned density-based methods [Tukel, 1993] are a popular way of minimizing stiffness [Tukel, 1999], and their origins can be traced back to the artificial compressibility method [Chorin, 1967]. They were developed to deal with the convective/acoustic stiffness [Tukel, 1987] and pressure round-off errors accumulation [Merkle and Choi, 1988] present in low Mach number compressible flow equations. Nevertheless, these methods have been optimized to reduce stiffness caused by stretched meshes [Buelow et al., 1994] and chemical reactions as well [Venkateswaran et al., 1995]. Early on its development, time-derivative preconditioning was implemented with DTS to allow simulations of unsteady compressible flows at low Mach numbers [Hosangadi et al., 1990]. Another popular convergence acceleration technique that alters time evolution is known as multigrid [Wesseling, 1991]. It was utilized with DTS for the first time to allow

simulations of unsteady compressible flows at high Mach numbers [Jameson, 1991], as drivers for implicit time-marching schemes [Rai and Chakravarthy, 1986] in pseudo-time. Time-derivative preconditioning [Shuen et al., 1993] and multigrid [Rango and Zingg, 1996] have been used extensively since then, even simultaneously [Dailey and Pletcher, 1996, Liu et al., 1998, Battaglia et al., 1998, Shieh and Morris, 1998].

A common feature of all previously mentioned work is the use of multi-step schemes for physical-time marching [Alves, 2009]. Furthermore, these studies have been limited to second-order accuracy at most [Suh et al., 2006]. The reason for this is simple: there are no A-stable multi-step schemes with accuracy orders greater than two [Dahlquist, 1963]. Only a few higher order versions are available, but they are all conditionally stable [Lomax et al., 2001]. Despite the claim that bigger stencils can be used when higher orders are required [Shuen et al., 1993], practical experience with multi-step finite-difference equations indicates that BDF3 has little numerical stability and is often divergent for convection dominated problems, whereas BDF4 is rarely stable at all [Bijl et al., 2002]. This is the reason why implicit Runge-Kutta (IRK) methods became a popular alternative [Butcher, 2008]. Their high accuracy-orders with improved numerical stability properties are achieved by increasing the number of intermediate stages used. Furthermore, their self-starting nature makes it straightforward to implement variable time-step sizes, which is not the case with multi-step schemes. Today, many different variations of IRK methods have been employed with success to simulate high speed compressible flows with high temporal resolution [Zhong, 1996, Ascher et al., 1997, Kennedy and Carpenter, 2003, Yoh and Zhong, 2004a,b, Pareschi and Russo, 2005, Ferracina and Spijker, 2008].

Recently, DTS was modified to allow the use of multi-stage schemes in physical-time for unsteady simulations of high speed compressible flows [Bijl et al., 2002]. Multigrid and implicit residual smoothing were employed to drive iterative procedures, where inviscid and viscous fluxes were treated explicitly, for the intermediate stage equations of explicit singly-diagonally-implicit Runge-Kutta (ESDIRK) schemes. It was shown that higher order methods are more efficient than second-order methods when error tolerances are smaller than required by engineering simulations (one or two digits of accuracy). In fact, order yielding minimal computational costs was shown to increase as tolerance decreases. A few similar studies have followed [Jothiprasad et al., 2003, Bijl and Carpenter, 2005, Wang and Mavriplis, 2007], focussing on different multigrid strategies to increase efficiency. The main goal of the present paper is to extend this formulation with multi-stage schemes in physical-time towards low Mach number flows as well, using all the knowledge gained over the past two and a half decades about preconditioned density-based methods with multi-step schemes in physical-time. There are many applications for high-order time-marching methods. They are important in receptivity as well as thermal and hydrodynamic stability studies, where the temporal evolution of small perturbations towards nonlinear waves must be accurately simulated. Only then a correct transition to turbulence will be observed. Another example is aeroacoustics, where sound is generated by both large and small coherent structures present in turbulent flows. However, acoustic waves are generally much smaller in amplitude than vorticity waves, creating the need for such high precision techniques.

2. LOW MACH PRECONDITIONED DUAL TIME STEPPING

Consider the compressible flow equations from fluid dynamics, written as

$$\frac{\partial \mathbf{Q}}{\partial t} = \mathbf{f}(\mathbf{Q}) , \quad (1)$$

where t is the physical-time variable, \mathbf{Q} is the standard conservative variable vector and $\mathbf{f}(\mathbf{Q})$ is the steady-state residue, which is problem dependent. In order to generate discrete solutions from these conservation equations at low Mach numbers, two different techniques must be employed. First it is important to solve for pressure instead of density, since the latter is independent of the former at low Mach numbers. Furthermore, pressure should be decomposed into a sum of its hydrodynamic ($P_H \sim \rho u^2$) and thermodynamic ($P_T \sim \rho c^2$) contributions, since the latter is essentially constant and orders of magnitude higher than the former in this same limit. Hence, equation (1) is now written as

$$\mathbf{T} \frac{\partial \hat{\mathbf{Q}}}{\partial t} = \mathbf{f}(\mathbf{Q}) , \quad (2)$$

where $\mathbf{T} = \partial \mathbf{Q} / \partial \hat{\mathbf{Q}}$ and $\hat{\mathbf{Q}}$ is an arbitrary primitive variable vector that includes the hydrodynamic pressure, velocity components and, usually but not limited to, temperature. Nevertheless, equation (2) becomes increasingly stiff as the Mach number decreases below $M \sim 0.3$. Here is where the second technique, preconditioning, must be employed. This is achieved by replacing conservative to primitive variable Jacobian \mathbf{T} by preconditioning matrix $\mathbf{\Gamma}$, leading to

$$\mathbf{\Gamma} \frac{\partial \hat{\mathbf{Q}}}{\partial \tau} = \mathbf{f}(\mathbf{Q}) , \quad (3)$$

where τ is the pseudo-time variable. The preconditioning matrix $\mathbf{\Gamma}$ is in fact a variable Jacobian matrix \mathbf{T} modified in such a way to artificially force all eigenvalues of equation (2) to be of the same magnitude. However, this procedure alters the original physical-time t , creating a pseudo-time τ in which the flow evolves. This procedure is applied to accelerate

convergence of the original system, in the form of either (1) or (2), towards steady-state. At this point, the same solution $\mathbf{f}(\mathbf{Q}) = 0$ is recovered from preconditioned system (3) as well.

Physical-time accuracy is still possible for preconditioned density-based methods, represented by equation (3), but only if the original physical-time derivative is re-introduced into this equation, yielding

$$\Gamma \frac{\partial \hat{\mathbf{Q}}}{\partial \tau} + \frac{\partial \mathbf{Q}}{\partial t} = \mathbf{f}(\mathbf{Q}) , \quad (4)$$

a procedure known as dual time stepping, or DTS. Whenever pseudo-time steady state is reached within each physical-time step, the original governing equation (1) is recovered since $\partial \hat{\mathbf{Q}} / \partial \tau \simeq 0$ at this point.

2.1 Multi-step schemes in physical-time

The traditional approach to solving equation (4) treats the physical-time derivative as a source term, incorporating it into $\mathbf{f}(\mathbf{Q})$. Marching such a problem in pseudo-time using the implicit Euler method leads to

$$\Gamma^p \frac{\hat{\mathbf{Q}}^{p+1} - \hat{\mathbf{Q}}^p}{\Delta \tau} \simeq \left(\mathbf{f}(\mathbf{Q}) - \frac{\partial \mathbf{Q}}{\partial t} \right)^{p+1} , \quad (5)$$

where p and $p + 1$ represent the known and unknown pseudo-time steps, respectively.

2.1.1 Crank-Nicholson scheme: two-point & second-order & A-stable

Probably the first approach employed to march equation (5) in physical-time was the second-order Crank-Nicholson scheme, known as CN2. This multi-step scheme is A-stable, which means its gain is always bounded even for high CFL numbers when applied to a linear and homogeneous ODE. Furthermore, it is also non-dissipative, an attractive property for turbulent simulations. In this scenario, equation (5) becomes

$$\Gamma^p \frac{\hat{\mathbf{Q}}^{p+1} - \hat{\mathbf{Q}}^p}{\Delta \tau} \simeq \frac{\mathbf{f}(\mathbf{Q}^{p+1}) + \mathbf{f}(\mathbf{Q}^n)}{2} - \frac{\mathbf{Q}^{p+1} - \mathbf{Q}^n}{\Delta t} , \quad (6)$$

where dependent variables at the latest physical ($n + 1$) and pseudo ($p + 1$) time steps were set to be equal, a necessary approximation whenever DTS with multi-step schemes in physical-time is employed [Tukel and Vatsa, 2005]. Finally, linearization of equation (6) in pseudo-time leads to

$$\left\{ \frac{\Gamma}{\Delta \tau} + \frac{\mathbf{T}}{\Delta t} - \frac{1}{2} \frac{\partial \mathbf{f}}{\partial \hat{\mathbf{Q}}} \right\}^p \Delta \hat{\mathbf{Q}} \simeq \frac{\mathbf{f}(\mathbf{Q}^p) + \mathbf{f}(\mathbf{Q}^n)}{2} - \frac{\mathbf{Q}^p - \mathbf{Q}^n}{\Delta t} , \quad (7)$$

where $\Delta \hat{\mathbf{Q}} = \hat{\mathbf{Q}}^{p+1} - \hat{\mathbf{Q}}^p$ is the solution increment in pseudo-time, $\mathbf{T} = \partial \mathbf{Q} / \partial \hat{\mathbf{Q}}$ is the conservative to primitive variable Jacobian and $\partial \mathbf{f} / \partial \hat{\mathbf{Q}}$ contains the remaining Jacobians with respect to the arbitrary primitive variable vector.

2.1.2 Backwards differentiation formula: three-point & second-order & L-stable

Arguably the most used multi-step method to march equation (5) in physical-time is the second-order backwards differentiation formula known as BDF2. This is so because of its L-stability, a property equivalent to A-stability with the additional advantage of having zero gain in the limit of very large CFL numbers. In this case, equation (5) becomes

$$\Gamma^p \frac{\hat{\mathbf{Q}}^{p+1} - \hat{\mathbf{Q}}^p}{\Delta \tau} \simeq \mathbf{f}(\mathbf{Q}^{p+1}) - \frac{3 \mathbf{Q}^{p+1} - 4 \mathbf{Q}^n + \mathbf{Q}^{n-1}}{2 \Delta t} , \quad (8)$$

which, after the same steps applied previously, yields

$$\left\{ \frac{\Gamma}{\Delta \tau} + 3 \frac{\mathbf{T}}{\Delta t} - \frac{\partial \mathbf{f}}{\partial \hat{\mathbf{Q}}} \right\}^p \Delta \hat{\mathbf{Q}} \simeq \mathbf{f}(\mathbf{Q}^p) - \frac{3 \mathbf{Q}^p - 4 \mathbf{Q}^n + \mathbf{Q}^{n-1}}{2 \Delta t} . \quad (9)$$

2.2 Multi-stage schemes in physical-time

As mentioned previously, unsteady simulations using preconditioned density-based methods published in the available literature always employ multi-step schemes in physical-time, usually one of the two types presented above. Higher order versions of these schemes are not successful because of their poor numerical stability, rendering them not robust enough for practical applications. Implicit Runge-Kutta schemes represent a viable alternative, since they are able to achieve high orders and strong numerical stability simultaneously. Although successfully used for unsteady simulations of high speed compressible flows, they have yet to be employed for their low Mach number counterparts. In order to do so, the

preconditioned pseudo-time derivative in equation (4) is treated as a source term instead, and incorporated into $\mathbf{f}(\mathbf{Q})$. Marching this equation in physical-time with a general Runge-Kutta method yields

$$\mathbf{Q}^{n+1} = \mathbf{Q}^n + \Delta t \sum_{i=1}^k \omega_i \left(\mathbf{f}(\mathbf{k}_i) - \mathbf{\Gamma}_i \frac{\partial \hat{\mathbf{k}}_i}{\partial \tau} \right), \quad (10)$$

where k is the number of intermediate stages and ω_i their respective time advancement weights. The stage dependent variables in primitive and conservative forms $\hat{\mathbf{k}}_i$ and \mathbf{k}_i , respectively, are obtained from equations

$$\mathbf{k}_i = \mathbf{Q}^n + \Delta t \sum_{j=1}^k \beta_{i,j} \left(\mathbf{f}(\mathbf{k}_j) - \mathbf{\Gamma}_j \frac{\partial \hat{\mathbf{k}}_j}{\partial \tau} \right) \quad \text{for } 1 \leq i \leq k, \quad (11)$$

where $\beta_{i,j}$ are additional parameters that, alongside ω_i , satisfy predetermined order and stability requirements. Collecting the preconditioned pseudo-time derivative in equation (11), marching them forward with the same implicit Euler method used previously and linearizing the end result in pseudo-time accordingly leads to

$$\left\{ \frac{\mathbf{\Gamma}_i}{\Delta \tau} - \frac{\partial \mathbf{f}_i}{\partial \hat{\mathbf{k}}_i} \right\}^p \Delta \hat{\mathbf{k}}_i + \sum_{j=1}^k \bar{\beta}_{i,j} \frac{\mathbf{T}_j^p}{\Delta t} \Delta \hat{\mathbf{k}}_j \simeq \mathbf{f}(\mathbf{k}_i^p) + \sum_{j=1}^k \bar{\beta}_{i,j} \left(\frac{\mathbf{Q}^n - \mathbf{k}_j^p}{\Delta t} \right) \quad \text{for } 1 \leq i \leq k, \quad (12)$$

where $\Delta \hat{\mathbf{k}}_i = \hat{\mathbf{k}}_i^{p+1} - \hat{\mathbf{k}}_i^p$, $\mathbf{T}_j = \partial \mathbf{k}_j / \partial \hat{\mathbf{k}}_j$, $\partial \mathbf{f}_i / \partial \hat{\mathbf{k}}_i$ contains the remaining Jacobians and $\bar{\beta}$ can be obtained *a priori* from β with simple algebra. Linearization employed in the development of this formula considered the general case of a fully implicit Runge-Kutta scheme. At any given stage in diagonally implicit schemes, previous intermediate stage dependent variables don't need to be linearized in pseudo-time. In such cases this equation can be re-written as

$$\left\{ \frac{\mathbf{\Gamma}_i}{\Delta \tau} + \bar{\beta}_{i,i} \frac{\mathbf{T}_i}{\Delta t} - \frac{\partial \mathbf{f}_i}{\partial \hat{\mathbf{k}}_i} \right\}^p \Delta \hat{\mathbf{k}}_i \simeq \mathbf{f}(\mathbf{k}_i^p) + \sum_{j=1}^{i-1} \bar{\beta}_{i,j} \left(\frac{\mathbf{Q}^n - \mathbf{k}_j^{p+1}}{\Delta t} \right) + \bar{\beta}_{i,i} \frac{\mathbf{Q}^n - \mathbf{k}_i^p}{\Delta t} \quad \text{for } 1 \leq i \leq k. \quad (13)$$

Once pseudo-time steady state is reached in system of equations (12) or (13), one may use

$$\mathbf{Q}^{n+1} \simeq \mathbf{Q}^n + \Delta t \sum_{i=1}^k \omega_i \mathbf{f}(\mathbf{k}_i), \quad (14)$$

instead since $\partial \hat{\mathbf{k}}_i / \partial \tau \simeq 0$ in equation (10) at this point. Arbitrary Runge-Kutta methods may be employed using the above formulation, but we will focus only on singly diagonally implicit schemes in the present paper [Butcher, 2008].

2.2.1 IRK2-AS scheme: one-stage & second-order & A-stable

The preconditioned version (PIRK2-AS) of this scheme may be written from equation (13) as

$$\left\{ \frac{\mathbf{\Gamma}_1}{\Delta \tau} + 2 \frac{\mathbf{T}_1}{\Delta t} - \frac{\partial \mathbf{f}_1}{\partial \hat{\mathbf{k}}_1} \right\}^p \Delta \hat{\mathbf{k}}_1 = \mathbf{f}(\mathbf{k}_1^p) + 2 \frac{\mathbf{Q}^n - \mathbf{k}_1^p}{\Delta t}, \quad (15)$$

and uses equation (14) to update its solution in physical-time with

$$\mathbf{Q}^{n+1} \simeq \mathbf{Q}^n + \Delta t \mathbf{f}(\mathbf{k}_1). \quad (16)$$

2.2.2 SDIRK3-AS scheme: two-stages & third-order & A-stable

The preconditioned version (PSDIRK3-AS) of this scheme may be written from equation (13) as

$$\left\{ \frac{\mathbf{\Gamma}_1}{\Delta \tau} + (3 - \sqrt{3}) \frac{\mathbf{T}_1}{\Delta t} - \frac{\partial \mathbf{f}_1}{\partial \hat{\mathbf{k}}_1} \right\}^p \Delta \hat{\mathbf{k}}_1 = \mathbf{f}(\mathbf{k}_1^p) + (3 - \sqrt{3}) \frac{\mathbf{Q}^n - \mathbf{k}_1^p}{\Delta t} \quad \text{and} \quad (17)$$

$$\left\{ \frac{\mathbf{\Gamma}_2}{\Delta \tau} + (3 - \sqrt{3}) \frac{\mathbf{T}_2}{\Delta t} - \frac{\partial \mathbf{f}_2}{\partial \hat{\mathbf{k}}_2} \right\}^p \Delta \hat{\mathbf{k}}_2 = \mathbf{f}(\mathbf{k}_2^p) + (3 - \sqrt{3}) \frac{\mathbf{Q}^n - \mathbf{k}_2^p}{\Delta t} - (6 - 4\sqrt{3}) \frac{\mathbf{Q}^n - \mathbf{k}_1^{p+1}}{\Delta t},$$

and uses equation (14) to update its solution in physical-time with

$$\mathbf{Q}^{n+1} \simeq \mathbf{Q}^n + \frac{\Delta t}{2} \left(\mathbf{f}(\mathbf{k}_1) + \mathbf{f}(\mathbf{k}_2) \right). \quad (18)$$

2.2.3 SDIRK4-AS scheme: three-stages & fourth-order & A-stable

The preconditioned version (PSDIRK4-AS) of this scheme may be written from (13) as

$$\begin{aligned} \left\{ \frac{\Gamma_1}{\Delta\tau} + \left(\frac{2}{1+\delta} \right) \frac{\mathbf{T}_1}{\Delta t} - \frac{\partial \mathbf{f}_1}{\partial \hat{\mathbf{k}}_1} \right\}^p \Delta \hat{\mathbf{k}}_1 &= \mathbf{f}(\mathbf{k}_1^p) + \left(\frac{2}{1+\delta} \right) \frac{\mathbf{Q}^n - \mathbf{k}_1^p}{\Delta t}, \\ \left\{ \frac{\Gamma_2}{\Delta\tau} + \left(\frac{2}{1+\delta} \right) \frac{\mathbf{T}_2}{\Delta t} - \frac{\partial \mathbf{f}_2}{\partial \hat{\mathbf{k}}_2} \right\}^p \Delta \hat{\mathbf{k}}_2 &= \mathbf{f}(\mathbf{k}_2^p) + \left(\frac{2}{1+\delta} \right) \frac{\mathbf{Q}^n - \mathbf{k}_2^p}{\Delta t} + \delta_1 \frac{\mathbf{Q}^n - \mathbf{k}_1^{p+1}}{\Delta t} \quad \text{and} \\ \left\{ \frac{\Gamma_3}{\Delta\tau} + \left(\frac{2}{1+\delta} \right) \frac{\mathbf{T}_3}{\Delta t} - \frac{\partial \mathbf{f}_3}{\partial \hat{\mathbf{k}}_3} \right\}^p \Delta \hat{\mathbf{k}}_3 &= \mathbf{f}(\mathbf{k}_3^p) + \left(\frac{2}{1+\delta} \right) \frac{\mathbf{Q}^n - \mathbf{k}_3^p}{\Delta t} + \delta_2 \frac{\mathbf{Q}^n - \mathbf{k}_2^{p+1}}{\Delta t} + \delta_3 \frac{\mathbf{Q}^n - \mathbf{k}_1^{p+1}}{\Delta t}, \end{aligned} \quad (19)$$

where $\delta = 2 \cos[\pi/18]/\sqrt{3}$, $\delta_1 = 2\delta/(1+\delta)^2$, $\delta_2 = (4+8\delta)/(1+\delta)^2$ and $\delta_3 = 4(\delta^2 - \delta - 1)/(1+\delta)^3$, and uses equation (14) to update its solution in physical-time with

$$\mathbf{Q}^{n+1} \simeq \mathbf{Q}^n + \frac{\Delta t}{6\delta^2} \left(\mathbf{f}(\mathbf{k}_1) + (6\delta^2 - 2) \mathbf{f}(\mathbf{k}_2) + \mathbf{f}(\mathbf{k}_3) \right). \quad (20)$$

2.2.4 SDIRK2-SSS: two-stages & second-order & strongly S-stable

The preconditioned version (PSDIRK2-SSS) of this scheme may be written from (13) as

$$\begin{aligned} \left\{ \frac{\Gamma_1}{\Delta\tau} + \frac{1}{\delta} \frac{\mathbf{T}_1}{\Delta t} - \frac{\partial \mathbf{f}_1}{\partial \hat{\mathbf{k}}_1} \right\}^p \Delta \hat{\mathbf{k}}_1 &= \mathbf{f}(\mathbf{k}_1^p) + \frac{\mathbf{Q}^n - \mathbf{k}_1^p}{\delta \Delta t} \quad \text{and} \\ \left\{ \frac{\Gamma_2}{\Delta\tau} + \frac{1}{\delta} \frac{\mathbf{T}_2}{\Delta t} - \frac{\partial \mathbf{f}_2}{\partial \hat{\mathbf{k}}_2} \right\}^p \Delta \hat{\mathbf{k}}_2 &= \mathbf{f}(\mathbf{k}_2^p) + \frac{\mathbf{Q}^n - \mathbf{k}_2^p}{\delta \Delta t} + \left(1 - \frac{1}{\delta} \right) \frac{\mathbf{Q}^n - \mathbf{k}_1^{p+1}}{\delta \Delta t}, \end{aligned} \quad (21)$$

where $\delta = (2 - \sqrt{2})/2$, and uses equation (14) to update its solution in physical-time with

$$\mathbf{Q}^{n+1} \simeq \mathbf{Q}^n + \Delta t \left((1 - \delta) \mathbf{f}(\mathbf{k}_1) + \delta \mathbf{f}(\mathbf{k}_2) \right) \simeq \mathbf{k}_2. \quad (22)$$

2.2.5 SDIRK3-SSS scheme: three-stages & third-order & strongly S-stable

The preconditioned version (PSDIRK3-SSS) of this scheme may be written from (13) as

$$\begin{aligned} \left\{ \frac{\Gamma_1}{\Delta\tau} + \frac{1}{\delta} \frac{\mathbf{T}_1}{\Delta t} - \frac{\partial \mathbf{f}_1}{\partial \hat{\mathbf{k}}_1} \right\}^p \Delta \hat{\mathbf{k}}_1 &= \mathbf{f}(\mathbf{k}_1^p) + \frac{\mathbf{Q}^n - \mathbf{k}_1^p}{\delta \Delta t}, \\ \left\{ \frac{\Gamma_2}{\Delta\tau} + \frac{1}{\delta} \frac{\mathbf{T}_2}{\Delta t} - \frac{\partial \mathbf{f}_2}{\partial \hat{\mathbf{k}}_2} \right\}^p \Delta \hat{\mathbf{k}}_2 &= \mathbf{f}(\mathbf{k}_2^p) + \frac{\mathbf{Q}^n - \mathbf{k}_2^p}{\delta \Delta t} + \left(\frac{\delta - 1}{2\delta^2} \right) \frac{\mathbf{Q}^n - \mathbf{k}_1^{p+1}}{\Delta t} \quad \text{and} \\ \left\{ \frac{\Gamma_3}{\Delta\tau} + \frac{1}{\delta} \frac{\mathbf{T}_3}{\Delta t} - \frac{\partial \mathbf{f}_3}{\partial \hat{\mathbf{k}}_3} \right\}^p \Delta \hat{\mathbf{k}}_3 &= \mathbf{f}(\mathbf{k}_3^p) + \frac{\mathbf{Q}^n - \mathbf{k}_3^p}{\delta \Delta t} - \frac{\delta_2}{\delta^2} \frac{\mathbf{Q}^n - \mathbf{k}_2^{p+1}}{\Delta t} + \left(\frac{\delta_2 - \delta(2\delta_1 + \delta_2)}{2\delta^3} \right) \frac{\mathbf{Q}^n - \mathbf{k}_1^{p+1}}{\Delta t}, \end{aligned} \quad (23)$$

where $\delta_1 = -(1 - 16\delta + 6\delta^2)/4$, $\delta_2 = (5 - 20\delta + 6\delta^2)/4$ and $\delta \simeq 0.435866521508458999 \pm 10^{-18}$, and uses equation (14) to update its solution in physical-time with

$$\mathbf{Q}^{n+1} \simeq \mathbf{Q}^n + \Delta t \left(\delta_1 \mathbf{f}(\mathbf{k}_1) + \delta_2 \mathbf{f}(\mathbf{k}_2) + \delta \mathbf{f}(\mathbf{k}_3) \right) \simeq \mathbf{k}_3. \quad (24)$$

2.2.6 SDIRK4-SSS scheme: five-stages & fourth-order & strongly S-stable

Both four-stage fourth-order strongly S-stable and fifth-order A-stable SDIRK methods do not exist [Alexander, 1977]. An additional stage is necessary to generate either method. The preconditioned version (PSDIRK4-SSS) of this scheme may be written from (13) as

$$\begin{aligned} \left\{ \frac{\Gamma_1}{\Delta\tau} + \frac{1}{\alpha} \frac{\mathbf{T}_1}{\Delta t} - \frac{\partial \mathbf{f}_1}{\partial \hat{\mathbf{k}}_1} \right\}^p \Delta \hat{\mathbf{k}}_1 &= \mathbf{f}(\mathbf{k}_1^p) + \frac{\mathbf{Q}^n - \mathbf{k}_1^p}{\alpha \Delta t}, \\ \left\{ \frac{\Gamma_2}{\Delta\tau} + \frac{1}{\alpha} \frac{\mathbf{T}_2}{\Delta t} - \frac{\partial \mathbf{f}_2}{\partial \hat{\mathbf{k}}_2} \right\}^p \Delta \hat{\mathbf{k}}_2 &= \mathbf{f}(\mathbf{k}_2^p) + \frac{\mathbf{Q}^n - \mathbf{k}_2^p}{\alpha \Delta t} + c_{2,1} \frac{\mathbf{Q}^n - \mathbf{k}_1^{p+1}}{\alpha \Delta t}, \end{aligned}$$

$$\left\{ \frac{\Gamma_3}{\Delta\tau} + \frac{1}{\alpha} \frac{\mathbf{T}_3}{\Delta t} - \frac{\partial \mathbf{f}_3}{\partial \hat{\mathbf{k}}_3} \right\}^p \Delta \hat{\mathbf{k}}_3 = \mathbf{f}(\mathbf{k}_3^p) + \frac{\mathbf{Q}^n - \mathbf{k}_3^p}{\alpha \Delta t} + c_{3,2} \frac{\mathbf{Q}^n - \mathbf{k}_2^{p+1}}{\alpha \Delta t} + c_{3,1} \frac{\mathbf{Q}^n - \mathbf{k}_1^{p+1}}{\alpha \Delta t} , \quad (25)$$

$$\left\{ \frac{\Gamma_4}{\Delta\tau} + \frac{1}{\alpha} \frac{\mathbf{T}_4}{\Delta t} - \frac{\partial \mathbf{f}_4}{\partial \hat{\mathbf{k}}_4} \right\}^p \Delta \hat{\mathbf{k}}_4 = \mathbf{f}(\mathbf{k}_4^p) + \frac{\mathbf{Q}^n - \mathbf{k}_4^p}{\alpha \Delta t} + c_{4,3} \frac{\mathbf{Q}^n - \mathbf{k}_3^{p+1}}{\alpha \Delta t} + \dots + c_{4,1} \frac{\mathbf{Q}^n - \mathbf{k}_1^{p+1}}{\alpha \Delta t} \quad \text{and}$$

$$\left\{ \frac{\Gamma_5}{\Delta\tau} + \frac{1}{\alpha} \frac{\mathbf{T}_5}{\Delta t} - \frac{\partial \mathbf{f}_5}{\partial \hat{\mathbf{k}}_5} \right\}^p \Delta \hat{\mathbf{k}}_5 = \mathbf{f}(\mathbf{k}_5^p) + \frac{\mathbf{Q}^n - \mathbf{k}_5^p}{\alpha \Delta t} + c_{5,4} \frac{\mathbf{Q}^n - \mathbf{k}_4^{p+1}}{\alpha \Delta t} + \dots + c_{5,1} \frac{\mathbf{Q}^n - \mathbf{k}_1^{p+1}}{\alpha \Delta t} ,$$

and uses equation (14) to update its solution in physical-time with

$$\mathbf{Q}^{n+1} \simeq \mathbf{Q}^n + \Delta t \left(b_1 \mathbf{f}(\mathbf{k}_1) + b_2 \mathbf{f}(\mathbf{k}_2) + b_3 \mathbf{f}(\mathbf{k}_3) + b_4 \mathbf{f}(\mathbf{k}_4) + \alpha \mathbf{f}(\mathbf{k}_5) \right) \simeq \mathbf{k}_5 . \quad (26)$$

The $c_{i,j}$ coefficients are used to reduce the size of system (25), and are defined as

$$\begin{aligned} c_{2,1} &= -a_{2,1}/\alpha , \quad c_{3,1} = (a_{2,1} a_{3,2} - a_{3,1} \alpha)/\alpha^2 , \quad c_{3,2} = -a_{3,2}/\alpha , \quad c_{4,1} = -(a_{2,1} a_{3,2} a_{4,3} - (a_{2,1} a_{4,2} \\ &+ a_{3,1} a_{4,3}) \alpha + a_{4,1} \alpha^2)/\alpha^3 , \quad c_{4,2} = (a_{3,2} a_{4,3} - a_{4,2} \alpha)/\alpha^2 , \quad c_{4,3} = -a_{4,3}/\alpha , \quad c_{5,1} = (a_{2,1} a_{3,2} a_{4,3} b_4 \\ &- (a_{2,1} a_{3,2} b_3 + (a_{2,1} a_{4,2} + a_{3,1} a_{4,3}) b_4) \alpha + (a_{2,1} b_2 + a_{3,1} b_3 + a_{4,1} b_4) \alpha^2 - b_1 \alpha^3)/\alpha^4 , \end{aligned} \quad (27)$$

$$c_{5,2} = -(a_{3,2} a_{4,3} b_4 - (a_{3,2} b_3 + a_{4,2} b_4) \alpha + b_2 \alpha^2)/\alpha^3 , \quad c_{5,3} = (a_{4,3} b_4 - b_3 \alpha)/\alpha^2 \quad \text{and} \quad c_{5,4} = -b_4/\alpha ,$$

where the remaining coefficients are given by Alexander [1977].

3. GOVERNING EQUATIONS

Performance and accuracy of the methods described in the previous sections are going to be compared for the unsteady one-dimensional compressible Euler equations. Their physical-time steady-state residue can be written as

$$\mathbf{f}(\mathbf{Q}) = - \frac{\partial \mathbf{E}_i}{\partial x} , \quad (28)$$

where the inviscid flux \mathbf{E}_i is given by

$$\mathbf{E}_i = \{ \rho u , \rho u^2 + P_H , (\rho E + P) u \} . \quad (29)$$

The conservative \mathbf{Q} and primitive $\hat{\mathbf{Q}}$ variable vectors are

$$\mathbf{Q} = \{ \rho , \rho u , \rho E \} \quad \text{and} \quad \hat{\mathbf{Q}} = \{ P_H , u , T \} , \quad (30)$$

and ρ stands for density, u for velocity, $E = e + u^2/2$ for total internal energy per unit mass, e for thermal internal energy per unit mass, T for temperature, c for speed of sound and $P = P_H + P_T$ as pressure, with the hydrostatic contribution, associated with buoyancy effects, neglected in this study. This pressure splitting is used to minimize pressure round-off error propagation [Sesterhenn et al., 1999, Lee, 2007].

The preconditioning matrix Γ employed in this study is

$$\Gamma = \begin{pmatrix} \rho_P & 0 & \rho_T \\ u \rho_P & \rho & u \rho_T \\ H \rho_P - (1 - \rho \frac{\partial h}{\partial P}) & \rho u & H \rho_T + \rho \frac{\partial h}{\partial T} \end{pmatrix} , \quad (31)$$

where $H = h + u^2/2$ is the total enthalpy per unit mass and $h = e + P/\rho$ is the enthalpy per unit mass [Venkateswaran and Merkle, 2000]. This matrix has the same form of Jacobian \mathbf{T} , except for the density derivatives with respect to pressure and temperature, now replaced by

$$\rho_P = \frac{1}{c_p^2} - \rho_T \left(1 - \rho \frac{\partial h}{\partial P} \right) / \left(\rho \frac{\partial h}{\partial T} \right) \quad \text{and} \quad \rho_T = \delta \frac{\partial \rho}{\partial T} , \quad (32)$$

with c_p being defined as the pseudo-speed of sound and $\delta = 1$.

Although arbitrary equations of state can be utilized [Merkle et al., 1998], only thermally perfect gases are going to be considered in this study. Hence, one can define

$$P = \rho R T \quad \text{and} \quad h = h_f^0 + \int_{T_0}^T C_P(s) ds , \quad (33)$$

and also generate relations

$$\frac{\partial \rho}{\partial P} = \frac{1}{RT} , \quad \frac{\partial \rho}{\partial T} = -\frac{\rho}{T} , \quad \frac{\partial h}{\partial P} = 0 \quad \text{and} \quad \frac{\partial h}{\partial T} = C_P(T) , \quad (34)$$

where T_0 is a reference temperature, $C_P(T)$ is the specific heat at constant pressure, R is the specific gas constant and h_f^0 is the heat of formation per unit mass.

3.1 Model problem

These equations are used to simulate entropy waves at Mach numbers ranging from $M = 10^{-5}$ to 10^{-1} . They are generated with density perturbations at constant pressure and flow velocity. Respective temperature changes are obtained from equation of state (33). Perturbation theory indicates that compressible flows can sustain three distinct disturbance fields: entropy, vorticity and sound waves. They are independent whenever fluctuation intensities are small but interact otherwise. Hence, small density disturbances are simply convected downstream without any changes in amplitude, frequency or phase. If an arbitrary dimensionless function $f(x)$ is used as initial condition for density, its transient behavior is given by $\rho(x, t)/\rho_0 = f(x - u_0 t)$. The wave form

$$f(x) = 1 + \delta_0 \sin[2\pi x/l_0] , \quad \text{with } 0 \leq x \leq l_0 , \quad (35)$$

is chosen here in order to maintain mean values unaltered. In the description above, ρ_0 and u_0 are the average density and constant flow speed, and $\delta_0 = 1\%$, l_0 and $t_0 = l_0/u_0$ are the perturbation amplitude, wavelength and period, respectively. Reference pressure and temperature are given by $P_0 = 101325 Pa$ and $T_0 = 300^\circ K$, respectively.

3.2 Spatial Integration

As defined above, the selected model problem is periodic in both time and space. Spatial periodicity eliminates the need to use boundary closure schemes and, hence, avoids additional error sources that would otherwise contaminate the numerical solution. Two and three-dimensional problems were avoided for the same reason. Periodic boundary conditions are exactly enforced on the right-hand (or explicit) side of equations (7) or (9) and (12) or (13), but are approximated on their left-hand (or implicit) side to avoid a cyclic matrix structure.

In every case reported next, the number of uniformly spaced grid points per wavelength N_X was made large enough in order to guarantee that numerical resolution is dominated by temporal instead of spatial errors. High-order spatial resolution for the inviscid fluxes is achieved with a fifth-order version Schwer [1999] of the well-known flux-difference method with a preconditioned artificial dissipation matrix Buelow et al. [2001]. Finally, spatial derivatives present in the aforementioned implicit matrixes employ three-point stencils at most. This leads to a single block tri-diagonal system from equation (7) or (9), but to multiple ones from equation (13) for all DIRK methods. On the other hand, FIRK methods with one, two and three-stages solve one single tri, penta and hepta-diagonal system from equation (12), respectively.

3.3 Physical-Time Integration

Temporally periodic and inviscid problems are ideal choices when evaluating the temporal resolution of methods designed for unsteady flows Wang and Mavriplis [2007]. They provide a straightforward way to quantify dissipation and dispersion errors associated with long integration times of each marching scheme, since the model problem chosen has a known period and no physical dissipation. In the present case, density values at a fixed spatial location, here chosen to be $x/l_0 = 0$, are tracked in physical-time. Peak to peak changes of the expected maximum or minimum values after each period indicate amplitude variation, which is a measure of dissipative (or anti-dissipative) errors. Similarly, deviations from the expected periodic location of each one of these peaks indicate phase variation, which is a measure of dispersive (either lagging or leading) errors. Time-steps used to march the solution in physical-time are defined as $\Delta t = t_0/N_T$, where N_T is the number of points per period of oscillation. One hundred periods were simulated in each run with $N_T = 4, 8, 16, 32, 64$ and 128 .

The PIRK methodology described previously is applied to an extensive list of IRK schemes available in the literature, in order to demonstrate its capability to generate at least *i*) unconditionally-stable numerical schemes capable of simulating unsteady compressible flows at low-speeds with *ii*) arbitrary high-order temporal resolution. The original Butcher tableaus of the selected methods and their respective preconditioned system of equations are given in the appendix. Second, third and fourth-order accurate SDIRK schemes that are either A-stable or strongly S-stable were chosen. FIRK schemes were also tested but are not competitive with or without preconditioning and, hence, are not shown in this study. Temporal integration using multi-step schemes BDF and AM described previously were also performed. Density-based methods using both multi-stage and multi-step schemes will have the capital letter P added when preconditioned.

It is important to note that any other multi-stage scheme; such as DIRK, ESDIRK, IMEX-RK, SSP-RK, and so on; could have been used instead. Hence, storage requirements for the particular PIRK method constructed are the same ones already required by its respective IRK scheme plus the same number of additional vector storages required by unsteady preconditioning using multi-step schemes. This additional number is usually one, for pressure, but can be as high as five, which includes pressure, temperature and all three velocity components in a three-dimensional simulation.

3.4 Pseudo-Time Integration

Preconditioning has been shown time and again to improve convergence towards physical-time steady-state of density-based methods for low speed flows Turkel [1993, 1999]. The same is true when multi-stage schemes are used instead for

physical-time marching. In order to demonstrate this point, the original IRK schemes selected (without preconditioning) are also utilized to solve the model problem discussed above. This is achieved by changing the preconditioning matrix $\mathbf{\Gamma}$ back to the variables Jacobian \mathbf{T} . One can still refer to τ as pseudo-time in such case because the explicit and implicit sides of this equation utilize finite-difference approximations with different spatial accuracy orders Rai [1987].

Pseudo-time iterations per physical-time step were performed until maxima pseudo-time increment (either $\Delta\hat{\mathbf{Q}}$ or $\Delta\hat{\mathbf{k}}$) and residue (either $\tilde{\mathbf{f}}(\mathbf{Q})$ or $\tilde{\mathbf{f}}(\mathbf{k})$) based on the velocity field's L_∞ -norm were below $M \times 10^{-7}$, which is close to machine precision for all Mach numbers simulated. Average values within a given period may have a considerable standard deviation, but change very little over one hundred periods. The exception is due to the presence of large dissipative errors, which force convergence to an artificial physical-time steady-state and reduce the number of iterations per period in the process. Nevertheless, this iteration count averaged over all one hundred periods is named \bar{m} and utilized as a global measure of convergence. Separate iteration counts \bar{m}_i for each stage i are performed when multiple stages are used, leading to $\bar{m} = \sum \bar{m}_i$. Respective optimal CFL numbers are obtained with a procedure similar to Brent's method Press et al. [1992]: First, CFL is gradually increased in uniform steps until a minimum in the number of pseudo-time iterations is first observed. Preliminary tests have shown that the global minimum is very close to this first local minimum, when they are not one and the same, limiting errors to approximately less than 10%. Once this minimum is bracketed, inverse quadratic interpolation is employed for fast convergence towards its value. When a particular CFL number within bounds generates an iteration count greater than the one at both bounds, bisection is used instead to guarantee convergence towards the minimum. This procedure was stopped when the average quadratic difference between bounds was below 5×10^{-3} . Lower tolerance values created convergence problems because of the oscillatory nature of the iteration count.

4. RESULTS AND DISCUSSION

An in depth analysis of PIRK methods not shown here demonstrate that both accuracy and order of Runge-Kutta schemes are machine precision equal with and without preconditioning. Hence, detailed results about dissipation and dispersion errors as well as order reduction are not needed here. Furthermore, PCN2-AS and PIRK2-AS generate machine precision identical results, as is the case when preconditioning is not utilized. Finally, unsteady simulations with third-order multi-step schemes PAM3 and PBDF3 in physical-time were also attempted, but without success. The former is too numerically unstable and is not able to generate results past the first 6 periods of oscillation for any Mach number. Although the latter is more stable, it could only simulate all 100 periods when $N_T \geq 256$. Nevertheless, it did so for all Mach numbers. This numerical stability problem renders PBDF3 too inefficient and its results will not be shown here.

4.1 Accuracy orders

Density absolute error per physical-time step, both in dimensionless form, is shown in Figure 1 for all schemes tested for physical-time marching with $M = 10^{-5}$ at $t/t_0 = 1$ and 25. It is clear that, after one period of oscillation, all schemes still have their respective designed accuracy. As mentioned previously, BDF3 and AM3 did generate results up to $t/t_0 = 1$. However, they were not numerically stable enough to generate results up to $t/t_0 = 25$. For this reason they are not presented in either figure. Furthermore, at this later physical-time, PIRK schemes with a given number of stages and stronger numerical stability are better able to sustain their order and maintain higher accuracy than their respective schemes with weaker numerical stability. The same reasoning is not true about BDF2 and CN2 schemes, but that is likely due to the latter's non-dissipative nature.

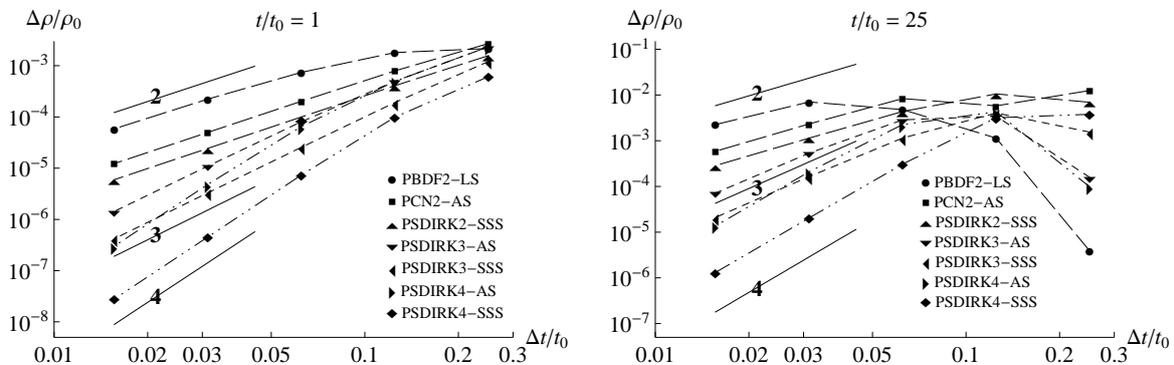


Figure 1. Dimensionless density's absolute error per dimensionless physical-time step generated with $M = 10^{-5}$ at $t/t_0 = 1$ and 25 for all preconditioned density-based methods tested.

4.2 Numerical stability

The number of pseudo-time iterations per dimensionless physical-time step, named here \bar{m} , is shown in Figure 2 for PSDIRK2-SSS, SDIRK2-SSS, PSDIRK3-AS and SDIRK3-AS as well as all Mach numbers simulated. This parameter is averaged and optimized as explained before. Discussion is restricted to these schemes, but the same qualitative behavior is observed for all other schemes tested. The efficiency of PSDIRK2-SSS, inversely proportional to \bar{m} , is upper bounded by PBD2 and lower bounded by PCN2, making it competitive with these traditional multi-step schemes even though it solves a preconditioned system containing twice as many equations. PSDIRK3-AS is not as efficient, but it is as competitive as PCN2 and more accurate for short term simulations. As expected, non-preconditioned versions of these schemes have much higher computational costs. SDIRK2-SSS requires 1.35 (with $N_T = 128$) to 4.11 (with $N_T = 4$) times as many pseudo-time iterations per physical-time step as PSDIRK2-SSS when $M = 10^{-1}$. These \bar{m} ratios go to 2.17 and 49.1, respectively, when the Mach number is reduced by one order of magnitude. A similar trend is observed between SDIRK3-AS and PSDIRK3-AS. The former requires approximately 1.09 (with $N_T = 128$) to 5.62 (with $N_T = 4$) times more iterations when $M = 10^{-1}$. These numbers increase to 2.53 and 51.48, respectively, when $M = 10^{-2}$. One may argue that the relative gains introduced by preconditioning are weakly dependent on the numerical stability of a given multi-stage scheme. This seems to be true even though a stronger numerical stability does lead to a smaller \bar{m} in the absence of preconditioning. Furthermore, PSDIRK2-SSS and PSDIRK3-AS also have time-step independent convergence rates, as observed for PIRK2. However, they are much closer to Mach number independent \bar{m} values than PIRK2.

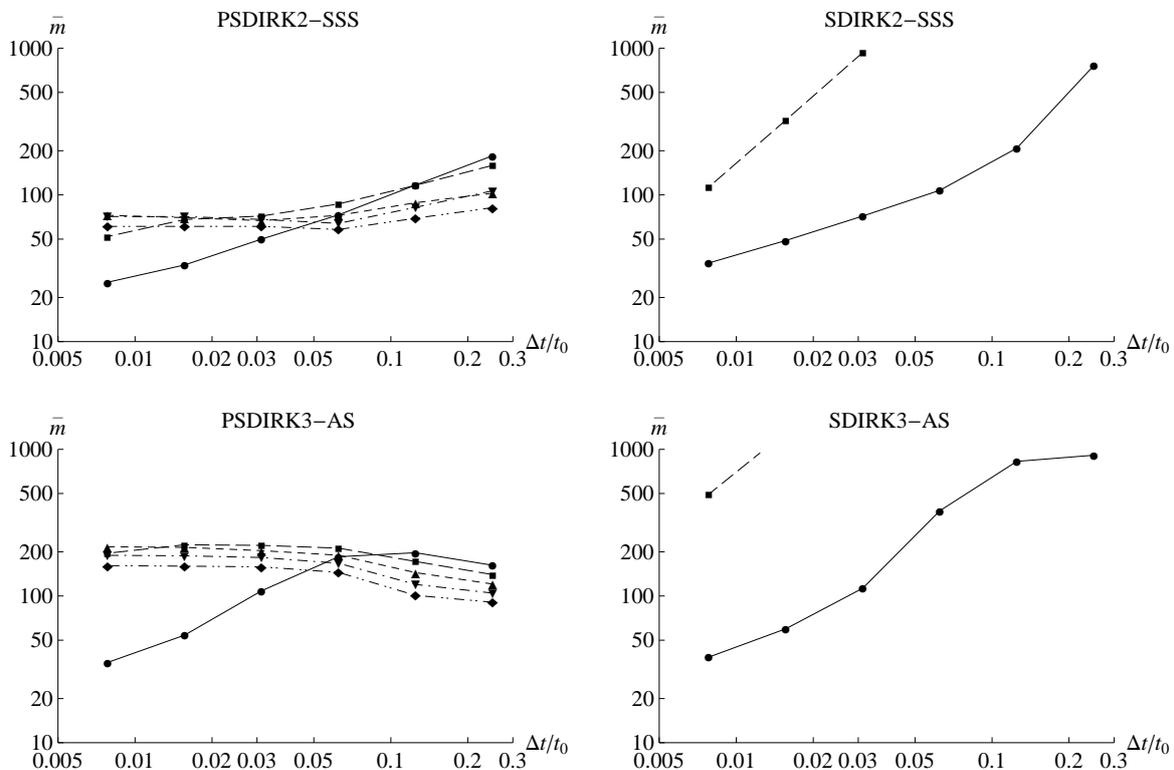


Figure 2. Optimal average number of pseudo-time iterations per dimensionless physical-time step from PSDIRK-SSS, SDIRK2-SSS, PSDIRK3-AS and SDIRK3-AS with $N_X = 52$. Curves generated with $M = 10^{-1}$ (solid), 10^{-2} (long dash), 10^{-3} (short dash), 10^{-4} (dash dot) and 10^{-5} (dash double dot).

4.3 Overall efficiency

It has already been shown that high-order multi-stage schemes are more efficient than multi-step schemes in the context of high speed compressible flows whenever error tolerances must be smaller than required by engineering applications. This is mostly due to the latter's limitations in terms of achievable accuracy orders. Results presented in this paper indicate the same is true for low speed compressible flows if PIRK methods are employed. Figure 3 provides quantitative data supporting this claim. It presents the absolute error data from previous figures as a fraction of the perturbation amplitude versus the number of pseudo-time iterations per physical-time period for $M = 10^{-5}$ with $t/t_0 = 1$ and 25. In the former case, PBD2 is the optimal choice only when errors higher than 10% can be tolerated. PSDIRK2-SSS is the best choice below this value, but only up to a certain point, since PSDIRK4-SSS outperforms all other methods if errors below 0.1% are required. PSDIRK3-SSS is not far behind the latter two methods in both cases. A more quantitative example is given by the imposition of a 1% tolerance: PSDIRK2-SSS is approximately 1.46 times faster than PSDIRK3-SSS,

1.76 than PSDIRK4-SSS, 2.25 than PSDIRK3-AS, 2.89 than PBDF2-LS, 4.16 than PCN2-AS and 4.93 than PSDIRK4-AS. Differences become even more pronounced at later physical-times. Still imposing a 1% error tolerance but now at $t/t_0 = 25$, PSDIRK4-SSS is estimated to be 1.22 times faster than PSDIRK3-SSS, 1.53 than PSDIRK2-SSS, 1.99 than PSDIRK3-AS, 2.88 than PSDIRK4-AS, 4.41 than PBDF2-LS and 6.57 than PCN2-AS. As noted earlier, the drop-off as $\bar{m}N_T$ decreases is due to excessive dissipation, which leads to an artificial steady-state solution. These results clearly show that preconditioned k -stage methods with higher accuracy-orders and stronger numerical stabilities perform better than their multi-step counterparts for low Mach number flows even though they solve k times more equations per pseudo-time step. One should also note that PIRK methods are very close to physical-time step and Mach number independent convergence rates. These features are responsible for making this high-order time integration technique more efficient than its multi-step counterparts for low Mach number compressible flows.

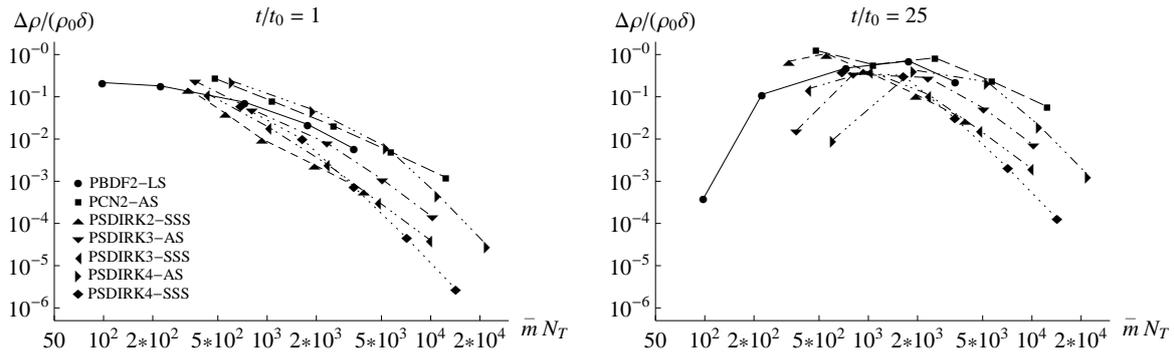


Figure 3. Dimensionless density's absolute error as a fraction of the perturbation amplitude versus the number of pseudo-time iterations per physical-time period generated with $M = 10^{-5}$ at $t/t_0 = 1$ and 25.

5. CONCLUSIONS AND FUTURE WORK

Preconditioned density-based methods were developed to eliminate stiffness caused by the large disparities between acoustic and convective time scales in low Mach number flows. They made possible numerical simulations of incompressible and low speed compressible flows using the compressible flow governing equations. This modified system of equations remains entirely hyperbolic at low speeds, allowing the use of well established time-marching schemes originally developed for high speed compressible flows. Even though preconditioning alters the natural transient evolution of the flow, its effect disappears when steady-state is reached, allowing accurate solutions to be recovered in this limit.

Time accurate simulations require a dual time stepping (DTS) formulation, which reintroduces the original physical-time derivative back into the system of equations modified by a preconditioned pseudo-time derivative. Marching these equations towards pseudo-time steady-state in every physical-time step yields an accurate transient solution of the original system of equations. In its traditional form, this formulation has only been used with multi-step schemes in physical-time. Hence, numerical simulations of even moderately stiff initial value problems were restricted to second-order accuracy because all higher order multi-step schemes are conditionally stable at best.

However, this physical-time accurate methodology is not limited to multi-step schemes. The present paper describes how this very same DTS formulation can be used to properly introduce a preconditioned pseudo-time derivative into the intermediate stages of implicit Runge-Kutta (IRK) methods. Hence, preconditioning techniques originally developed for steady and unsteady problems using multi-step schemes become available to their multi-stage counterparts. These capabilities are demonstrated by utilizing this so-called Preconditioned Implicit Runge-Kutta (PIRK) methods to numerically simulate the unsteady compressible Euler equations with Mach numbers as low as 10^{-5} in an efficient way. Several different singly-diagonally implicit Runge-Kutta (SDIRK) schemes with either A-stability or strong S-stability and second to fourth-orders of accuracy were selected to run the test cases. Now, the rich field of IRK methods with very high accuracy-orders and strong numerical stability already successfully applied to simulate high speed compressible flows can be adapted for the first time in a rigorous but straightforward way to low Mach number flows.

Current research is extending the use of PIRK methods for all speed flows to many different fronts, including flows in two and three dimensions. The impact of viscous effects in its performance are being investigated, although similar results are very likely. Different IRK schemes, such as ESDIRK, SIRK, IMEX and SSP, are also being preconditioned for low Mach number simulations of compressible flows. General linear methods, which combine multi-step and multi-stage schemes, are receiving similar attention. Finally, other acceleration techniques, such as multigrid, are being incorporated into these methods. These new techniques will be applied to several problems within the low Mach number compressible flow framework, such as thermal stability with non-Oberbeck-Boussinesq effects, supercritical fluid flow, transition to turbulence and aeroacoustics.

6. ACKNOWLEDGEMENTS

The author would like to thank Pradeep Singh Rawat, currently finishing his Ph.D. studies at the Mechanical and Aerospace Engineering Department in University of California at Los Angeles, USA, and Marcio Teixeira de Mendonça, senior researcher at the *Centro de Referência em Turbinas a Gás* in *Instituto Tecnológico de Aeronáutica*, Brazil, for helpful suggestions and comments. He also wants to acknowledge the financial support from *Conselho Nacional de Desenvolvimento Científico e Tecnológico* under grant number 480599/2007 – 6 from *Edital MCT/CNPq 15/2007 - Universal - Faixa A*.

7. REFERENCES

- J. Blazek. *Computational Fluid Dynamics: Principles and Applications*, chapter 9: Acceleration Techniques, pages 303–349. Elsevier Science, 2006.
- C. L. Merkle and M. Athavale. A time accurate unsteady incompressible algorithm based on artificial compressibility. In *AIAA Conference Paper*, 87-1137, 1987.
- A. J. Chorin. A numerical method for solving incompressible viscous flow problems. *Journal of Computational Physics*, 2:12–26, 1967.
- M. M. Rai. Navier-stokes simulations of blade-vortex interaction using high-order accurate upwind schemes. In *AIAA Conference Paper*, 87-0543, 1987.
- M. M. Rai and S. R. Chakravarthy. An implicit form for the osher upwind scheme. *AIAA Journal*, 24(5):735–743, 1986.
- E. Turkel. Review of preconditioning methods for fluid dynamics. *Applied Numerical Mathematics*, 12:257–284, 1993.
- E. Turkel. Preconditioning techniques in computational fluid dynamics. *Annual Review of Fluid Mechanics*, 31:385–416, 1999.
- E. Turkel. Preconditioned methods for solving the incompressible and low speed compressible equations. *Journal of Computational Physics*, 72(2):277–298, 1987.
- C. L. Merkle and Y. H. Choi. Computation of low speed compressible flows with time-marching methods. *International Journal for Numerical Methods in Engineering*, 25:293–311, 1988.
- P. E. O. Buelow, S. Venkateswaran, and C. L. Merkle. Effect of grid aspect ratio on convergence. *AIAA Journal*, 32(12):2402–2408, 1994.
- S. Venkateswaran, M. Deshpande, and C. L. Merkle. The application of preconditioning to reacting flow computations. In *AIAA Conference Paper*, 95-1673, pages 306–316, 1995.
- A. Hosangadi, C. L. Merkle, and S. R. Turns. Analysis of forced combustor jets. *AIAA Journal*, 28(8):1473–1480, 1990.
- P. Wesseling. *An Introduction to Multigrid Methods*. Pure and Applied Mathematics. John Wiley & Sons, New York, 1991.
- A. Jameson. Time dependent calculations using multigrid, with applications to unsteady flows past airfoils and wings. In *AIAA Conference Paper*, 91-1596, 1991.
- J. S. Shuen, K. H. Chen, and Y. H. Choi. A coupled implicit method for chemical non-equilibrium flows at all speeds. *Journal of Computational Physics*, 106:306–318, 1993.
- S. De Rango and D. W. Zingg. Improvements to a dual time stepping method for computing unsteady flows. *AIAA Journal*, 35(9):1548–1550, 1996.
- L. D. Dailey and R. H. Pletcher. Evaluation of multigrid acceleration for preconditioned time-accurate Navier-Stokes algorithms. *Computer & Fluids*, 25(8):791–811, 1996.
- C. Liu, X. Zheng, and C. H. Sung. Preconditioned multigrid methods for unsteady incompressible flows. *Journal of Computational Physics*, 139(1):35–57, 1998.
- F. Battaglia, A. K. Kulkarni, J. Feng, and C. L. Merkle. Simulations of planar flapping jets in confined channels. *AIAA Journal*, 36(8):1425–1431, 1998.
- C. M. Shieh and P. J. Morris. High-order accurate dual time-stepping algorithm for viscous aeroacoustic simulations. In *AIAA Conference Paper*, 98-2361, pages 912–922, 1998.
- L. S. B. Alves. Review of numerical methods for the compressible flow equations at low mach numbers. In *XII Encontro de Modelagem Computacional*, Rio de Janeiro, RJ, Brazil, Dezembro 2009.
- J. Suh, S. H. Frankel, L. Mongeau, and M. W. Plesniak. Compressible large eddy simulations of wall-bounded turbulent flows using a semi-implicit numerical scheme for low Mach number aeroacoustics. *Journal of Computational Physics*, 215(2):526–551, 2006.
- G. Dahlquist. A special stability problem for linear multistep methods. *BIT Numerical Mathematics*, 3:27–43, 1963.
- H. Lomax, T. H. Pulliam, and D. W. Zingg. *Fundamentals of Computational Fluid Dynamics*. Scientific Computation. Springer & Verlag, Berlin, 2001.
- H. Bijl, M. H. Carpenter, V. N. Vatsa, and C. A. Kennedy. Implicit time integration schemes for the unsteady compressible Navier-Stokes equations: Laminar flow. *Journal of Computational Physics*, 179:313–329, 2002.
- J. C. Butcher. *Numerical Methods for Ordinary Differential Equations*. John Wiley & Sons, Inc., England, 2008.
- X. Zhong. Additive semi-implicit Runge-Kutta methods for computing high-speed nonequilibrium reactive flows. *Journal of Computational Physics*, 128:19–31, 1996.

- U. M. Ascher, S. J. Ruuth, and R. J. Spiteri. Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations. *Applied Numerical Mathematics*, 25:151–167, 1997.
- C. A. Kennedy and M. H. Carpenter. Additive Runge-Kutta schemes for convection-diffusion-reaction equations. *Applied Numerical Mathematics*, 44:139–181, 2003.
- J. J. Yoh and X. Zhong. New hybrid Runge-Kutta methods for unsteady reactive flow simulation. *AIAA Journal*, 42(8):1593–1600, 2004a.
- J. J. Yoh and X. Zhong. New hybrid Runge-Kutta methods for unsteady reactive flow simulation: Applications. *AIAA Journal*, 42(8):1601–1611, 2004b.
- L. Pareschi and G. Russo. Implicit-explicit Runge-Kutta schemes and applications to hyperbolic systems with relaxation. *Journal of Scientific Computing*, 25(1):129–155, 2005.
- L. Ferracina and M. N. Spijker. Strong stability of singly-diagonally-implicit runge-kutta methods. *Applied Numerical Mathematics*, 58:1675–1686, 2008.
- G. Jothiprasad, D. J. Mavriplis, and D. A. Caughey. Higher-order time integration schemes for the unsteady Navier-Stokes equations on unstructured meshes. *Journal of Computational Physics*, 191(2):542–566, 2003.
- H. Bijl and M. H. Carpenter. Iterative solution techniques for unsteady flow computations using higher order time integration schemes. *International Journal for Numerical Methods in Fluids*, 47:857–862, 2005.
- L. Wang and D. J. Mavriplis. Implicit solution of the unsteady Euler equations for high-order accurate discontinuous Galerkin discretizations. *Journal of Computational Physics*, 225(2):1994–2015, 2007.
- E. Turkel and V. N. Vatsa. Local preconditioners for steady and unsteady flow applications. *ESAIM: Mathematical Modelling and Numerical Analysis*, 39(3):515–535, 2005.
- R. Alexander. The modified newton method in the solution of stiff ordinary differential equations. *SIAM Journal on Numerical Analysis*, 14(6):1006–1021, 1977.
- J. Sesterhenn, B. Müller, and H. Thomann. On the cancellation problem in calculating compressible low Mach number flows. *Journal of Computational Physics*, 151(2):597–615, 1999.
- S.-H. Lee. Cancellation problem of preconditioning method at low Mach numbers. *Journal of Computational Physics*, 225(2):1199–1210, 2007.
- S. Venkateswaran and C. L. Merkle. Efficiency and accuracy issues in contemporary cfd algorithms. In *AIAA Conference Paper*, 2000-2251, pages 1–16, 2000.
- C. L. Merkle, J. Y. Sullivan, P. E. O. Buelow, and S. Venkateswaran. Computation of flows with arbitrary equations of state. *AIAA Journal*, 36(4):515–521, 1998.
- D. A. Schwer. *Numerical Study of Unsteadiness in Non-Reacting and Reacting Mixing Layers*. PhD thesis, Pennsylvania State University, 1999.
- P. E. O. Buelow, S. Venkateswaran, and C. L. Merkle. Stability and convergence analysis of implicit upwind schemes. *Computers and Fluids*, 30:961–988, 2001.
- J. C. Tannehill, D. A. Anderson, and R. H. Pletcher. *Computational Fluid Mechanics and Heat Transfer*. Taylor & Francis, Philadelphia, 1997.
- W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in FORTRAN 77: The Art of Scientific Computing*. Cambridge University Press, Cambridge, 2nd edition, 1992.

8. Responsibility notice

The author(s) is (are) the only responsible for the printed material included in this paper