Journal of Computational Mathematics, Vol.??, No.??, 2008, 1–21.

# VARIABLE STEP-SIZE IMPLICIT-EXPLICIT LINEAR MULTISTEP METHODS FOR TIME-DEPENDENT PARTIAL DIFFERENTIAL EQUATIONS*

Dong Wang
*Department of Civil and Environmental Engineering, University of Illinois at Urbana-Champaign, B231 Newmark Civil Engineering Laboratory, Urbana, IL 61801, USA*
*Email: dongwang@uiuc.edu*
Steven J. Ruuth
*Department of Mathematics, Simon Fraser University, Burnaby, BC, Canada V5A 1S6*
*Email: sruuth@sfu.ca*

**Abstract**

Implicit-explicit (IMEX) linear multistep methods are popular techniques for solving partial differential equations (PDEs) with terms of different types. While fixed time-step versions of such schemes have been developed and studied, implicit-explicit schemes also naturally arise in general situations where the temporal smoothness of the solution changes. In this paper we consider easily implementable variable step-size implicit-explicit (VSIMEX) linear multistep methods for time-dependent PDEs. Families of order-$p$, $p$-step VSIMEX schemes are constructed and analyzed, where $p$ ranges from 1 to 4. The corresponding schemes are simple to implement and have the property that they reduce to the classical IMEX schemes whenever constant time step-sizes are imposed. The methods are validated on the Burgers' equation. These results demonstrate that by varying the time step-size, VSIMEX methods can outperform their fixed time step counterparts while still maintaining good numerical behavior.

*Mathematics subject classification:* 65L06, 65M06, 65M20.
*Key words:* Implicit-explicit (IMEX) linear multistep methods, variable step-size, zero-stability, Burgers' equation.

## 1. Introduction

Many problems in physics, engineering, chemistry, biology and other areas involve the numerical solution of time-dependent partial differential equations (PDEs). Some types of PDEs can conveniently be transformed into large systems of ordinary differential equations (ODEs) in time by doing spatial discretizations based on finite difference methods, finite volume methods, spectral methods or finite element methods.

The corresponding large systems of ODEs often take the form

$$\dot{u} = f(u) + g(u). \tag{1.1}$$

The term $f(u)$ is a nonstiff and possibly nonlinear term which we do not wish to integrate implicitly. This might be because an iterative solution to the implicit equations is desired and the Jacobian of $f(u)$ is nonsymmetric and nondefinite, or the Jacobian of $f(u)$ could be dense. Or, perhaps, we may wish to take $f(u)$ explicitly for ease of implementation. The remaining

* Received / accepted /

term, $g(u)$, is a stiff term, and must be taken implicitly to avoid excessively small time steps. Thus it makes sense to treat $g(u)$ implicitly and $f(u)$ explicitly according to an IMEX scheme. See, e.g., [2, 3, 9] for further details on these powerful techniques.

For solutions of ODEs (1.1) with different time scales, i.e. solutions rapidly varying in some regions of the time domain while slowly changing in other regions, variable step-sizes are often essential to obtain computationally efficient, accurate results. For example, small time steps may be necessary to capture rapidly varying initial transients, while large time steps may be desirable to capture the subsequent slowly changing, long-term evolution of the system.

However, standard IMEX linear multistep methods are designed for the case of constant step-sizes. Thus starting values must be computed every time the temporal step-size is varied for these methods. A commonly used approach for handling variable step-sizes for linear multistep methods is the interpolation method [11]. Using this approach, all the starting values for the new step-size may need to be calculated using an interpolation method each time the temporal step-size is changed. Unfortunately, this process is sufficiently complicated that it is often avoided in practice.

This paper has three main objectives. The first of these is to develop easily implementable VSIMEX schemes up to fourth-order. The second objective is to explore the relationship between zero-stability of our VSIMEX schemes and variable step-sizes. The last objective is to numerically validate the proposed VSIMEX schemes.

The paper unfolds as follows. In Section 2, a variety of $p$-step, order-$p$ VSIMEX linear multistep schemes are derived. For order-$p$, $1 \leq p \leq 3$, a $p$-parameter family of schemes is presented. For order-four we only provide a VSIMEX scheme based on the popular fourth-order backward differentiation formula (BDF4) method. The zero-stability analysis of VSIMEX schemes is reviewed and studied in Section 3. Section 4 carries out numerical experiments for the Burgers' equation using various IMEX and VSIMEX schemes. In this section, accurate approximate solutions are obtained, and the expected orders of convergence for VSIMEX schemes are verified for a variety of time-stepping strategies. Finally, Section 5 contains a summary of this paper.

## 2. Derivation of VSIMEX Schemes

In this section, various VSIMEX linear multistep schemes up to fourth-order are derived. The first-, second- and third-order VSIMEX linear multistep schemes are families of methods which admit one, two and three free parameters respectively. Among all the fourth-order VSIMEX linear multistep scheme, we focus on the fourth-order variable step-size, semi-implicit, backward differentiation formula (VSSBDF4).

### 2.1. General VSIMEX Linear Multistep Methods

We consider an arbitrary grid $\{t_n\}$ and denote the step-size $k_{n+j} = t_{n+j+1} - t_{n+j}$. Furthermore, we assume that the previous $s$ approximations $U^{n+j}$ to $u(t_{n+j}), j = 0, 1, \ldots, s-1$, are known.

The general $s$-step VSIMEX linear multistep schemes for ODEs (1.1) take the form

$$\frac{1}{k_{n+s-1}} \sum_{j=0}^{s} \alpha_{j,n} U^{n+j} = \sum_{j=0}^{s-1} \beta_{j,n} f(U^{n+j}) + \sum_{j=0}^{s} \gamma_{j,n} g(U^{n+j}), \tag{2.1}$$

where $\alpha_{s,n} \neq 0$, $\gamma_{s,n} \neq 0$ and $s \geq 2$. The variable coefficients $\alpha_{j,n}$, $\beta_{j,n}$ and $\gamma_{j,n}$ are functions

of the step-size ratios $\omega_i = k_i/k_{i-1}$ for $i = n+1, ..., n+s-1$, $s \geq 2$ and must satisfy the order conditions (2.4) listed below.

Ascher, Ruuth and Wetton [2] proved for fixed step-sizes $k$ that $s$-step IMEX schemes can have at most order-$s$ accuracy and that this is achieved by an $s$-parameter family of schemes. In this paper, we only consider $s$-step, $O(\overline{k}^s)$ VSIMEX linear multistep schemes, where $\overline{k}$ is the average temporal step-size.

It is assumed that the step-size ratios $k_i/k_n$ and the variable coefficients $\alpha_{j,n}$, $\beta_{j,n}$ and $\gamma_{j,n}$ are all bounded for $i = n+1, ..., n+s-1$. Replacing the approximate solutions $U^{n+j}$, $j = 0, 1, ..., s$ by the corresponding exact solutions $u(t_{n+j})$ in the variable coefficient difference equation (2.1) yields the local truncation error, $\tau_n$,

$$\tau_n = \frac{1}{k_{n+s-1}} \sum_{j=0}^{s} \alpha_{j,n} u(t_{n+j}) - \sum_{j=0}^{s-1} \beta_{j,n} f \circ u(t_{n+j}) - \sum_{j=0}^{s} \gamma_{j,n} g \circ u(t_{n+j}), \tag{2.2}$$

where the composite functions $f \circ u, g \circ u$ are defined as $f \circ u(t) = f(u(t))$ and $g \circ u(t) = g(u(t))$ respectively.

For a smooth function $u(t)$, expanding equation (2.2) in a Taylor series about $t_n$ yields

$$\begin{aligned}
\tau_n = \frac{1}{k_{n+s-1}} \Bigg\{ & \alpha_{0,n} u(t_n) + \sum_{j=1}^{s} \alpha_{j,n} \Bigg[ u(t_n) + \dot{u}(t_n) \sum_{i=0}^{j-1} k_{n+i} + \\
& + \frac{\ddot{u}(t_n)}{2!} \left( \sum_{i=0}^{j-1} k_{n+i} \right)^2 + \cdots + \frac{u^{(p)}(t_n)}{p!} \left( \sum_{i=0}^{j-1} k_{n+i} \right)^p \Bigg] \Bigg\} \\
& - \beta_{0,n} f \circ u(t_n) - \sum_{j=1}^{s-1} \beta_{j,n} \Bigg[ f \circ u(t_n) + \frac{d(f \circ u)}{dt}(t_n) \sum_{i=0}^{j-1} k_{n+i} + \cdots \\
& + \frac{1}{(p-1)!} \frac{d^{(p-1)}(f \circ u)}{dt^{(p-1)}}(t_n) \left( \sum_{i=0}^{j-1} k_{n+i} \right)^{p-1} \Bigg] - \gamma_{0,n} g \circ u(t_n) \\
& - \sum_{j=1}^{s} \gamma_{j,n} \Bigg[ g \circ u(t_n) + \frac{d(g \circ u)}{dt}(t_n) \sum_{i=0}^{j-1} k_{n+i} + \cdots \\
& + \frac{1}{(p-1)!} \frac{d^{(p-1)}(g \circ u)}{dt^{(p-1)}}(t_n) \left( \sum_{i=0}^{j-1} k_{n+i} \right)^{p-1} \Bigg] + O(k_n^p).
\end{aligned} \tag{2.3}$$

Applying equation (1.1) to the local truncation error (2.3), a $p$th-order VSIMEX scheme is

obtained provided that the following constraints for $\alpha_{j,n}$, $\beta_{j,n}$ and $\gamma_{j,n}$ hold:

$$\begin{aligned}
\sum_{j=0}^{s} \alpha_{j,n} &= 0, \\
\sum_{j=1}^{s} \alpha_{j,n} \left( \sum_{i=0}^{j-1} k_{n+i} \right) &= k_{n+s-1} \sum_{j=0}^{s-1} \beta_{j,n} = k_{n+s-1} \sum_{j=0}^{s} \gamma_{j,n}, \\
&\vdots \\
\frac{1}{p!} \sum_{j=1}^{s} \alpha_{j,n} \left( \sum_{i=0}^{j-1} k_{n+i} \right)^{p} &= k_{n+s-1} \frac{1}{(p-1)!} \sum_{j=1}^{s-1} \beta_{j,n} \left( \sum_{i=0}^{j-1} k_{n+i} \right)^{p-1} \\
&= k_{n+s-1} \frac{1}{(p-1)!} \sum_{j=1}^{s} \gamma_{j,n} \left( \sum_{i=0}^{j-1} k_{n+i} \right)^{p-1}.
\end{aligned} \tag{2.4}$$

Taken together, these constraints are known as the order conditions.

### 2.2. First-Order VSIMEX Schemes

First-order, one-step IMEX schemes are actually VSIMEX schemes, since they allow for variable time-stepping. This one-parameter family of schemes for (1.1) can be expressed as [2]

$$U^{n+1} - U^n = k_n f(U^n) + k_n[(1-\gamma) g(U^n) + \gamma g(U^{n+1})]. \tag{2.5}$$

The leading order term in the local truncation error in (2.5) is given by

$$\frac{k_n}{2} \ddot{u}(t_n) - k_n \gamma \frac{d(g \circ u)}{dt}(t_n),$$

which suggests the restriction $\gamma \in [0, 1]$ to maintain a moderate local truncation error.

A few schemes in this one-parameter family are familiar:

1. $\gamma = \frac{1}{2}$ gives
$$U^{n+1} - U^n = k_n f(U^n) + \frac{1}{2} k_n [g(U^n) + g(U^{n+1})],$$
which applies the second-order, one-step Crank-Nicolson method to $g(u)$ and the forward Euler method to $f(u)$.

2. $\gamma = 1$ yields
$$U^{n+1} - U^n = k_n f(U^n) + k_n g(U^{n+1}), \tag{2.6}$$
which applies backward Euler to $g(u)$ and forward Euler to $f(u)$. As we know, the backward Euler method is the first-order member of the class of backward differentiation formulas (BDFs) [10]. In the following sections, we will also develop some order-$p$ VSIMEX schemes ($p = 2, 3, 4$) similar to (2.6), which apply BDFs to $g$ and extrapolate $f$ to time step $t_{n+p}$. Those schemes will be referred to as order-$p$ variable step-size semi-implicit BDF (VSSBDF$p$) schemes.

We now focus our attention on general second-order, two-step VSIMEX schemes with two free parameters, and highlight some particular VSIMEX schemes whose corresponding IMEX schemes are quite familiar to us.

### 2.3. Second-Order VSIMEX Schemes

Second-order, two-step VSIMEX schemes admit two free parameters. If our VSIMEX schemes are centered in time about time-step $t_{n+1+\gamma}$ to second-order, we derive second-order VSIMEX schemes, viz., a family of schemes involving two parameters $(\gamma, c)$ for which equation (2.1) is

$$\frac{1}{k_{n+1}}\sum_{j=0}^{2}\alpha_{j,n}U^{n+j} = \sum_{j=0}^{1}\beta_{j,n}f(U^{n+j}) + \sum_{j=0}^{2}\gamma_{j,n}g(U^{n+j}) \tag{2.7}$$

where

$$\begin{aligned}
\alpha_{0,n} &= \frac{(2\gamma-1)\omega_{n+1}^2}{1+\omega_{n+1}}, & \beta_{1,n} &= 1+\gamma\omega_{n+1},\\
\alpha_{1,n} &= (1-2\gamma)\omega_{n+1}-1, & \gamma_{0,n} &= \frac{c}{2},\\
\alpha_{2,n} &= \frac{1+2\gamma\omega_{n+1}}{1+\omega_{n+1}}, & \gamma_{1,n} &= 1-\gamma-(1+\frac{1}{\omega_{n+1}})\frac{c}{2},\\
\beta_{0,n} &= -\gamma\omega_{n+1}, & \gamma_{2,n} &= \gamma+\frac{c}{2\omega_{n+1}},
\end{aligned}$$

and $\gamma \in [0,1]$.

In the constant step-size case (i.e., if we set all consecutive step-size ratios $\omega_{n+1} = 1$ for all $n = 0, 1, \ldots, N-2$, where $N$ is the number of total nodes in time interval $[\,0, T\,]$), the schemes (2.7) reduce to the family of IMEX schemes described in equation (14) in [2]. The following lists some VSIMEX schemes whose IMEX counterparts are given in Section 3.2 in [2]:

1. $(\gamma, c) = (1, 0)$ gives

$$\begin{aligned}
\frac{1}{k_{n+1}}\left[\frac{1+2\omega_{n+1}}{1+\omega_{n+1}}U^{n+2} - (1+\omega_{n+1})U^{n+1} + \frac{\omega_{n+1}^2}{1+\omega_{n+1}}U^{n}\right] =\\
(1+\omega_{n+1})f(U^{n+1}) - \omega_{n+1}f(U^{n}) + g(U^{n+2}).
\end{aligned} \tag{2.8}$$

   As mentioned in Section 2.2, this scheme applies a variable step-size BDF2 scheme to the stiff part and extrapolates the nonstiff part to time step $t_{n+2}$. This scheme will be referred to as the variable step-size second-order semi-implicit BDF (VSSBDF2).

2. $(\gamma, c) = (\frac{1}{2}, 0)$ gives

$$\begin{aligned}
\frac{1}{k_{n+1}}\left[U^{n+2} - U^{n+1}\right] &= \left(1+\frac{1}{2}\omega_{n+1}\right)f(U^{n+1}) - \frac{1}{2}\omega_{n+1}f(U^{n})\\
&\quad + \frac{1}{2}\left[g(U^{n+2}) + g(U^{n+1})\right].
\end{aligned} \tag{2.9}$$

   Since it applies Crank-Nicolson to the stiff term and the variable step-size second-order Adams-Bashforth scheme to the nonstiff term, this scheme is referred to VSCNAB (variable step-size Crank-Nicolson, Adams-Bashforth).

3. $(\gamma, c) = (\frac{1}{2}, \frac{1}{8})$ gives

$$\begin{aligned}
\frac{1}{k_{n+1}}\left(U^{n+2} - U^{n+1}\right) &= \frac{1}{2}\left[(2+\omega_{n+1})f(U^{n+1}) - \omega_{n+1}f(U^{n})\right]\\
&\quad + \frac{1}{16\omega_{n+1}}\left[(8\omega_{n+1}+1)g(U^{n+2}) + (7\omega_{n+1}-1)g(U^{n+1}) + \omega_{n+1}g(U^{n})\right].
\end{aligned} \tag{2.10}$$

The scheme is referred to as the modified VSCNAB, corresponding to its constant step-size version called the modified CNAB method [2].

4. $(\gamma, c) = (0, 1)$ gives

$$\frac{1}{k_{n+1}}\left[\frac{1}{1+\omega_{n+1}}U^{n+2} + (\omega_{n+1}-1)U^{n+1} - \frac{\omega_{n+1}^2}{1+\omega_{n+1}}U^n\right] = f(U^{n+1}) + \frac{1}{2}\left[\frac{1}{\omega_{n+1}}g(U^{n+2}) + (1-\frac{1}{\omega_{n+1}})g(U^{n+1}) + g(U^n)\right]. \tag{2.11}$$

which applies a scheme somewhat like the Crank-Nicolson to the stiff part and a Leap-Frog scheme to nonstiff part. In [2], its IMEX counterpart is referred to as CNLF (Crank-Nicolson, Leap-Frog). Consequently, we call scheme (2.11) VSCNLF (variable step-size CNLF).

### 2.4. Third- and Fourth- Order VSIMEX Schemes

Third-order, three-step VSIMEX schemes admit three free parameters. One particular parameterization for equation (2.1) can be derived by introducing the three parameters $(\gamma, \theta, c)$. This leads to

$$\frac{1}{k_{n+2}}\sum_{j=0}^{3}\alpha_{j,n}U^{n+j} = \sum_{j=0}^{2}\beta_{j,n}f(U^{n+j}) + \sum_{j=0}^{3}\gamma_{j,n}g(U^{n+j}), \tag{2.12}$$

where

$$
\begin{aligned}
\alpha_{0,n} &= -\frac{\omega_{n+1}^3\omega_{n+2}^2[3\gamma^2\omega_{n+2}+2\gamma(1-\omega_{n+2})-1]}{(1+\omega_{n+1})[1+\omega_{n+1}(1+\omega_{n+2})]},\\
\alpha_{1,n} &= -\frac{\omega_{n+2}^2[\gamma\omega_{n+1}\omega_{n+2}(2-3\gamma)+(1-2\gamma)(1+\omega_{n+1})]}{1+\omega_{n+2}},\\
\alpha_{2,n} &= -\frac{\omega_{n+1}(1+\gamma\omega_{n+2})[1+\omega_{n+2}(3\gamma-2)]+\omega_{n+2}(2\gamma-1)+1}{1+\omega_{n+1}}-\theta,\\
\alpha_{3,n} &= \frac{1+2\gamma\omega_{n+2}+\omega_{n+1}(1+\gamma\omega_{n+2})(1+3\gamma\omega_{n+2})}{(1+\omega_{n+2})[1+\omega_{n+1}(1+\omega_{n+2})]}+\theta,\\
\beta_{0,n} &= \frac{\omega_{n+1}^2\omega_{n+2}[6\gamma(1+\gamma\omega_{n+2})+\theta(3+2\omega_{n+2})]}{6(1+\omega_{n+1})},\\
\beta_{1,n} &= -\gamma\omega_{n+2}[1+\omega_{n+1}(1+\gamma\omega_{n+2})]-\frac{\theta\omega_{n+2}[3+\omega_{n+1}(3+2\omega_{n+2})]}{6},\\
\beta_{2,n} &= \frac{(1+\gamma\omega_{n+2})[1+\omega_{n+1}(1+\gamma\omega_{n+2})]}{(1+\omega_{n+1})}\\
&\quad+\theta\left[1+\frac{\omega_{n+2}}{2}+\frac{\omega_{n+1}\omega_{n+2}(3+2\omega_{n+2})}{6(1+\omega_{n+1})}\right],\\
\gamma_{0,n} &= \frac{\theta\omega_{n+1}^2\omega_{n+2}(3+2\omega_{n+2})}{6(1+\omega_{n+1})}-c,\\
\gamma_{1,n} &= \frac{c(1+\omega_{n+1})[1+\omega_{n+1}(1+\omega_{n+2})]-\omega_{n+1}^2\omega_{n+2}^2\gamma(1-\gamma)}{\omega_{n+1}^2(1+\omega_{n+2})}\\
&\quad-\frac{\theta\omega_{n+2}[3+\omega_{n+1}(3+2\omega_{n+2})]}{6},\\
\gamma_{2,n} &= \frac{\omega_{n+1}^2\omega_{n+2}(1-\gamma)(1+\gamma\omega_{n+2})-c[1+\omega_{n+1}(1+\omega_{n+2})]}{\omega_{n+1}^2\omega_{n+2}}\\
&\quad+\theta\left[1+\frac{\omega_{n+2}}{2}+\frac{\omega_{n+1}\omega_{n+2}(3+2\omega_{n+2})}{6(1+\omega_{n+1})}\right],\\
\gamma_{3,n} &= \frac{\omega_{n+1}^2\omega_{n+2}\gamma(1+\gamma\omega_{n+2})+c(1+\omega_{n+1})}{\omega_{n+1}^2\omega_{n+2}(1+\omega_{n+2})}.
\end{aligned}
\tag{2.13}
$$

Note that scheme (2.12) is centered at time step $t_{n+2+\gamma}$ whenever $\theta = 0$. Restricting scheme (2.12) to constant step-sizes leads to the three-parameter family of third-order IMEX schemes introduced in equation (23) in [2]. Setting $(\gamma, \theta, c) = (1, 0, 0)$ in scheme (2.12) yields the variable step-size third-order semi-implicit BDF (VSSBDF3)

$$
\frac{1}{k_{n+2}}\sum_{j=0}^{3}\alpha_{j,n}U^{n+j} = \sum_{j=0}^{2}\beta_{j,n}f(U^{n+j}) + g(U^{n+3}),
\tag{2.14}
$$

where

$$
\begin{aligned}
\alpha_{0,n} &= -\frac{\omega_{n+1}^3\omega_{n+2}^2(1+\omega_{n+2})}{(1+\omega_{n+1})(1+\omega_{n+1}+\omega_{n+1}\omega_{n+2})},\\
\alpha_{1,n} &= \omega_{n+2}^2(\omega_{n+1}+\frac{1}{1+\omega_{n+2}}),\\
\alpha_{2,n} &= -1-\omega_{n+2}-\frac{\omega_{n+1}\omega_{n+2}(1+\omega_{n+2})}{1+\omega_{n+1}},\\
\alpha_{3,n} &= 1+\frac{\omega_{n+2}}{1+\omega_{n+2}}+\frac{\omega_{n+1}\omega_{n+2}}{1+\omega_{n+1}(1+\omega_{n+2})},\\
\beta_{0,n} &= \frac{\omega_{n+1}^2\omega_{n+2}(1+\omega_{n+2})}{1+\omega_{n+1}},\\
\beta_{1,n} &= -\omega_{n+2}[1+\omega_{n+1}(1+\omega_{n+2})],\\
\beta_{2,n} &= \frac{(1+\omega_{n+2})[1+\omega_{n+1}(1+\omega_{n+2})]}{1+\omega_{n+1}}.
\end{aligned}
$$

The constant step-size version of this scheme, SBDF3, is a particularly popular scheme and performs well on problems with stiff dissipative terms, so we shall limit our analysis and numerical tests to this member of the three parameter family.

For the fourth-order, four-step VSIMEX schemes, we only develop the variable step-size fourth-order, four-step semi-implicit BDF (VSSBDF4) scheme. This particular fourth-order scheme for equation (2.1) has the form

$$
\frac{1}{k_{n+3}}\sum_{j=0}^{4}\alpha_{j,n}U^{n+j} = \sum_{j=0}^{3}\beta_{j,n}f(U^{n+j}) + g(U^{n+4}), \tag{2.15}
$$

where

$$\begin{aligned}
\alpha_{0,n} &= \frac{1+\omega_{n+3}}{1+\omega_{n+1}}\frac{A_2}{A_1}\frac{\omega_{n+1}^4\omega_{n+2}^3\omega_{n+3}^2}{A_3},\\
\alpha_{1,n} &= -\omega_{n+2}^3\omega_{n+3}^2\frac{1+\omega_{n+3}}{1+\omega_{n+2}}\frac{A_3}{A_2},\\
\alpha_{2,n} &= \omega_{n+3}\left[\frac{\omega_{n+3}}{1+\omega_{n+3}} + \omega_{n+2}\omega_{n+3}\frac{A_3+\omega_{n+1}}{1+\omega_{n+1}}\right],\\
\alpha_{3,n} &= -1-\omega_{n+3}\left[1+\frac{\omega_{n+2}(1+\omega_{n+3})}{1+\omega_{n+2}}(1+\frac{\omega_{n+1}A_2}{A_1})\right],\\
\alpha_{4,n} &= 1+\frac{\omega_{n+3}}{1+\omega_{n+3}}+\frac{\omega_{n+2}\omega_{n+3}}{A_2}+\frac{\omega_{n+1}\omega_{n+2}\omega_{n+3}}{A_3},\\
\beta_{0,n} &= -\omega_{n+1}^3\omega_{n+2}^2\omega_{n+3}\frac{1+\omega_{n+3}}{1+\omega_{n+1}}\frac{A_2}{A_1},\\
\beta_{1,n} &= \omega_{n+2}^2\omega_{n+3}\frac{1+\omega_{n+3}}{1+\omega_{n+2}}A_3,\\
\beta_{2,n} &= -A_2A_3\frac{\omega_{n+3}}{1+\omega_{n+1}},\\
\beta_{3,n} &= \frac{\omega_{n+2}(1+\omega_{n+3})}{1+\omega_{n+2}}\frac{(1+\omega_{n+3})(A_3+\omega_{n+1})+\frac{1+\omega_{n+1}}{\omega_{n+2}}}{A_1},\\
A_1 &= 1+\omega_{n+1}(1+\omega_{n+2}),\\
A_2 &= 1+\omega_{n+2}(1+\omega_{n+3}),\\
A_3 &= 1+\omega_{n+1}A_2.
\end{aligned}$$

In the constant temporal step-size case, equation (2.15) reduces to the fourth-order SBDF; see equation (33) in [2].

## 3. Zero-Stability Analysis of VSIMEX Schemes

In this section, we study the stability properties of our VSIMEX schemes. As we know, the necessary and sufficient conditions for a linear multistep method to be convergent are that it is both consistent and zero-stable [10]. If the constraints (2.4) are satisfied, then the VSIMEX methods (2.1) are consistent. Hence, we are interested in the zero-stability properties of our VSIMEX schemes, namely, what restrictions on step-size ratios $\omega_i$, $i = n+1,\ldots,n+s-1$ are required in order to ensure order-$s$, $s$-step VSIMEX schemes are stable.

Zero-stability measures how computational errors, such as errors in the starting values, round-off errors, etc., propagate as the computation proceeds and as the temporal step-size approaches zero. A zero-stable linear multistep method is insensitive to perturbations such as round-off errors [10]. Thus, zero-stability is an essential property of any usable linear multistep method. In this section, our goal is to find restrictions on the step-size variations which ensure that our VSIMEX schemes are zero-stable. Our presentation mainly follows the zero-stability analysis of variable step-size multistep methods explained in [6] and [8].

Applying the order-$s$, $s$-step VSIMEX method (2.1) to the scalar differential equation $\dot{u} = 0$ yields the variable coefficient difference equation

$$U^{n+s} + \sum_{j=0}^{s-1}\lambda_{j,n}U^{n+j} = 0, \tag{3.1}$$

where

$$\lambda_{j,n} = \frac{\alpha_{j,n}}{\alpha_{s,n}}, \text{for } \; j = 0, 1, \ldots, s-1, \;\; \text{and } \alpha_{j,n}, \; j = 0, 1, \cdots, s \text{ are defined in (2.1).}$$

We define the polynomial $\rho_n(z)$ of degree $s$,

$$\rho_n(z) = z^s + \sum_{j=0}^{s-1} \lambda_{j,n} z^j \tag{3.2}$$

and remark that the consistency condition of our VSIMEX schemes implies

$$\rho_n(1) = 0, \qquad n = 0, 1, \ldots, N-s,$$

where $N$ is the number of total nodes in time interval $[\,0, T\,]$.

Define the divided polynomials $\rho_n^*(z)$ of degree $(s-1)$ [6]

$$\rho_n^*(z) = \frac{\rho_n(z)}{z-1} = z^{s-1} + \sum_{j=0}^{s-2} \lambda_{j,n}^* z^j. \tag{3.3}$$

From (3.2) and (3.3), we have

$$\begin{aligned} \lambda_{s-2,n}^* &= 1 + \lambda_{s-1,n}, \\ \lambda_{0,n}^* &= -\lambda_{0,n}, \\ \lambda_{s-j-1,n}^* - \lambda_{s-j,n}^* &= \lambda_{s-j,n}, \;\; \text{for } \; j = 2, \ldots, s-1. \end{aligned}$$

Note that equation (3.1) may be written in the matrix-vector form

$$\begin{bmatrix} U^{n+s} \\ U^{n+s-1} \\ \vdots \\ U^{n+2} \\ U^{n+1} \end{bmatrix} = \begin{bmatrix} -\lambda_{s-1,n} & -\lambda_{s-2,n} & \cdots & . & -\lambda_{0,n} \\ 1 & 0 & \cdots & . & 0 \\ & 1 & & . & 0 \\ & & \ddots & \vdots & \vdots \\ & & & 1 & 0 \end{bmatrix} \begin{bmatrix} U^{n+s-1} \\ U^{n+s-2} \\ \vdots \\ U^{n+1} \\ U^{n} \end{bmatrix}. \tag{3.4}$$

Letting vector $U_n = (U^{n+s-1}, U^{n+s-2}, \ldots, U^n)^T$, equation (3.4) can be expressed as

$$U_{n+1} = A_n U_n, \tag{3.5}$$

where $A_n$, given in (3.4), is known as the companion matrix. It can be shown that the roots of the polynomial $\rho_n(z)$ are the eigenvalues of the companion matrix $A_n$. Similar to $A_n$, the companion matrix $A_n^*$ associated with the divided polynomial $\rho_n^*$ in (3.3) is given by

$$A_n^* = \begin{bmatrix} -\lambda_{s-2,n}^* & -\lambda_{s-3,n}^* & \cdots & . & -\lambda_{0,n}^* \\ 1 & 0 & \cdots & . & 0 \\ & 1 & \cdots & . & 0 \\ & & \ddots & \vdots & \vdots \\ & & & 1 & 0 \end{bmatrix}. \tag{3.6}$$

Having defined the companion matrices, $A_n$, we are in a position to give a precise definition of stability. The following definition is based on Definition 5.4 in [8], p. 403:

**Definition 3.1.** *VSIMEX schemes (2.1) are called zero-stable whenever*

$$\|A_{n+m}A_{n+m-1}\cdots A_{n+1}A_n\| < M$$

*for all $n$ and $m \geq 0$ and $M$ is a constant, independent of $n$ and $m$.*

We are also interested in exploring the relationship between $A_n$ and $A_n^*$ defined above. This can be easily done by introducing matrix $T$ and its inverse $T^{-1}$ [6, 8]

$$T = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ & 1 & 1 & \dots & 1 \\ & & 1 & \dots & 1 \\ & 0 & & \ddots & \vdots \\ & & & & 1 \end{bmatrix}, \quad T^{-1} = \begin{bmatrix} 1 & -1 & & & 0 \\ & 1 & -1 & & \\ & & 1 & \ddots & \\ & 0 & & \ddots & -1 \\ & & & & 1 \end{bmatrix}$$

with dimensions $s \times s$. A simple calculation leads to

$$T^{-1}A_nT = \begin{bmatrix} A_n^* & 0 \\ e_{s-1}^T & 1 \end{bmatrix},$$

where block matrix $A_n^*$ is defined in (3.6) with dimension $(s-1)\times(s-1)$; $e_{s-1}^T = [0, \cdots, 0, 1]$ with dimension $1 \times (s-1)$; the zero block matrix has dimension $(s-1)\times 1$ and 1 is just a scalar.

The key result from this section is a theorem which is based on Theorem 4.(13) in [6] and Theorem 5.6 in [8]:

**Theorem 3.1.** *The order-$s$, $s$-step VSIMEX scheme (2.1) is zero-stable if and only if the following two conditions are satisfied for all $n$ and $m \geq 0$*

$$\begin{array}{ll} (a) & \left\|A_{n+m}^*\cdots A_{n+1}^*A_n^*\right\| < M_1 \\ (b) & \left\|e_{s-1}^T\sum_{j=n}^{n+m}\left(\prod_{i=n}^{j-1}A_i^*\right)\right\| < M_2, \end{array} \tag{3.7}$$

*where $M_1, M_2$ are constants, independent of $m$ and $n$.*

An important attribute of Theorem 3.1 lies in that the dimension of the matrices under consideration is reduced by one [8]. This is especially useful for the zero-stability analysis of second-order, two-step VSIMEX methods which is provided in Section 3.2.

From the above discussion, we have the following observations:

1. The purpose of introducing the companion matrix $A_n$ in (3.4) is to conveniently set up the framework for the zero-stability analysis. By the recurrence relation (3.5), we can easily get
$$U_{n+m+1} = A_{n+m}A_{n+m-1}\cdots A_{n+1}A_nU_n. \tag{3.8}$$
To measure whether computational errors (or perturbations) in $U_n$ are under control as the computation proceeds and the step-size approaches zero, we simply require that $\|A_{n+m}\cdots A_{n+1}A_n\|$ is bounded for some suitably chosen matrix norm. This naturally leads to the Definition 3.1.

2. In general, the companion matrix $A_n^*$ involves the step-size ratios $\omega_j, j = n+1, \dots, n+s-1$. The variable coefficients $\lambda_{j,n}^*, j = 0, \dots, s-2$ are functions of step-size ratios, hence Theorem 3.1 will impose restrictions on these values $\omega_j$ in order to ensure zero-stability.

We now proceed to use Definition 3.1 and Theorem 3.1 to derive the zero-stability restrictions on step-size ratios for our first and second order VSIMEX schemes.

### 3.1. First-Order VSIMEX Schemes

To test the zero-stability property, apply (2.5) to $\dot{u} = 0$. This yields

$$U^{n+1} - U^n = 0$$

and the companion matrix $A_n$ defined in Section 3 is simply

$$A_n = [1]$$

for all $n$. Thus Definition 3.1 is automatically satisfied for schemes (2.5), which implies that first-order VSIMEX schemes (2.5) are zero-stable for any step-size sequence.

### 3.2. Second-Order VSIMEX Schemes

Applying second-order, two-step VSIMEX schemes (2.7) to $\dot{u} = 0$ and scaling so that the coefficient of $U^{n+2}$ equals 1 yields

$$U^{n+2} + \lambda_{1,n} U^{n+1} + \lambda_{0,n} U^n = 0,$$

where

$$\lambda_{1,n} = \frac{(1+\omega_{n+1})[(1-2\gamma)\omega_{n+1} - 1]}{1 + 2\gamma\omega_{n+1}}, \qquad \lambda_{0,n} = \frac{(2\gamma-1)\omega_{n+1}^2}{1+2\gamma\omega_{n+1}}, \qquad n = 0, 1, 2, \ldots, N-2,$$

and $N$ is the number of total nodes in time interval $[\,0, T\,]$. In this case, the companion matrices $A_n$ and $A_n^*$ are given by

$$A_n = \begin{bmatrix} -\lambda_{1,n} & -\lambda_{0,n} \\ 1 & 0 \end{bmatrix}, \quad A_n^* = \begin{bmatrix} -\lambda_{0,n}^* \end{bmatrix},$$

where $\lambda_{0,n}^* = -\lambda_{0,n}$ so that Theorem 3.1 takes a simple form, which is presented in Theorem 3.2.

**Theorem 3.2.** *The second-order, two-step VSIMEX scheme (2.7) is zero-stable if and only if the following two conditions are satisfied for all $n$ and $m \geq 0$,*

$$\begin{array}{ll} (a) & \sup\left(|\lambda_{0,n+m} \cdots \lambda_{0,n+1}\lambda_{0,n}|\right) < M_1 \\ (b) & \sup\left(|1 + \lambda_{0,n} + \lambda_{0,n}\lambda_{0,n+1} + \cdots + \lambda_{0,n}\lambda_{0,n+1}\cdots\lambda_{0,n+m-1}|\right) < M_2. \end{array}$$

*where $M_1$ and $M_2$ are constants, independent of $m$ and $n$.*

We observe that if there exists a positive number $q < 1$ such that $|\lambda_{0,i}| \leq q < 1$ for all $i = n, n+1, \ldots, n+m$ then conditions $(a)$ and $(b)$ in Theorem 3.2 will be satisfied and imply zero-stability.

Recall

$$\lambda_{0,i} = \frac{(2\gamma-1)\omega_{i+1}^2}{1+2\gamma\omega_{i+1}}.$$

Solving for $|\lambda_{0,i}| \leq q$, we obtain the following analytical results on the step-size ratio constraints which ensure zero-stability of second-order VSIMEX schemes.

**Corollary 3.1.** *Consider the family of second-order, two-step VSIMEX schemes (2.7) having two parameters $(\gamma, c)$ with $0 \le \gamma \le 1$. If there exists a positive number $q < 1$ such that*

$$\begin{cases} 0 < \omega_{i+1} \le \frac{q\gamma + \sqrt{q^2\gamma^2 + q(1-2\gamma)}}{1-2\gamma} & \texttt{if} \quad 0 \le \gamma < \frac{1}{2} \\ \\ 0 < \omega_{i+1} \le \frac{q\gamma + \sqrt{q^2\gamma^2 + q(2\gamma-1)}}{2\gamma-1} & \texttt{if} \quad \frac{1}{2} < \gamma \le 1 \end{cases}$$

*for all $i > 0$, then the underlying VSIMEX scheme is zero-stable.*

In particular, this corollary implies VSCNAB (2.9), modified VSCNAB (2.10) and other VSIMEX schemes with $\gamma = \frac{1}{2}$ are zero-stable for any step-size sequence since $\lambda_{0,i} = 0$ for every $i = 1, 2, \ldots, N$. All other schemes from the family of second order VSIMEX schemes will have some upper bound on the step-size ratio. For example, the VSCNLF and VSSBDF schemes are zero stable provided the step-size ratios satisfy $\omega_i \le \sqrt{q} < 1$ and $\omega_i \le q + \sqrt{q^2 + q} < 1 + \sqrt{2}$, respectively. This last result for VSSBDF has been known for some time in the context of BDF schemes, and appears in a paper by Grigorieff [6].

### 3.3. Third- and Fourth- Order VSSBDF Schemes

For third- and higher- order schemes, it is also possible to apply Theorem (3.1) to obtain restrictions on the step-size ratios. This analysis was carried out by Grigorieff [6] for the BDF schemes and Wang [12] for a variety of third-order VSIMEX schemes. Table 3.1 gives the upper ($B$) and lower ($b$) bounds on the step-size ratio obtained in [12] for a variety of third-order VSIMEX schemes from our three parameter family (2.12).

For three- and four- step methods it has been found that sharper bounds on the step-size ratios can be obtained by introducing suitable elliptic type norms into the analysis, as was carried out in [4] for the crucial case of BDF schemes. One implication of this work is that the BDF3 and BDF4 schemes are zero stable provided the step-size ratios satisfy $\omega_i \le 1.476$ and $\omega_i \le 1.101$, respectively. Still sharper bounds for the BDF3 scheme have been found by Guglielmi and Zennaro [7] using a spectral radius approach. They find that zero stability is ensured for a slightly relaxed bound of $\omega_i \le 1.501$.

Of course, zero stability for BDF schemes also implies zero stability for VSSBDF schemes. Thus VSSBDF3 is zero stable for step-size ratios $\omega_i \le 1.501$ and VSSBDF4 is zero stable for step-size ratios $\omega_i \le 1.101$.

Note that it has been stated that bounds of this type "are surely unrealistic, since all pathological step-size variations are admitted" [8]. Indeed, our numerical experiments reported below and those reported in [12] occasionally violated these time-stepping restrictions without triggering an obvious numerical instability. See also [4,8] for further discussion and analysis on the subject of zero stability of multistep schemes.

## 4. Numerical Experiments

In this section, we carry out numerical experiments which verify that the expected orders of convergence of our various VSIMEX schemes are achieved. Our test problem is the Burgers' equation

$$u_t + uu_x = \lambda u_{xx}, \tag{4.1}$$

Table 3.1: Ranges of zero-stable step-size ratios $[b, B]$ for various third-order, three-step VSIMEX schemes as found in [12].

| $(\gamma, \theta, c)$ | $b$ | $B$ |
|---|---|---|
| (0, -2.036, -0.876) | 0.7010 | 1.2111 |
| (0.5, -1.25, -0.52) | 0.5818 | 1.1892 |
| (0.75, -0.43, -0.17) | 0.7343 | 1.1836 |
| VSSBDF3 (1, 0, 0) | 0.8351 | 1.1273 |
| (1.5, 3.04, 1.26) | 0.7565 | 1.1612 |
| ($\sqrt{3}$, 5.075, 2.105) | 0.7321 | 1.1689 |
| (1.75, 5.14, 2.13) | 0.7361 | 1.1674 |
| (2, 7.72, 3.2) | 0.7166 | 1.1728 |
| (2.5, 14.02, 5.81) | 0.6900 | 1.1789 |

subject to periodic boundary conditions on the interval [-1, 1] and initial conditions

$$u(x, 0) = \sin(\pi x), \tag{4.2}$$

where $\lambda = \frac{1}{10}$ is a constant coefficient.

Methods for determining starting values are chosen according to the order of the underlying VSIMEX method. For second-order, two-step VSIMEX schemes, we use the first-order SBDF1 method with a very small temporal step-size. For the third-order, three-step VSIMEX case, we use the third-order implicit-explicit Runge-Kutta method IMEX RK(3,4,3) presented in [3]. This method applies a third-order, three-stage diagonally-implicit Runge-Kutta (DIRK) method for the stiff term, and a third-order, four-stage explicit Runge-Kutta (ERK) method for the nonstiff term. Finally, for the fourth-order, four-step VSIMEX case, we use the fourth-order implicit-explicit additive Runge-Kutta method ARK4(3)6L[2] presented in [9]. This method applies a fourth-order, six-stage stiffly-accurate, explicit, singly diagonally implicit Runge-Kutta (ESDIRK) method for the stiff term, and a fourth-order, six-stage ERK method for the nonstiff term.

In the following experiments, we test several second-order, two-step VSIMEX schemes: VSCNLF, VSCNAB, modified VSCNAB and VSSBDF2. For third- and fourth- order, we only consider the VSSBDF3 and VSSBDF4 schemes.

### 4.1. Second-Order VSIMEX Schemes

Consider the test problem (4.1)-(4.2) where the spatial derivatives $u_x$ and $u_{xx}$ have been approximated by standard second-order central differences:

$$\begin{aligned} u_x(x_j, t_n) &= \frac{U^n_{j+1} - U^n_{j-1}}{2\Delta x} + O((\Delta x)^2), \\ u_{xx}(x_j, t_n) &= \frac{U^n_{j+1} - 2U^n_j + U^n_{j-1}}{(\Delta x)^2} + O((\Delta x)^2). \end{aligned}$$

In all experiments, we compute the solution $u$ to time $t = 2$ and fix $\Delta x = \frac{1}{2500}$.

To choose a variable time step-size, we first break the interval [0, 2] into 5 subintervals with an equal length of 0.4, then split each subinterval into smaller subintervals of different sizes. For example, subinterval [0, 0.4] is divided by 6, [0.4, 0.8] by 4, etc., as in partitioning scheme no. 2 in Table 4.1.

Table 4.1: Partitioning schemes for the time interval [ 0, 2 ] (total nodes=25)

| Scheme No. | $[0, 0.4]$ | $[0.4, 0.8]$ | $[0.8, 1.2]$ | $[1.2, 1.6]$ | $[1.6, 2.0]$ |
|---|---|---|---|---|---|
| 1 | 8 | 7 | 3 | 3 | 4 |
| 2 | 6 | 4 | 3 | 7 | 5 |
| 3 | 3 | 3 | 4 | 7 | 8 |
| 4 | 1 | 1 | 5 | 8 | 10 |
| 5 | 3 | 7 | 2 | 5 | 8 |

Table 4.2: Partitioning schemes for the time interval [ 0, 2 ] (total nodes=50)

| Scheme No. | $[0, 0.4]$ | $[0.4, 0.8]$ | $[0.8, 1.2]$ | $[1.2, 1.6]$ | $[1.6, 2.0]$ |
|---|---|---|---|---|---|
| 1 | 16 | 14 | 6 | 6 | 8 |
| 2 | 12 | 8 | 6 | 14 | 10 |
| 3 | 6 | 6 | 8 | 14 | 16 |
| 4 | 2 | 2 | 10 | 16 | 20 |
| 5 | 6 | 14 | 4 | 10 | 16 |

For all partitions, our coarsest temporal grid takes 25 nodes over the time interval [ 0, 2.0 ], as shown in Table 4.1. Finer grids are obtained by doubling the nodes in time, while keeping the ratios of nodes between consecutive subintervals unchanged. This partition pattern is continued until we have 800 nodes in time. E.g. for 50 nodes, we partition the time interval [ 0, 2.0 ] according to Table 4.2.

To assess the quality of the solution, we compute the maximum norm of the absolute error, i.e., $\|U_j^N - u(x_j, 2)\|_\infty$ (which we write simply as $\|U - u_e\|_\infty$, where $u_e(x) \equiv u(x, 2)$). A high resolution solution is used to approximate the exact solution $u_e$. This high resolution result is obtained using the SBDF3 scheme with $\Delta t = \frac{1}{500}$, i.e. 1000 nodes in the time interval [ 0, 2 ].

Table 4.3 summarizes the computational results for a variety of second-order, two-step VSIMEX schemes as well as the results from their counterpart constant step-size (IMEX) versions. A clear second-order convergence rate is achieved for all the VSIMEX and partitioning schemes tested. Note that for three of the four schemes tested (VSCNLF, modified VSCNAB and VSSBDF) the smallest error was obtained using a non-uniform step-size, specifically partitioning scheme no. 2.

Introducing variable sized steps into IMEX schemes often has little impact on the amount of computational work per step, since most of the work typically comes from evaluating the explicit term, or solving the implicit equations rather than evaluating the coefficients of the scheme. For example, consider the Burgers' equation example with partitioning scheme no. 2. and 800 nodes in time. Using VSSBDF2, the CPU time[1)] for the calculation is 3.68 seconds (and the error is 4.155e-7). Applying the constant step-size counterpart of this scheme (SBDF2) requires 3.60 seconds to carry out the same calculation (but in this case the error is 9.117e-7). Assuming a quadratic error form, as is observed in Table 4.3, we see that the step-size for SBDF2 would need to be decreased by about a third to achieve the same error tolerance as VSSBDF2 with Partition 2.

[1)] The computer program was written in Fortran 77, and executed in Toshiba Satellite 1110 1.8GHz Celeron notebook computer (operating system: Fedora Core 1 Linux).

Note that while savings of about 30% were observed in this simple example, we anticipate that much larger savings may be attainable in problems with widely varying time-scales if error control strategies (see, eg, AUTO [5], COLSYS [1]) are combined with VSIMEX schemes.

### 4.2. Third- and Fourth- Order VSIMEX Schemes

Similar to the second-order case, we consider the test problem (4.1)-(4.2) and compute the solution $u$ to time $t = 2$. For the spatial discretization, however, we apply fourth-order accurate finite difference approximations of $u_x$ and $u_{xx}$, i.e. we use the 5-point formulas

$$\begin{aligned} u_x(x_j, t_n) &= \frac{1}{12\Delta x}[U^n_{j-2} - 8U^n_{j-1} + 8U^n_{j+1} - U^n_{j+2}] + O((\Delta x)^4), \\ u_{xx}(x_j, t_n) &= -\frac{1}{12(\Delta x)^2}[U^n_{j-2} - 16U^n_{j-1} + 30U^n_j - 16U^n_{j+1} + U^n_{j+2}] + O((\Delta x)^4). \end{aligned}$$

Our approximations using the VSSBDF3 scheme consider a fixed $\Delta x = \frac{1}{250}$. The exact solution is approximated using the SBDF3 scheme with $\Delta t = \frac{1}{500}$, i.e. 1000 nodes in the time interval $[\,0,\, 2\,]$. For the VSSBDF4 scheme, we choose $\Delta x = \frac{1}{350}$. In this case, the approximation to the exact solution is computed using the SBDF4 scheme with $\Delta t = \frac{1}{500}$.

To vary the time step-size, we follow the exact same partitioning methods used for our second-order schemes. Also, we use the maximum norm $\|U^N_j - u(x_j, 2)\|_\infty$ to measure the computational error.

We summarize the computational results for VSSBDF3 in Table 4.4 and for VSSBDF4 in Table 4.5. The corresponding absolute errors for VSSBDF3 and VSSBDF4 are plotted with respect to the number of total nodes in Figs. 4.1 and 4.2. These data and figures indicate that for all partitioning schemes considered, and both VSSBDF schemes, the expected convergence rates are achieved. Furthermore, a significant improvement in the errors can be obtained by selecting a variable time-stepping strategy (eg, partitioning scheme no. 1) over the constant step-size version.

## 5. Summary

In this paper, we construct and study a variety of new variable step-size IMEX linear multistep schemes up to fourth-order. All our VSIMEX schemes are order-$s$, $s$-step linear multistep methods.

First-order, one-step IMEX schemes are also VSIMEX schemes. A family of such schemes with one free parameter is given. The family of second-order, two-step VSIMEX schemes with two free parameters is derived. Included in this family of schemes are VSSBDF2 (2.8), VSCNAB (2.9), Modified VSCNAB (2.10) and VSCNLF (2.11), whose corresponding IMEX schemes are popular in practice. A particular parameterization of third-order, three-step VSIMEX schemes is also provided, which admits three free parameters. Our analysis and numerics focus on VSSBDF3 (2.14) because of its suitability for treating stiff terms and because of the popularity of the corresponding IMEX scheme. For fourth-order, we give the four-step VSIMEX scheme, VSSBDF4 (2.15). In the constant step-size case, this scheme reduces to the popular SBDF4 scheme, which is also known for its good treatment of stiff, dissipative terms.

The zero-stability of VSIMEX schemes is also considered. Zero stability imposes restrictions on the step-size variations required to ensure VSIMEX schemes remain stable as step-sizes approach zero. Based on this analysis, analytical results on restrictions of the step-size ratios

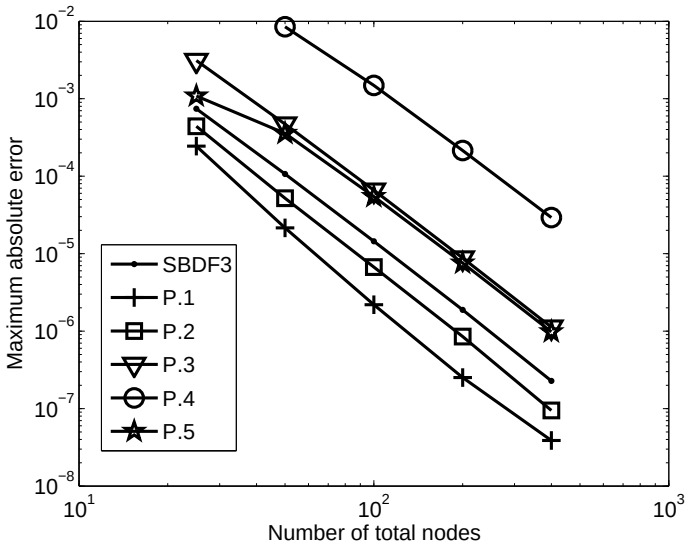


Fig. 4.1. Absolute errors for the VSSBDF3 scheme in solving the Burgers' equation. The slopes are approximately -3.0 (using logarithmic scales), which indicates third-order convergence.

for general second-order VSIMEX schemes are obtained and presented (see Corollary 3.1). Zero stability results for the third- and fourth- order VSSBDF schemes are also reviewed in this section.

Numerical experiments for Burgers' equation are carried out using various IMEX and VSIMEX schemes. In our tests, the expected orders of convergence for VSIMEX schemes are achieved and accurate numerical solutions are obtained. It is demonstrated that VSIMEX schemes can give improved accuracy over classical IMEX schemes when variable step-sizes are suitably chosen in solving Burgers' equation. For example, when the time-stepping partitioning scheme No.1 (see Tables 4.1 and Table 4.2) is chosen, the error declines by 90% when VSSBDF4 is used instead of its IMEX counterpart, SBDF4.

For future work, it would be interesting to combine error control strategies (see, eg, [1, 5]) with VSIMEX schemes. We anticipate that such combinations could give much more pronounced efficiency gains in problems where different time-scales arise.

**Acknowledgments.** The work of the first author was partially supported by an NSERC Canada Postgraduate Scholarship. The work of the second author was partially supported by a grant from NSERC Canada. The first author thanks Dr. Wentao Sun for his many helpful discussions and suggestions.

## References

[1] U.M. Ascher, J. Christiansen, and R.D. Russell, Algorithm COLYSYS: collocation software for boundary value ODE's, *ACM Trans. on Math. Software*, **7** (1981), 223–229.

[2] U.M. Ascher, S.J. Ruuth, and B.T.R. Wetton, Implicit-explicit methods for time-dependent partial differential equations, *SIAM J. Numer. Anal.*, **32** (1995), 797–823.

[3] U.M. Ascher, S.J. Ruuth, and R. Spiteri, Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations, *Appl. Numer. Math.*, **25** (1997), 151–167.

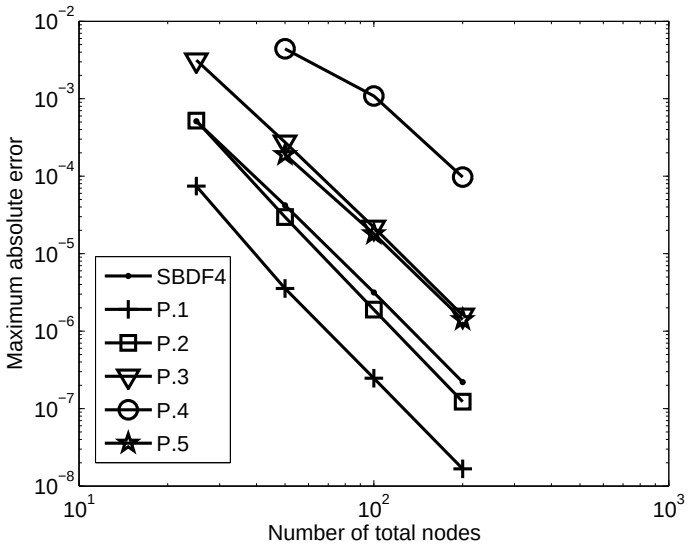


Fig. 4.2. Absolute errors for the VSSBDF4 scheme in solving the Burgers' equation. The slopes are approximately -4.0 (using logarithmic scales), which indicates fourth-order convergence.

[4] M. Calvo and T. Grande and R.D. Grigorieff, On the zero stability of the variable order variable stepsize BDF-formulas, *Numer. Math.*, **57** (1990), 39–50.

[5] E.J. Doedel, AUTO: A program for the automatic bifurcation analysis of autonomous systems, *Cong. Num.*, **30** (1981), 265–284.

[6] R.D. Grigorieff, Stability of multistep-methods on variable grids, *Numer. Math.*, **42** (1983), 359–377.

[7] N. Guglielmi and M. Zennaro, On the zero-stability of variable stepsize multistep methods: the spectral radius approach, *Numer. Math.*, **88** (2001), 445–458.

[8] E. Hairer, S.P. Norsett, and G. Wanner, Solving Ordinary Differential Equation I, Springer-Verlag, 1993.

[9] C. Kennedy and M. Carpenter, Additive Runge-Kutta schemes for convection-diffusion-reaction equations, *Appl. Numer. Math.*, **44** (2003),139–181.

[10] J.D. Lambert, Numerical Methods for Ordinary Differential Systems - the Initial Value Problem, John Wiley & Sons Ltd. 2000.

[11] A. Nordsieck, On the numerical integration of ordinary differential equations, *Math. Comp.*, **16** (1962), 22–49.

[12] D. Wang, Variable step-size implicit-explicit linear multistep methods for time-dependent PDEs, M.Sc. Thesis, Department of Mathematics, Simon Fraser University, Burnaby, Canada, 2005.

Table 4.3: Numerical results for Burgers' equation using various second-order IMEX and VSIMEX schemes.

| | Nodes | Constant Stepsize | | Partition 1 | | Partition 2 | | Partition 3 | | Partition 4 | | Partition 5 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Scheme | in time | $\|U-u_e\|_\infty$ | Order | $\|U-u_e\|_\infty$ | Order | $\|U-u_e\|_\infty$ | Order | $\|U-u_e\|_\infty$ | Order | $\|U-u_e\|_\infty$ | Order | $\|U-u_e\|_\infty$ | Order |
| CNLF | 25 | 9.359e-4 | | 1.004e-3 | | 7.908e-4 | | 1.625e-3 | | 1.662e-2 | | 1.795e-3 | |
| | 50 | 2.356e-4 | 1.990 | 2.383e-4 | 2.075 | 2.097e-4 | 1.915 | 4.590e-4 | 1.824 | 2.292e-3 | 2.859 | 5.309e-4 | 1.757 |
| | 100 | 6.151e-5 | 1.937 | 6.016e-5 | 1.986 | 5.468e-5 | 1.939 | 1.265e-4 | 1.859 | 8.584e-4 | 1.417 | 1.400e-4 | 1.923 |
| | 200 | 1.571e-5 | 1.969 | 1.509e-5 | 1.995 | 1.393e-5 | 1.973 | 3.326e-5 | 1.927 | 2.490e-4 | 1.786 | 3.609e-5 | 1.955 |
| | 400 | 3.950e-6 | 1.992 | 3.761e-6 | 2.005 | 3.496e-6 | 1.995 | 8.512e-6 | 1.966 | 6.742e-5 | 1.885 | 9.157e-6 | 1.979 |
| | 800 | 9.704e-7 | 2.025 | 9.200e-7 | 2.031 | 8.557e-7 | 2.030 | 2.133e-6 | 1.997 | 1.754e-5 | 1.942 | 2.287e-6 | 2.002 |
| CNAB | 25 | 1.774e-4 | | 5.345e-4 | | 4.218e-4 | | 3.352e-4 | | 1.181e-2 | | 5.041e-4 | |
| | 50 | 4.904e-5 | 1.855 | 1.232e-4 | 2.117 | 9.831e-5 | 2.101 | 9.918e-5 | 1.757 | 4.570e-4 | 4.692 | 8.777e-5 | 2.522 |
| | 100 | 1.309e-5 | 1.905 | 2.945e-5 | 2.065 | 2.336e-5 | 2.073 | 2.850e-5 | 1.799 | 2.755e-4 | 0.730 | 2.500e-5 | 1.812 |
| | 200 | 3.382e-6 | 1.953 | 7.203e-6 | 2.031 | 5.686e-6 | 2.039 | 7.700e-6 | 1.888 | 7.849e-5 | 1.811 | 6.740e-6 | 1.891 |
| | 400 | 8.445e-7 | 2.002 | 1.796e-6 | 2.003 | 1.418e-6 | 2.004 | 1.987e-6 | 1.954 | 2.128e-5 | 1.883 | 1.737e-6 | 1.956 |
| | 800 | 1.955e-7 | 2.111 | 4.644e-7 | 1.952 | 3.708e-7 | 1.935 | 4.857e-7 | 2.033 | 5.545e-6 | 1.940 | 4.232e-7 | 2.038 |
| MCNAB | 25 | 3.431e-4 | | 4.315e-4 | | 2.870e-4 | | 6.688e-4 | | 1.312e-2 | | 8.674e-4 | |
| | 50 | 9.243e-5 | 1.892 | 9.690e-5 | 2.155 | 6.393e-5 | 2.166 | 1.920e-4 | 1.801 | 1.251e-3 | 3.391 | 1.835e-4 | 2.240 |
| | 100 | 2.423e-5 | 1.932 | 2.283e-5 | 2.085 | 1.474e-5 | 2.117 | 5.283e-5 | 1.862 | 4.552e-4 | 1.458 | 5.052e-5 | 1.861 |
| | 200 | 6.201e-6 | 1.966 | 5.546e-6 | 2.042 | 3.528e-6 | 2.063 | 1.394e-5 | 1.922 | 1.265e-4 | 1.848 | 1.334e-5 | 1.921 |
| | 400 | 1.552e-6 | 1.998 | 1.380e-6 | 2.006 | 8.762e-7 | 2.010 | 3.567e-6 | 1.966 | 3.377e-5 | 1.905 | 3.416e-6 | 1.966 |
| | 800 | 3.707e-7 | 2.066 | 3.588e-7 | 1.944 | 2.331e-7 | 1.910 | 8.827e-7 | 2.015 | 8.736e-6 | 1.951 | 8.453e-7 | 2.015 |
| SBDF | 25 | 9.526e-4 | | 7.245e-4 | | 4.364e-4 | | 2.130e-3 | | 1.707e-2 | | 2.012e-3 | |
| | 50 | 2.370e-4 | 2.007 | 1.679e-4 | 2.109 | 1.079e-4 | 2.015 | 5.303e-4 | 2.006 | 5.471e-3 | 1.642 | 5.199e-4 | 1.953 |
| | 100 | 5.955e-5 | 1.993 | 4.103e-5 | 2.033 | 2.735e-5 | 1.981 | 1.337e-4 | 1.987 | 1.253e-3 | 2.126 | 1.320e-4 | 1.977 |
| | 200 | 1.494e-5 | 1.995 | 1.015e-5 | 2.015 | 6.914e-6 | 1.984 | 3.375e-5 | 1.986 | 3.135e-4 | 1.999 | 3.349e-5 | 1.979 |
| | 400 | 3.725e-6 | 2.004 | 2.513e-6 | 2.015 | 1.725e-6 | 2.003 | 8.474e-6 | 1.994 | 7.866e-5 | 1.995 | 8.424e-6 | 1.991 |
| | 800 | 9.117e-7 | 2.031 | 6.102e-7 | 2.042 | 4.155e-7 | 2.054 | 2.104e-6 | 2.010 | 1.974e-5 | 1.994 | 2.093e-6 | 2.009 |

Table 4.4: Numerical results for Burgers' equation using the VSSBDF3 scheme.

| Partition No. | Nodes in time | $\|U - u_e\|_\infty$ | Order |
|---|---|---|---|
| Constant step-size | 25 | 7.418e-4 | |
| | 50 | 1.066e-4 | 2.80 |
| | 100 | 1.447e-5 | 2.88 |
| | 200 | 1.881e-6 | 2.94 |
| | 400 | 2.273e-7 | 3.05 |
| 1 | 25 | 2.445e-4 | |
| | 50 | 2.152e-5 | 3.51 |
| | 100 | 2.191e-6 | 3.30 |
| | 200 | 2.514e-7 | 3.12 |
| | 400 | 3.874e-8 | 2.70 |
| 2 | 25 | 4.403e-4 | |
| | 50 | 5.201e-5 | 3.08 |
| | 100 | 6.702e-6 | 2.96 |
| | 200 | 8.506e-7 | 2.98 |
| | 400 | 9.471e-8 | 3.17 |
| 3 | 25 | 3.117e-3 | |
| | 50 | 4.711e-4 | 2.73 |
| | 100 | 6.586e-5 | 2.84 |
| | 200 | 8.790e-6 | 2.91 |
| | 400 | 1.127e-6 | 2.96 |
| 4 | 25 | 1.174e-3 | |
| | 50 | 8.486e-3 | |
| | 100 | 1.484e-3 | 2.52 |
| | 200 | 2.149e-4 | 2.79 |
| | 400 | 2.928e-5 | 2.88 |
| 5 | 25 | 1.084e-3 | |
| | 50 | 3.577e-4 | 1.60 |
| | 100 | 5.460e-5 | 2.71 |
| | 200 | 7.546e-6 | 2.86 |
| | 400 | 9.794e-7 | 2.94 |

Table 4.5: Numerical results for Burgers' equation using the VSSBDF4 scheme.

| Partition No. | Nodes in time | $\|U - u_e\|_\infty$ | Order |
|---|---|---|---|
| Constant step-size | 25 | 5.112e-4 | |
| | 50 | 4.209e-5 | 3.60 |
| | 100 | 3.160e-6 | 3.74 |
| | 200 | 2.196e-7 | 3.85 |
| 1 | 25 | 7.461e-5 | |
| | 50 | 3.556e-6 | 4.39 |
| | 100 | 2.469e-7 | 3.85 |
| | 200 | 1.667e-8 | 3.89 |
| 2 | 25 | 5.221e-4 | |
| | 50 | 2.972e-5 | 4.14 |
| | 100 | 1.898e-6 | 3.97 |
| | 200 | 1.230e-7 | 3.95 |
| 3 | 25 | 3.152e-3 | |
| | 50 | 2.739e-4 | 3.52 |
| | 100 | 2.188e-5 | 3.65 |
| | 200 | 1.601e-6 | 3.77 |
| 4 | 25 | 1.075e-4 | |
| | 50 | 4.415e-3 | |
| | 100 | 1.084e-3 | 2.03 |
| | 200 | 9.731e-5 | 3.48 |
| 5 | 25 | 6.783e-5 | |
| | 50 | 1.900e-4 | |
| | 100 | 1.806e-5 | 3.40 |
| | 200 | 1.403e-6 | 3.69 |