Journal of Computational Mathematics, Vol.26, No.6, 2008, 838–855.

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@illinois.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 December 20, 2007 / Revised version received May 15, 2008 / Accepted May 27, 2008 /

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 schemes, 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 $u(t_{n+j}), j = 0, 1, \cdots, s-1$, 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