ASTFE
American Society
of Thermal and Fluids Engineers

3rd Thermal and Fluids Engineering Conference (TFEC)
March 4–7, 2018
Fort Lauderdale, FL, USA

TFEC-2018-21966

COMBINED EXPLICIT AND IMPLICIT METHODS FOR TIME INTEGRATION IN PARTIAL DIFFERENTIAL EQUATIONS

Antonio M. Zarzur,[1,*] Haroldo F. Campos Velho,[1] Saulo R. Freitas,[2] Stephan Stephany[1]

[1]Instituto Nacional de Pesquisas Espaciais, São José dos Campos, SP, 12227-010, Brazil
[2]NASA Goddard Space Flight Center, Greenbelt, MD, 20771, USA

ABSTRACT

Numerical methods for the solution of partial differential equations are broadly categorized as either implicit or explicit. A more recent approach, denominated IMEX, takes advantage of both strategies to form a combined scheme that excels in the solution of problems containing both fast and slow time scales. In this work, a family of multistep IMEX schemes based on Adams methods is used to solve the two-dimensional viscous Burgers' equation, which is one of the fundamental equations in fluid dynamics. Two numerical experiments are proposed, where the exact solution is used to verify the results, and the accuracy of the proposed methods is compared to a fully explicit and a fully implicit implementation.

KEY WORDS: Burgers' Equation, Finite Differences, IMEX, Time Integration

1. INTRODUCTION

The computational simulation of many problems in physics and engineering is based on the numerical solution of partial differential equations (PDEs). The integration of dynamical systems that arise from these PDEs after spatial discretization can be achieved by various methods, which are traditionally classified into two broad categories: explicit and implicit. Explicit methods calculate the future state of a system using only information from previously known states, leading to simple implementations. Their numerical stability, however, is conditioned by a relation between temporal and spatial discretizations, according to the Lax equivalence theorem. Such constraints may imply in very small time steps for certain classes of problems, significantly increasing the execution time of the corresponding algorithms. Implicit methods, on the other hand, have no such restrictions, but must solve a system of equations that include both known and unknown states to find a solution to the problem. While the latter approach is unconditionally stable, it can also lead to computationally expensive systems of nonlinear equations that must be solved at each time step.

An alternative approach, denominated implicit-explicit (IMEX), combines these two methods into a single type of scheme designed to solve problems containing both fast and slow time scales. This type of problem has been thoroughly studied (such as in [1] and [2]), and their system of equations typically assume the form

$$u_t = f(u) + \nu g(u) \tag{1}$$

where subscripts mean derivative and ν is a nonnegative parameter. In Equation 1, the terms collected in $f(u)$ are on a slow time scale, and therefore not prone to cause instabilities. Because they are possibly nonlinear, the implementation of an implicit integration scheme faces performance challenges, whether from the poor performance of iterative solvers or a complex Jacobian matrix associated to the problem. It would make

[*]Corresponding Antonio M. Zarzur: mauricio.zarzur@gmail.com

sense, then, to solve this term explicitly. The terms in $g(u)$, however, are on a fast time scale, and their explicit solution may require excessively small time steps in order to maintain numerical stability. Being usually linear in nature, they can be solved implicitly without further complications.

IMEX schemes propose exactly that approach – the solution of the fast term implicitly and the slow term explicitly, resulting in a combination of previously unrelated schemes in order to optimize processing times by avoiding complex nonlinear systems and unnecessarily small time steps. The stability of such methods is still bound by the Courant-Friedrichs-Lewy (CFL) condition, but because the fast terms are treated implicitly, these conditions are less strict when compared to a fully explicit scheme. This allows for larger temporal increments, reducing the number of time steps required to complete a simulation. If a stability condition can be reached such that the increase in time step size offsets the additional cost in the computations per single time step, then the overall performance of the algorithm can be improved.

In [6], Durran and Blossey propose a new family of IMEX schemes based on Adams methods, whose stability properties make them potentially attractive choices for IMEX approximations to certain problems related to the dynamics of the atmosphere. The general formulation allows for different schemes to be chosen based on a set of numerical parameters, providing multiple options for the solution of a given problem.

This work applies that approach to solve the two-dimensional viscous Burgers' equation. Numerical experiments are performed to assess the accuracy of the methods compared to the analytical solution.

2. MULTISTEP IMEX SCHEMES BASED ON ADAMS METHODS

Multistep schemes take advantage of information from more than one known state in order to calculate the next step in the simulation. A time integration strategy is then called an s-step scheme when it uses a total of s steps (including the unknown state) in its formulation. A general formulation for a linear IMEX M+1 step approximation to Equation 1, consistent with the works of [2] and [6], may be expressed in the following form:

$$\sum_{k=-M}^{1} \alpha_k u^{n+k} = \Delta t \left[\sum_{k=-M}^{0} \beta_k f(u^{n+k}) + \nu \sum_{k=-M}^{1} \lambda_k g(u^{n+k}) \right] \tag{2}$$

Durran and Blossey construct their Adams-based multistep methods by choosing Adams schemes for which $\alpha_1 = 1$, $\alpha_0 = -1$, and $\alpha_{-1} = 0$. According to [6], the highest-order explicit Adams scheme that can be obtained using s steps is the s-step Adams-Bashforth method, while the highest-order implicit scheme is the s-step Adams-Moulton method. But because the only A-stable Adams-Moulton method does not damp high-frequency oscillations, the authors choose instead to consider implicit Adams methods with a lower order of accuracy than the best that can be achieved for a given number of time steps. These methods possess stability properties that still make them attractive for IMEX approximations.

The integration strategy for the Adams-implicit family is given by a one-parameter equation, which takes the form:

$$g(u) \approx \frac{(1+c)}{2} g(u^{n+1}) + \frac{(1-2c)}{2} g(u^n) + \frac{c}{2} g(u^{n-1}) \tag{3}$$

The explicit integration strategy is similarly described by another one-parameter equation, given by:

$$f(u) \approx \frac{(3+b)}{2} f(u^n) - \frac{(1+2b)}{2} f(u^{n-1}) + \frac{b}{2} f(u^{n-2}) \tag{4}$$

Three combinations of such methods are proposed in [6]. Their corresponding parameters and nomenclature are shown in Table 1.

Table 1 Parameters for multistep IMEX schemes based on Adams methods

Implicit Method	**Explicit Method**	b	c	β_0	β_{-1}	β_{-2}	λ_1	λ_0	λ_{-1}
MCN	AX2+	3/8	1/8	27/16	27/8	3/16	9/16	3/8	1/16
AM2*	AX2*	1/2	1/2	7/4	-1	1/4	3/4	0	1/4
AI2*	AB3	5/6	3/2	23/12	-4/3	5/12	5/4	-1	3/4

3. APPLICATION TO TWO-DIMENSIONAL BURGERS' EQUATION

Burgers' equation is one the fundamental equations in fluid dynamics. Initially proposed by Bateman [3], who derived the steady-state solution for the one-dimensional equation, it was later used by Burgers [5] in his study of turbulence. This equation arises frequently in the mathematical modeling of problems related to fluid flows because it is capable of directly incorporating the interaction between the nonlinear convection processes and the diffusive viscous processes [7]. Because its exact solution is well-known, and because of its range of practical applications, Burgers' equation has been widely used as a model equation for testing and comparing different computational algorithms.

When considering two spatial dimensions (x, y) and two velocity components (u, v), it becomes a system of two coupled PDEs, which are essentially a simplification of the Navier-Stokes equations by removing the pressure gradient terms and continuity equation [9]:

$$u_t + uu_x + vu_y = \nu(u_{xx} + u_{yy}) \tag{5}$$

$$v_t + uv_x + vv_y = \nu(v_{xx} + v_{yy}) \tag{6}$$

One could rewrite Equation 5 and Equation 6 into a single equation in matrix form:

$$\Phi_t = f(\Phi) + \nu g(\Phi) \tag{7}$$

where Φ is the velocity vector with components u and v, and functions f and g are given by:

$$f(\Phi) = -\frac{1}{2}\nabla(\Phi \cdot \Phi) \tag{8}$$

$$g(\Phi) = \nabla^2\Phi \tag{9}$$

It is immediately apparent that Equation 7 matches the canonical IMEX form, as described by Equation 1. In this case, the term $f(\Phi)$ collects the nonlinear convective processes, while $g(\Phi)$ collects the diffusive viscous processes of the fluid flow. Therefore, Burgers' equation seems to be a suitable candidate for IMEX integration. By applying the Adams-family scheme (Equation 3 for the implicit term, and Equation 4 for the explicit term) to Equation 7, the general time stepping strategy for Burgers' equation is obtained:

$$\begin{aligned}\frac{\Phi^{n+1} - \Phi^n}{\Delta t} = &\frac{(3+b)}{2}f(\Phi^n) - \frac{(1+2b)}{2}f(\Phi^{n-1}) + \frac{b}{2}f(\Phi^{n-2}) \\ &+\nu\left[\frac{(1+c)}{2}g(\Phi^{n+1}) + \frac{(1-2c)}{2}g(\Phi^n) + \frac{c}{2}g(\Phi^{n-1})\right]\end{aligned} \tag{10}$$

where the choice of parameters b and c defines the resulting multistep IMEX scheme. From Table 1, time stepping schemes for Burgers' equation can be achieved through each combination of implicit and explicit Adams method. For example, choosing $b = \frac{3}{8}$ and $c = \frac{1}{8}$ and applying these values to Equation 10 renders the MCN-AX2+ method for Burgers' equation:

$$\frac{\Phi^{n+1} - \Phi^n}{\Delta t} = \frac{27}{16}f(\Phi^n) + \frac{27}{8}f(\Phi^{n-1}) + \frac{3}{16}f(\Phi^{n-2}) + \nu\left[\frac{9}{16}g(\Phi^{n+1}) + \frac{3}{8}g(\Phi^n) + \frac{1}{16}g(\Phi^{n-1})\right] \tag{11}$$

Choosing $b = \frac{1}{2}$ and $c = \frac{1}{2}$ renders the AM2*-AX2*+ method:

$$\frac{\Phi^{n+1} - \Phi^n}{\Delta t} = \frac{7}{4} f(\Phi^n) - f(\Phi^{n-1}) + \frac{1}{4} f(\Phi^{n-2}) + \nu \left[\frac{3}{4} g(\Phi^{n+1}) + \frac{1}{4} g(\Phi^{n-1})\right] \tag{12}$$

The method hereby called AI2*-AB3 is obtained by choosing $b = \frac{5}{6}$ and $c = \frac{3}{2}$. When applied to Equation 10, its general form is given by:

$$\frac{\Phi^{n+1} - \Phi^n}{\Delta t} = \frac{23}{12} f(\Phi^n) - \frac{4}{3} f(\Phi^{n-1}) + \frac{5}{12} f(\Phi^{n-2}) + \nu \left[\frac{5}{4} g(\Phi^{n+1}) - g(\Phi^n) + \frac{3}{4} g(\Phi^{n-1})\right] \tag{13}$$

Discretization of the spatial derivatives of u and v in the terms $f(\Phi)$ and $g(\Phi)$ can be handled by many different methods; this work focuses only on the finite differences approach, using second order approximations for the derivatives in order to complete the discretization of Burgers' equation. Accordingly, approximations for the two-dimensional spatial first derivatives are given by:

$$u_x \approx \frac{u^n_{i+1,j} - u^n_{i-1,j}}{2\Delta x} \tag{14}$$

$$u_y \approx \frac{u^n_{i,j+1} - u^n_{i,j-1}}{2\Delta y} \tag{15}$$

while the second derivatives are approximated by:

$$u_{xx} \approx \frac{u^n_{i+1,j} - 2u^n_{i,j} + u^n_{i-1,j}}{\Delta x^2} \tag{16}$$

$$u_{yy} \approx \frac{u^n_{i,j+1} - 2u^n_{i,j} + u^n_{i,j-1}}{\Delta y^2} \tag{17}$$

where $u^n_{i,j}$ denotes $u(x_i, y_j, t_n)$. This provides second order approximations for both the advection and diffusion terms.

4. NUMERICAL EXPERIMENTS

In this section, two numerical experiments previously validated in the literature were chosen to verify the accuracy of the IMEX implementations. Additionally, the results are compared to a fully explicit solution based on the Euler-forward (FTCS) method [8] and a fully implicit solution based on the Crank-Nicolson method [9].

The first test case is taken from [10], where Zhu et al. use the exact solution of the two-dimensional Burgers' equation obtained by [7] to generate initial and boundary conditions for a numerical experiment. The exact solutions for u and v are given by Equation 18 and 19, respectively:

$$u(x, y, t) = \frac{3}{4} - \frac{1}{4(1 + e^{(-t-4x+4y)/(32\nu)})} \tag{18}$$

$$v(x, y, t) = \frac{3}{4} + \frac{1}{4(1 + e^{(-t-4x+4y)/(32\nu)})} \tag{19}$$

The spatial domain is bounded by $x \in [0, 1]$, $y \in [0, 1]$; the viscosity coefficient is given by $\nu = \frac{1}{80}$. The problem is integrated from $t = 0s$ to $t = 0.5s$ using a time step of $10^{-4}s$ (i.e.: 5000 time steps). Table 2 shows the error given by the L^1 norm of u, taken at $t = 0.5s$, for each implemented method. Table 3 shows the same error metrics for v. It should be noted that, given the precision adopted in this work, Tables 2 and 3 look identical. This illusion is caused by the similarity of the initial conditions, which makes the difference between errors so small as to only show up in the 12th to 14th decimal, or in this case E-016 to E-018. In practice, the

Table 2 Error given by the L^1 norm in u from each method in the first test case

Grid size	FTCS	Crank-Nicolson	MCN-AX2+	AM2*-AX2*	AI2*-AB3
10x10	9.75542 E-004	9.74883 E-004	9.74884 E-004	9.74884 E-004	9.74884 E-004
20x20	2.38169 E-004	2.37644 E-004	2.37644 E-004	2.37644 E-004	2.37644 E-004
30x30	1.04259 E-004	1.03780 E-004	1.03780 E-004	1.03780 E-004	1.03780 E-004
40x40	5.86397 E-005	5.81134 E-005	5.81135 E-005	5.81136 E-005	5.81136 E-005
50x50	3.77086 E-005	3.71849 E-005	3.71849 E-005	3.71850 E-005	3.71850 E-005

Table 3 Error given by the L^1 norm in v from each method in the first test case

Grid size	FTCS	Crank-Nicolson	MCN-AX2+	AM2*-AX2*	AI2*-AB3
10x10	9.75542 E-004	9.74883 E-004	9.74884 E-004	9.74884 E-004	9.74884 E-004
20x20	2.38169 E-004	2.37644 E-004	2.37644 E-004	2.37644 E-004	2.37644 E-004
30x30	1.04259 E-004	1.03780 E-004	1.03780 E-004	1.03780 E-004	1.03780 E-004
40x40	5.86397 E-005	5.81134 E-005	5.81135 E-005	5.81136 E-005	5.81136 E-005
50x50	3.77086 E-005	3.71849 E-005	3.71849 E-005	3.71850 E-005	3.71850 E-005

difference is so close to machine zero that for all intents and purposes the error is considered identical. The same is true when comparing results between different IMEX schemes, although in this case the difference is smaller. For example, in Table 2, the error from AM2*-AX2* in a 50x50 grid is 3.7185015241578159E-005, while AI2*-AB3 gives 3.7185088058818792E-005. Since the difference is small, results are presented as equal.

All of the proposed IMEX methods closely match the accuracy of the fully implicit method for this problem, and show slightly higher accuracy than a purely explicit scheme. An empyrical analysis of the rate of convergence using the Big O definition in [4] shows that all methods attain second order convergence. Figures 1 and 2 show that the solutions obtained via the AI2*-AB3 method for u and v, respectively, are consistent with those obtained by [10] using a discrete Adomian decomposition method. This confirms that the proposed schemes are capable of solving the studied test case.

In [8], Kweyu et al. show via the Hopf-Cole transformation that a set of exact solutions for u and v can be

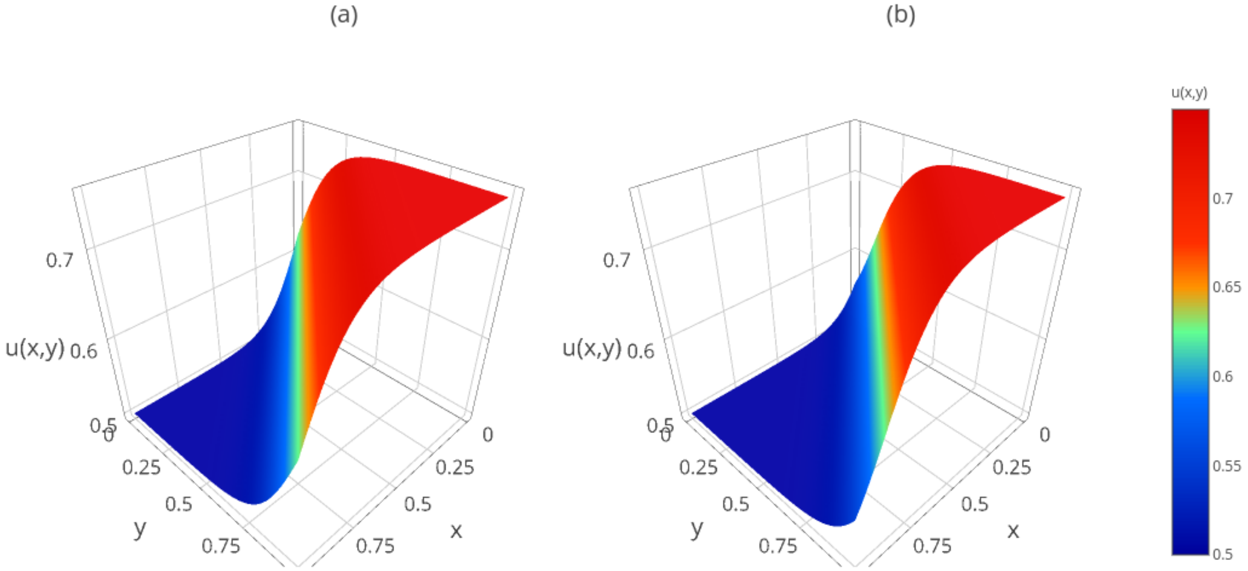

Fig. 1 Visualization setup.

Table 4 Error given by the L^1 norm in u from each method in the second test case

Grid size	FTCS	Crank-Nicolson	MCN-AX2+	AM2*-AX2*	AI2*-AB3
4x4	3.37083 E-007	3.78923 E-007	3.78788 E-007	3.78638 E-007	3.78238 E-007
8x8	5.20385 E-008	8.58507 E-008	8.57402 E-008	8.56140 E-008	8.52775 E-008
16x16	1.10676 E-008	2.09068 E-008	2.08018 E-008	2.06812 E-008	2.03598 E-008
32x32	N/A	5.15543 E-009	5.05180 E-009	4.93264 E-009	4.61497 E-009
64x64	N/A	1.24730 E-009	1.14401 E-009	1.02521 E-009	7.08456 E-010

obtained by Equations 20 and 21:

$$u(x,y,t) = \nu \left[\frac{-2y - 2\pi e^{-2\nu\pi^2 t} \sin(\pi y)(\cos(\pi x) - \sin(\pi x))}{100 + xy + e^{-2\nu\pi^2 t} \sin(\pi y)(\cos(\pi x) - \sin(\pi x))} \right] \tag{20}$$

$$v(x,y,t) = \nu \left[\frac{-2x - 2\pi e^{-2\nu\pi^2 t} \cos(\pi y)(\cos(\pi x) + \sin(\pi x))}{100 + xy + e^{-2\nu\pi^2 t} \sin(\pi y)(\cos(\pi x) - \sin(\pi x))} \right] \tag{21}$$

These equations are used to generate initial (taken at $t = 0$) and boundary conditions for the spatial domain bounded by $x \in [0,1]$, $y \in [0,1]$. However, the viscosity coefficient chosen by the authors is very low ($\nu = \frac{1}{4000}$), rendering the the contribution from the diffusive term $g(\Phi)$ is so small that its implicit treatment has little to no effect. That being the case, this work proposes a variation of their experiment, using a much higher diffusion coefficient $\nu = 0.5$ with the same initial and boundary conditions. In this case, the increased viscosity renders the FTCS scheme unstable as the spatial resolution increases, as it also factors in the CFL stability condition. The error of u and v at time $t = 1s$, given by the L^1 norm, is shown in Tables 4 and 5, respectively. A time step of $10^{-3}s$ is used for the integration (i.e.: 1000 time steps).

At a grid size of 16x16 cells, it is easily noticeable that the explicit scheme is no longer stable, as the error for v unexpectedly returns an order of magnitude smaller than the other schemes. An increase in resolution confirms the instability, as the code fails to finish the following simulations. Both the implicit and IMEX schemes show comparable accuracy for the tested spatial resolutions, but AI2*-AB3 already presents signs of

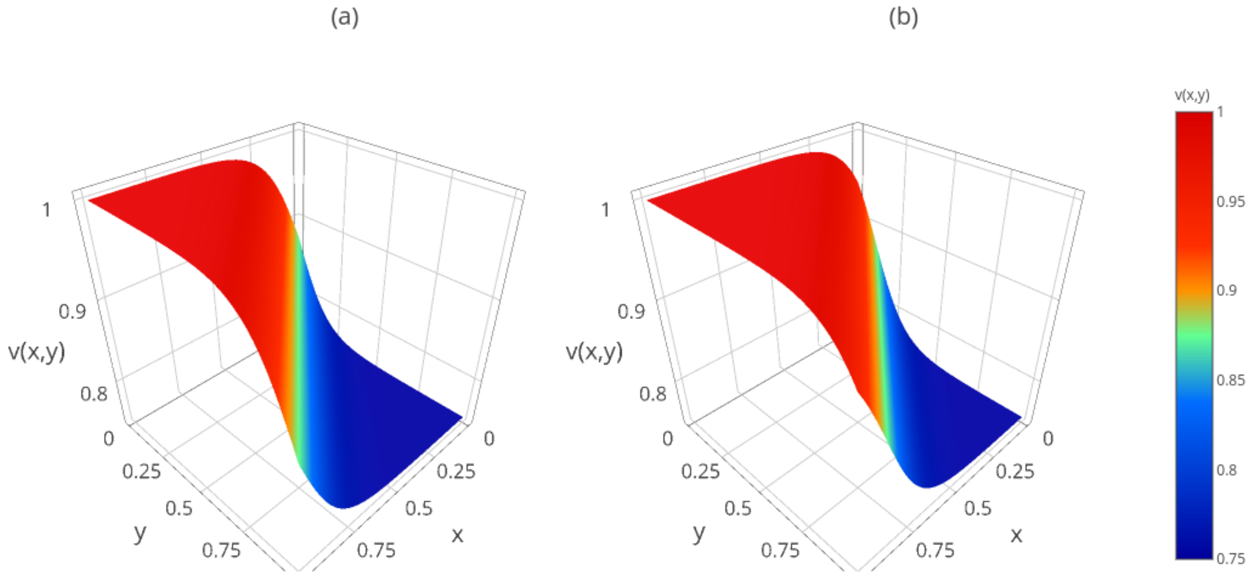

Fig. 2 Visualization setup.

Table 5 Error given by the L^1 norm in v from each method in the second test case

Grid size	FTCS	Crank-Nicolson	MCN-AX2+	AM2*-AX2*	AI2*-AB3
4x4	1.55049 E-008	1.72357 E-008	1.72329 E-008	1.72264 E-008	1.72090 E-008
8x8	2.94051 E-009	4.76637 E-009	4.76383 E-009	4.75695 E-009	4.73862 E-009
16x16	6.41358 E-010	1.21990 E-009	1.21743 E-009	1.21045 E-009	1.19182 E-009
32x32	N/A	3.03978 E-010	3.01532 E-010	2.94531 E-010	2.75867 E-010
64x64	N/A	7.37489 E-011	7.13001 E-011	6.42968 E-011	4.56236 E-011

Table 6 Comparison of execution times (in seconds) for Crank-Nicolson and IMEX schemes

Grid Size	Crank-Nicolson	IMEX	Speedup
32x32	152.376	33.194	4.59
64x64	7013.726	1387.785	5.05

instability at a grid size of 64x64 cells by giving a rate of convergence higher than second order. The MCN-AX2+ method, however, closely matches the fully implicit solution even in the more refined grids while still treating the nonlinear term explicitly.

In addition to the numerical accuracy of the solution, another important consideration when implementing time integration schemes is the processing time. Where explicit methods are easy to code and optimize, and thereby inherently faster with naive implementations, IMEX and implicit schemes require careful code optimization to ensure good performance. While this investigation is not the focus of this work, a cursory comparison of the execution times of nave implementations for the IMEX and Crank-Nicolson algorithms for the second test case is presented in Table 6 for resolutions where the FTCS scheme is no longer stable. Because the IMEX formulation shown in this work is based on numerical parameters b and c to select the resulting method, the runtimes are presented as one. The last column shows the speedup achieved by the IMEX scheme over the corresponding Crank-Nicolson one.

5. CONCLUSIONS

This work applied the Adams-family multistep IMEX schemes proposed by [6] to solve the two-dimensional Burgers' equation, which is a simplified model in fluid dynamics that can simulate both nonlinear convective processes and diffusive viscous processes. The implementations are compared to a fully explicit implementation based on Euler-forward (FTCS) and a fully implicit Crank-Nicolson solution, both studied in the literature.

Explicit schemes are simpler to implement, but are not convenient for problems with fast time scales due to the small time steps required to ensure numerical stability, which may demand a high number of calculations in order to complete the simulation in high spatial resolutions. Implicit schemes, on the other hand, are not convenient for problems with slow time scales that require the solution of a nonlinear system of equations at each time step by employing iterative schemes that may also demand high processing times. IMEX schemes can be useful for problems that have both fast and slow time scales, combining the advantages of both explicit and implicit schemes in solving each term.

Two-dimensional test cases with exact analytical solutions were chosen, both validated in the literature, in order to compare the accuracy of the solutions obtained by the numerical implementations of each proposed IMEX scheme. The results show that IMEX methods can successfully solve Burgers' equation with accuracy comparable to a fully implicit method, even while treating the nonlinear term explicitly. Furthermore, such methods remain stable at higher resolutions than the explicit scheme, constituting an interesting alternative for these cases. In particular, the MCN-AX2+ method is able to match accuracy of the Crank-Nicolson solution while remaining stable for all the tested problems.

In this work, the focus was the accuracy obtained by the different IMEX schemes in these test cases. The computational performance is still being investigated. Nevertheless, a preliminary comparison shows that IMEX schemes perform significantly faster than the Crank-Nicolson scheme while maintaining the same order of numerical accuracy in test cases that a fully explicit scheme could not solve. Code optimization is currently underway with a focus on linear system solvers and some inefficient routines from standard libraries. A more formal analysis of processing times, including comparisons with explicit schemes and analysis of stability regions to determine optimal time step sizes, is left for future work.

ACKNOWLEDGMENTS

The authors would like to thank the following Brazilian research support agencies: CAPES, CNPq, and FAPESP.

REFERENCES

[1] Ascher, U. M., Ruuth, S. J., and Spiteri, R. J., “Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations,” *Applied Numerical Mathematics*, 25, pp. 151–167, (1997).

[2] Ascher, U. M., Ruuth, S. J., and Wetton, B. T. R., “Implicit-explicit methods for time-dependent partial differential equations,” *SIAM Journal on Numerical Analysis*, 32, pp. 797–823, (1995).

[3] Bateman, H., “Some recent researches on the motion of fluids,” *Monthly Weather Review*, 43, pp. 163–170, (1915).

[4] Burden, R. L. and Faires, J. D., *Numerical Analysis, 7th ed.*, Boston: Brooks/Cole, (2001).

[5] Burgers, J. M., “A mathematical model illustrating the theory of turbulence,” *Advances in Applied Mechanics*, Elsevier, pp. 171–199, (1948).

[6] Durran, D. R. and Blossey, P. N., “Implicit-explicit multistep methods for fast-wave-slow-wave problems,” *Monthly Weather Review*, 140, pp. 1307–1325, (2012).

[7] Fletcher, C. A. J., “Generating exact solutions of the two-dimensional Burgers’ equations,” *International Journal for Numerical Methods in Fluids*, 3, pp. 213–216, (1983).

[8] Kweyu, M. C., Manyonge, W. A., Koross, A., and Ssemaganda, V., “Numerical solutions of the Burgers’ system in two dimensions under varied initial and boundary conditions,” *Applied Mathematical Sciences*, 6, pp. 5603–5615, (2012).

[9] Srivastava, V. K., Tamsir, M., Bhardwaj, U., and Sanyasiraju, Y. V. S. S., “Crank-Nicolson scheme for numerical solutions of two-dimensional coupled Burgers’ equations,” *International Journal of Scientific & Engineering Research*, 2, pp. 44–50, (2011).

[10] Zhu, H., Shu, H., and Ding, M., “Numerical solutions of two-dimensional Burgers’ equations by discrete Adomian decomposition method,” *Computers & Mathematics with Applications*, 60, pp. 840–848, (2010).