

An Efficient Backward Semi-Lagrangian Scheme for Nonlinear Advection-Diffusion Equation

Soyoon Bak, Sunyoung Bu, Philsu Kim

Abstract—In this paper, a backward semi-Lagrangian scheme combined with the second-order backward difference formula is designed to calculate the numerical solutions of nonlinear advection-diffusion equations. The primary aims of this paper are to remove any iteration process and to get an efficient algorithm with the convergence order of accuracy 2 in time. In order to achieve these objects, we use the second-order central finite difference and the B-spline approximations of degree 2 and 3 in order to approximate the diffusion term and the spatial discretization, respectively. For the temporal discretization, the second order backward difference formula is applied. To calculate the numerical solution of the starting point of the characteristic curves, we use the error correction methodology developed by the authors recently. The proposed algorithm turns out to be completely iteration free, which resolves the main weakness of the conventional backward semi-Lagrangian method. Also, the adaptability of the proposed method is indicated by numerical simulations for Burgers' equations. Throughout these numerical simulations, it is shown that the numerical results is in good agreement with the analytic solution and the present scheme offer better accuracy in comparison with other existing numerical schemes.

Keywords—Semi-Lagrangian method, Iteration free method, Nonlinear advection-diffusion equation.

I. INTRODUCTION

THE governing equation to be considered in this paper is the one-dimensional nonlinear advection-diffusion equation described by

$$u_t + f(u)u_x = \nu u_{xx} \quad (\nu > 0), \quad x \in [x_L, x_R], \quad t \in [0, T], \quad (1)$$

together with boundary and initial data imposed as follows

$$\begin{cases} u(t, x_L) = g_1(t), & u(t, x_R) = g_2(t), & t > 0, \\ u(0, x) = u^0(x), & x \in [x_L, x_R]. \end{cases} \quad (2)$$

Among many numerical techniques, the backward semi-Lagrangian method (BSLM) is one of popular strategies to solve the above problem. It consists of two essential processes as follows. Defining the characteristic curves $\pi(t, x)$ given by the differential equation

$$\frac{d\pi(t)}{dt} = f(u(t, \pi(t))), \quad (3)$$

(1) can be changed in the form

$$\begin{aligned} \frac{d}{dt}u(t, \pi(t)) &= \frac{\partial u}{\partial t}(t, \pi(t)) + \frac{\partial u}{\partial x}(t, \pi(t)) \frac{d\pi}{dt}(t) \\ &= u_t(t, \pi(t)) + f(u(t, \pi(t)))u_x(t, \pi(t)) \\ &= \nu u_{xx}(t, \pi(t)). \end{aligned} \quad (4)$$

Soyoon Bak, Sunyoung Bu, Philsu Kim are with the Department of Mathematics, Kyungpook National University, Daegu702-701 Korea (e-mail:jiya525@naver.com).

The first process is to evolve the solution along the characteristic curve, which is described by a linear simple diffusion equation along the characteristic curve with the total time derivative and the diffusion term as described by (4). The other is to find the departure points of particles moving along the characteristic curves, whose solution is described by a self-consistent nonlinear initial value problem given by (3).

Traditionally, there are two main strategies to solve the highly nonlinear initial value problems (3) and to find the departure points. One is an implicit approach [7]. The other is a substepping method of an explicit type [7]. These methods are both second-order, but it is well known that the implicit method achieves a little bit accurate result. The small difference causes a sensitive effects in the accuracy. Furthermore, the bigger the Reynolds number is, the larger the gap of the effects is between these two methods. In addition, the explicit method may work ineffectively in some special cases and the implicit scheme is much better when the velocity changes in particular (see [5]). However, the conventional second-order backward integration schemes require an iteration process such as fixed point or Newton iteration. At each time and for every spatial point, this iteration process requires an interpolation scheme of the solution at the departure points of the particles, which yields considerable computational cost.

The primary aim of this paper is to develop a BSLM that does not require such iteration steps for solving the nonlinear equation of the characteristic curves, but have such good properties that the conventional BSLMs have. These objects can be obtained by evolving the solution along the characteristic curve with the second order central difference scheme for the diffusion term and the second-order backward difference formula (BDF2) for the total time derivative. In particular, we apply the error correction techniques, which are originated in our recent development (see [3], [4]), to solve the highly nonlinear initial value problem of finding the departure points. We introduce a modified Euler's polygon to derive the error correction scheme and use the A-stable midpoint rule as the time integration for the initial value problem. The proposed algorithm turns out to be completely iteration free, which resolves the main weakness of the conventional BSLM in time integration.

This paper is organized as following. In Section II, the backward semi-Lagrangian methods are described. Section III is devoted to describe the error correction method for solving (3). Several test problems are performed in Section IV. Finally, we provide some comments and conclusion in Section V.

II. BACKWARD SEMI-LAGRANGIAN FINITE DIFFERENCE SCHEME

This section aims to describe the backward semi-Lagrangian finite difference formula for (1). Hereafter, we define $\mathbf{u}^n := [u(t_n, x_0), u(t_n, x_1), \dots, u(t_n, x_M)]^T$ for each time t_n , where x_j are given grid points, and assume that the approximation vectors $\mathbf{U}^k := [U_0^k, U_1^k, \dots, U_M^k]^T$, ($k \leq n$) for the solution vector \mathbf{u}^k are already calculated and given. Since the departure positions of the particles arriving at Eulerian grid points do not typically coincide with the grid points, the solution at the departure positions must be interpolated. Hence, the interpolation procedure is one of the key ingredients in the implementation of BSLM. Among many techniques of interpolations, we use the B-spline interpolation in this paper. We assume that the weights of the B-splines $\mathcal{P}\mathbf{u}^n(x)$ and $\mathcal{P}\mathbf{U}^n(x)$ are calculated (see [1], [2]).

Let $\pi_j(t)$ be a characteristic curve satisfying (3) arriving at the position x_j at time t_{n+1} . Applying the BDF2 and the second order central difference for temporal and spatial derivatives, respectively, (4) leads to

$$\begin{aligned}
 & -\mu u(t_{n+1}, x_{j-1}) + (1 + 2\mu)u(t_{n+1}, x_j) - \mu u(t_{n+1}, x_{j+1}) \\
 & = \frac{4}{3}u(t_n, \pi_j(t_n)) - \frac{1}{3}u(t_{n-1}, \pi_j(t_{n-1})) \\
 & + \mathcal{O}(h^3 + h\Delta x^2), \quad j = 1, 2, \dots, M - 1,
 \end{aligned}$$

where $\mu := \frac{2\nu h}{3\Delta x^2}$. For the calculation of $\pi_j(t_n)$, we will use the information of $\pi_j(t_{n-1})$ as follows. Applying the Taylor's expansion of $\pi_j(t)$ about t_{n-1} and using the known value $\pi_j(t_{n+1}) = x_j$, one may get the following expansion of $\pi_j(t)$.

$$\begin{aligned}
 \pi_j(t) & = \pi_j(t_{n-1}) + (t - t_{n-1})f(u(t_{n-1}, \pi_j(t_{n-1}))) \\
 & + \frac{(t - t_{n-1})^2}{4h^2} \left(x_j - \pi_j(t_{n-1}) - 2hf(u(t_{n-1}, \pi_j(t_{n-1}))) \right) \\
 & + \mathcal{O}(h^3).
 \end{aligned}$$

Hence, evaluating (6) at time t_n and using the B-spline interpolation \mathcal{P} , one may get an approximate value of $\pi_j(t_n)$ given by

$$\pi_j^n \approx \frac{1}{4} \left(x_j + 3\pi_j(t_{n-1}) + 2h\mathcal{P}f(\mathbf{U}(t_{n-1}, \pi_j(t_{n-1}))) \right). \quad (7)$$

The approximation scheme of $\pi_j(t_{n-1})$ will be discussed in the next section.

Now, substituting (7) into (5) and dropping the unknown truncation term in (6), we obtain the following full-discretization

$$\mathcal{A}\mathbf{U}^{n+1} = \mathbf{d}^{n+1}, \quad (8)$$

where $(M - 1) \times (M - 1)$ tridiagonal matrix $\mathcal{A} := (a_{ij})$ whose $(i, j)^{th}$ entries are defined by

$$a_{ij} := \begin{cases} 1 + 2\mu, & i = j, \\ -\mu, & |i - j| = 1, \\ 0, & \text{otherwise} \end{cases} \quad (9)$$

and

$$\mathbf{d}^{n+1} := [d_1^{n+1} + \mu u(t_{n+1}, x_0), d_2^{n+1}, d_3^{n+1}, \dots, d_{M-2}^{n+1}, d_{M-1}^{n+1} + \mu u(t_{n+1}, x_M)]^T.$$

Here,

$$d_j^{n+1} := \frac{4}{3}\mathcal{P}\mathbf{U}^n(\pi_j^n) - \frac{1}{3}\mathcal{P}\mathbf{U}^{n-1}(\pi_j^{n-1})$$

with an appropriate approximation π_j^{n-1} for $\pi_j(t_{n-1})$.

III. ERROR CORRECTION METHOD

This section focuses on finding approximation values of the characteristic curves $\pi_j(t)$ at t_{n-1} such that

$$\begin{cases} \frac{d\pi_j(t)}{dt} = f(u(t, \pi_j(t))), & t < t_{n+1} \\ \pi_j(t_{n+1}) = x_j, \end{cases} \quad (10)$$

which is a highly nonlinear equation and self-consistent due to the unknown slope function $f(u)$.

We begin this section with the construction of a modified Euler's polygon recently developed by the authors. Also, we consider the family of all the characteristic curves $\pi_j(t)$ and we regard the function $\pi_j(t)$ as a two variable function defined by $\pi(x_j, t) := \pi_j(t)$. Further, we assume that $\pi(t, x)$ is sufficiently smooth with respect to both variables t and x . For the characteristic curve $\pi_j(t)$ satisfying (10), let us consider its Taylor's expansion at t_{n+1} and the Taylor's expansion of $u(t, \pi_j(t))$ at t_n , which is given by

$$\begin{aligned}
 \pi_j(t) & = x_j + (t - t_{n+1})f(u(t_{n+1}, \pi_j(t_{n+1}))) + \mathcal{O}(h^2) \\
 & = x_j + (t - t_{n+1})f(u(t_n, \pi_j(t_n))) + \mathcal{O}(h^2).
 \end{aligned} \quad (11)$$

To replace the unknown value $\pi_j(t_n)$ in the second equation of (11), we further apply the Taylor's expansion of $\pi(t, x_j)$ about (t_n, x_{j-1}) . Then, one may approximate $\pi_j(t_n)$ as follows.

$$\pi_j(t_n) \approx p_j^n := \begin{cases} x_0 & \text{if } j = 0 \\ y_{j-1}(t_n) & \text{otherwise,} \end{cases} \quad (12)$$

where $y_j(t)$ is a modified Euler's polygon defined by

$$y_j(t) := x_j + (t - t_{n+1})\mathcal{P}f(\mathbf{U}^n(p_j^n)). \quad (13)$$

Let $\psi_j(t)$ be the difference between the analytic solution $\pi_j(t)$ of (10) and modified Euler polygon $y_j(t)$ defined by

$$\psi_j(t) := \pi_j(t) - y_j(t). \quad (14)$$

By (14) and (13), one may see that $\psi_j(t)$ satisfies a first-order ODE given by

$$\psi_j'(t) \approx f(u_\pi(t_n, y_j(t_n)))\psi_j(t) + f(u(t, y_j(t))) - \mathcal{P}f(\mathbf{U}^n(p_j^n)), \quad (15)$$

where $u_\pi(t, \pi_j(t))$ denotes the Jacobian of u . The equation (15) can be obtained by using the Taylor's expansion of $u(t, \pi_j(t))$ at $(t, y_j(t))$. By integrating both sides of equation (15) over the interval $[t_{n-1}, t_{n+1}]$ and applying the mid-point integrating rule, one may get an asymptotic formula given by

$$\begin{aligned}
 -\psi(t_{n-1}) & = 2h\psi'(t_n) \\
 & = 2h \left(f(u_\pi(t_n, y_j(t_n)))\psi_j(t_n) + f(u(t_n, y_j(t_n))) \right. \\
 & \quad \left. - \mathcal{P}f(\mathbf{U}^n(p_j^n)) \right) + \mathcal{O}(h^3).
 \end{aligned} \quad (16)$$

Further, we use the fact $\psi_j(t_n) = \frac{1}{2}(\psi_j(t_{n+1}) + \psi_j(t_{n-1})) + \mathcal{O}(h^2)$, which can be proved by the Taylor's theorem. Then, one may approximate (16) as follows.

$$(1 + hf(u_\pi(t_n, y_j(t_n))))\psi_j(t_{n-1}) \approx -2h \left(f(u(t_n, y_j(t_n))) - \mathcal{P}f(\mathbf{U}^n(p_j^n)) \right). \quad (17)$$

$$f(u(t_n, y_j(t_n))) \approx \mathcal{P}f(\mathbf{U}^n(y_j(t_n))),$$

$$f(u_\pi(t_n, y_j(t_n))) \approx J_n = \frac{1}{h} \left(\mathcal{P}f(\mathbf{U}^n(y_j(t_n))) - \mathcal{P}f(\mathbf{U}^n(y_j(t_n) - h)) \right). \quad (18)$$

Combining (17) with (18) leads to

$$\psi_j^{n-1} \approx -2h \frac{(\mathcal{P}f(\mathbf{U}^n(y_j(t_n))) - \mathcal{P}f(\mathbf{U}^n(p_j^n)))}{(1 + hJ_n)}. \quad (19)$$

Thus, one may get the approximation π_j^{n-1} of the value $\pi_j(t_{n-1})$ by

$$\pi_j^{n-1} := y_j(t_{n-1}) + \psi_j^{n-1}. \quad (20)$$

IV. NUMERICAL EXPERIMENTS

We consider the numerical solutions of Burgers' equation described by

$$\begin{cases} u_t(t, x) + u(t, x)u_x(t, x) - \nu u_{xx}(t, x) = 0, & \nu > 0, \\ u(0, x) = u_0(x), & [x_{min}, x_{max}] \\ u(t, x_{min}) = g_1(t), & u(t, x_{max}) = g_2(t), \quad t \geq 0. \end{cases}$$

To investigate the convergence of the proposed scheme, the discrete L_2 norm error, $Err_2(t)$ is measured and which is defined by

$$Err_2(t) = \left(\frac{1}{n} \sum_{i=0}^n (U_i(t) - u(t, x_i))^2 \right)^{\frac{1}{2}},$$

where $U_i(t)$, $u(t, x_i)$ are numerical and analytical solution, respectively.

The proposed scheme is compared with the semi-Lagrangian schemes combining with BDF2 [7]. For the comparisons, we use two traditional schemes for solving the highly nonlinear initial value problems (3). One is an implicit approach [7] given by

$$\begin{aligned} \pi_j^{n-1} &= x_j - 2\alpha, & \alpha &= hf(u(t_n, x_j - \alpha)), \\ \pi_j^n &= x_j - \alpha_j, \\ \alpha_j &= h \left(\frac{3}{2}f(u(t_n, x_j - \alpha_j/2)) - \frac{1}{2}f(u(t_{n-1}, x_j - \alpha_j/2)) \right). \end{aligned} \quad (21)$$

The other is a substepping method of an explicit type [7] given by

$$\begin{aligned} \pi_j^{n-1} &= x_j - 2hf(u(t_n, x_j - hf(u(t_{n-1}, x_j))), \\ \pi_j^n &= x_j - h \left(\frac{3}{2}f(u(t_n, x_j - \frac{h}{2}f(u(t_n, x_j)))) \right. \\ &\quad \left. - \frac{1}{2}f(u(t_{n-1}, x_j - \frac{h}{2}f(u(t_n, x_j)))) \right). \end{aligned} \quad (22)$$

For simplicity, we will use the following notations.

- ECM2: proposed scheme.
- BDF2+Fixed: Using (5), (21) with using fixed point iteration to solve (21).
- BDF2+Explicit: Using (5), (22).

For implicit scheme, the tolerance and maximum number of iteration for the iteration process are taken by $tol = 1.0 \times 10^{-6}$, $maxiter = 100$.

Example 1. In the first example, we consider $\nu = 1$ and use the exact solution

$$u(t, x) = \frac{2 \sin(x)}{\cos(x) + \exp(t)}, \quad (t, x) \in [0, 1] \times [0, 1].$$

The initial and boundary conditions are obtained from the exact solution.

We test the example with a viscosity $\nu = 1.0$ and compare the numerical results with those of [6]. In Table I, we use a set of parameter, $\nu = 1.0$, $\Delta x = \frac{1}{1000}$ and varying time step size h from $\frac{1}{20}$ to $\frac{1}{640}$. Also, we use the cubic B-spline interpolation. It can be seen that the proposed method has numerically second order accuracy in time and has much better performances than the results of [6].

TABLE I
COMPARISONS OF THE PROPOSED SCHEME WITH J. WANG ET AL. [6] FOR SOLVING EXAMPLE 1., WHEN FIXED $\Delta x = \frac{1}{1000}$.

h	ECM2		[6]	
	Err ₂ (1)	Rates	Err ₂ (1)	Rates
$\frac{1}{20}$	2.46e-6	-	2.00e-5	-
$\frac{1}{40}$	3.83e-7	2.68	4.14e-6	2.27
$\frac{1}{80}$	1.03e-7	1.98	1.06e-5	-1.36
$\frac{1}{160}$	2.82e-8	1.88	3.16e-5	-1.58
$\frac{1}{320}$	6.65e-9	2.08	6.91e-5	-1.13
$\frac{1}{640}$	1.64e-9	2.02	1.42e-4	-1.04

Example 2. Consider the Burgers' equation with analytic solution

$$u(t, x) = \frac{x/t}{1 + (t/\nu_0)^{\frac{1}{2}} \exp(\frac{x^2}{4\nu t})}, \quad x \in [0, 1], \quad t \geq 1,$$

where $\nu_0 = \exp(\frac{1}{8\nu})$. The initial and boundary conditions are computed from the above analytic solution.

We first examine the temporal order of convergence for the present method with the viscosity value $\nu = 5.0 \times 10^{-3}$ and the fixed spatial grid size $\Delta x = \frac{1}{1000}$ by varying time step size h from $\frac{1}{10}$ to $\frac{1}{160}$. In this numerical test, we use the quadratic B-spline interpolation. The numerical results are listed in Table II. The result shows the present method and compared methods have numerically second order convergence in time. The proposed scheme is more accurate than the substepping method of an explicit type, while having similar time cost. In addition, the proposed scheme requires less time cost to

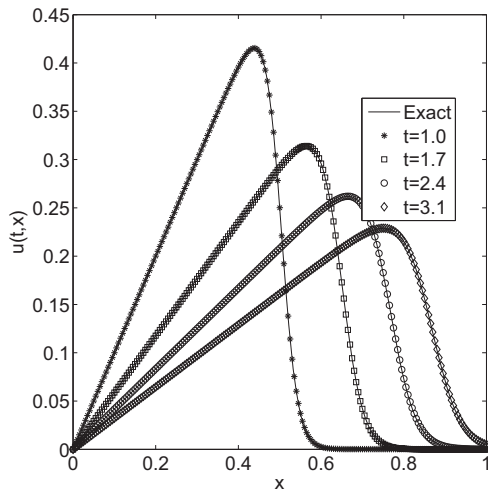


Fig. 1. Numerical solutions at different time stages with $dx = \frac{1}{1000}$, $h = \frac{1}{160}$, when $\nu = 5.0 \times 10^{-3}$.

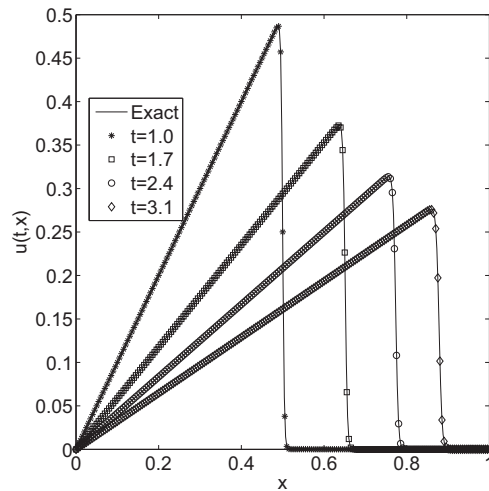


Fig. 2. Numerical solutions at different time stages with $dx = \frac{1}{1000}$, $h = \frac{1}{160}$, when $\nu = 5.0 \times 10^{-4}$.

have similar results for the implicit scheme with using fixed iteration to find the departure points.

Finally, we simulate solutions at some time stages for $\Delta x = \frac{1}{1000}$, $h = \frac{1}{160}$ with different $\nu = 5.0 \times 10^{-k}$, ($k = 3, 4$), which are plotted in Figure 1 and 2. Here, we use the cubic B-spline interpolation. The gradient of numerical solution is steep as the viscosity ν becomes small. It is seen that the numerical results agree with exact solutions.

that the proposed method obtains outstanding numerical results compared with existing methods and are in very good agreement with the exact solutions.

REFERENCES

- [1] C. DEBOOR, *A practical Guide to splines*, Springer Verlag, New York, (1978).
- [2] G. HÄMMERLIN AND K. H. HOFFMANN, *Numerical Mathematics*, Springer Verlag, New York, (1991).
- [3] P. KIM, X. PIAO AND S. D. KIM, *An error corrected Eluer method for solving stiff problems based on chebyshev collocation*, SIAM J. Numer. Anal. 49 (2011) pp. 2211–2230.
- [4] S. D. KIM, X. PIAO AND P. KIM, *Convergence on error correction methods for solving initial value problems*, J. Comput. Appl. Math., 236 (2012) pp. 4448–4461.
- [5] J. L. MCGREGOR, *Economical determination of departure points for semi-Lagrangian models*, Mon. Weather Rev., 121 (1993) pp. 221–230 .
- [6] J. WANG AND A. LAYTON, *New numerical methods for Burgers' equation based on semi-Lagrangian and modified equation approaches*, Appl. Numer. Math., 60 (2010) pp. 645–657.
- [7] D. XIU AND G. E. KARNIADAKIS, *A Semi-Lagrangian high-order method for Navier-Stokes equations* J. Comput. Physics, 172 (2001) pp. 658–684.

TABLE II

COMPARISONS OF THE PROPOSED SCHEME FOR SOLVING EXAMPLE 2., WHEN $\Delta x = \frac{1}{1000}$, $\nu = 5.0 \times 10^{-3}$

h	BDF-Fixed			BDF-Explicit		
	Err ₂ (2.4)	Rates	cpu	Err ₂ (2.4)	Rates	cpu
1/10	4.66e-4	-	1	2.81e-2	-	0.39
1/20	1.23e-4	1.92	1.7	1.12e-2	1.33	0.77
1/40	3.18e-5	1.95	3.1	3.42e-3	1.69	1.5
1/80	8.71e-6	1.87	5.9	9.41e-4	1.88	3.1
1/160	3.41e-6	1.35	11	2.44e-4	1.95	6.2
h	ECM2					
	Err ₂ (2.4)	Rates		cpu		
1/10	4.69e-4	-		0.57		
1/20	1.22e-4	1.94		1.1		
1/40	3.14e-5	1.96		2.2		
1/80	8.39e-6	1.91		4.4		
1/160	2.69e-6	1.64		8.8		

V. CONCLUSION

An iteration free backward semi-Lagrangian scheme for nonlinear advection-diffusion equations is developed. Unlike the traditional way to calculate departure points of particles, we suggest a new methodology that is iteration free and keeps good properties of the conventional second-order implicit method. Throughout several numerical results, it is shown