# An iteration free backward semi-Lagrangian method for coupled Burgers' equations

Soyoon Bak
Department of Mathematics
Kyungpook National University
Daegu, South Korea, 41566
jiya525@knu.ac.kr

Dojin Kim
Department of Mathematics
Kyungpook National University
Daegu, South Korea, 41566
kimdojin@knu.ac.kr

Philsu Kim
Department of Mathematics
Kyungpook National University
Daegu, South Korea, 41566
kimps@knu.ac.kr

***Abstract*—In this paper, we develop a backward semi-Lagrangian method for solving the coupled Burgers' equation. The main difficulty in the backward semi-Lagrangian method for this problem is treating the nonlinearity in the diffusion-reaction type equation, which has a reaction coefficient that is given in terms of coupled partial derivatives. To handle this difficulty, we proposed a new strategy using an extrapolation technique to split the nonlinearity into two diffusion-reaction boundary value problems, which are then solved in turn. In addition, we demonstrated the numerical accuracy and efficiency of the present method by comparing the numerical results with analytical solutions, or other existing numerical solutions that use alternate methods. In addition, we numerically proved that the proposed method exhibits second-order temporal convergence and fourth-order spatial convergence.**

## I. Introduction

The model problem is the coupled viscous Burgers' equation

$$\begin{aligned}&\frac{\partial u_1}{\partial t}-\delta_1\frac{\partial^2 u_1}{\partial x^2}+\alpha_{1,1}u_1\frac{\partial u_1}{\partial x}+\alpha_{1,2}\frac{\partial (u_1u_2)}{\partial x}=0,\\&\frac{\partial u_2}{\partial t}-\delta_2\frac{\partial^2 u_2}{\partial x^2}+\alpha_{2,1}u_2\frac{\partial u_2}{\partial x}+\alpha_{2,2}\frac{\partial (u_1u_2)}{\partial x}=0,\\&\quad t\in[0,T],\quad x\in[x_L,x_R]\end{aligned}\tag{1}$$

with initial and boundary conditions

$$\begin{aligned}&u_1(0,x)=u_1^0(x),\quad u_2(0,x)=u_2^0(x),\quad x\in[x_L,x_R],\\&u_1(t,x_L)=g_1(t),\quad u_1(t,x_R)=g_2(t),\quad t\in[0,T],\\&u_2(t,x_L)=g_3(t),\quad u_2(t,x_R)=g_4(t),\quad t\in[0,T],\end{aligned}\tag{2}$$

which is a simple model of the sedimentation or evolution of the scaled volume concentrations of two types of particles in fluid suspensions or colloids under the effect of gravity (see [1]). Here, $\delta_\kappa$ $(\kappa=1,2)$ are positive constants and $\alpha_{\kappa,\tau}$ $(\tau=1,2)$ are constant depending on the Stokes velocity.

The backward semi-Lagrangian method (BSLM), introduced by Robert [6] in the beginning of the 1980s, has been used extensively for numerical simulations in fluid dynamics. The BSLM has good stability, and solves the problems implicitly along the characteristic curves of fluid particles in the opposite-direction, with large time steps. Recently, the BSLM has become increasingly popular in applied mathematics and scientific problems of CFD, and has been applied to the numerical solution of the advection-diffusion equation [8], [9], Burgers' equations [7], the incompressible Navier-Stokes equation [4], [8], the gyro-kinetic problem [2], and the shallow-water model [5] among others. However, to the best of our knowledge, no BSLM for the coupled nonlinear Burgers equation has been developed so far. Hence, motivated by the success of the BSLM with the above advection processes, the goal of this paper is to further develop the BSLM to solve the model problem. For this purpose, let us introduce the curves $\pi_\kappa(s,x;t)$ $(\kappa=1,2)$ which satisfy the initial value problems (IVP)

$$\begin{aligned}&\frac{d\pi_\kappa(s,x;t)}{dt}=f(t,\pi_\kappa(s,x;t)),\quad t<s;\quad \pi_\kappa(s,x;s)=x,\\&f(t,\pi(t)):=\alpha_{\kappa,1}u_1(t,\pi(t))+\alpha_{\kappa,2}u_2(t,\pi(t)),\end{aligned}\tag{3}$$

where $u_\kappa$ are the solutions of (1), and $x$ denotes the arbitrary spatial variable. Then, from (1), the total derivatives of $u_\kappa(t,\pi_\kappa(s,x;t))$ along each curve $\pi_\kappa(s,x;t)$ satisfy the following system of coupled nonlinear diffusion type equations:

$$\begin{aligned}\frac{d}{dt}u_1\big(t,\pi_1(s,x;t)\big)&=\delta_1\frac{\partial^2 u_1}{\partial x^2}\big(t,\pi_1(s,x;t)\big)\\&\quad-\alpha_{1,2}\frac{\partial u_2}{\partial x}\big(t,\pi_1(s,x;t)\big)u_1\big(t,\pi_1(s,x;t)\big),\\\frac{d}{dt}u_2\big(t,\pi_2(s,x;t)\big)&=\delta_2\frac{\partial^2 u_2}{\partial x^2}\big(t,\pi_2(s,x;t)\big)\\&\quad-\alpha_{2,2}\frac{\partial u_1}{\partial x}\big(t,\pi_2(s,x;t)\big)u_2\big(t,\pi_2(s,x;t)\big)\big).\end{aligned}\tag{4}$$

Notice that the original problem (1) is changed by a coupled problem that simultaneously solves both the IVP (3) and the nonlinear diffusion type equations (4) at each integration step, which is a process of the BSLM. However, it must be noted that the diffusion-reaction problem (4) becomes a nonlinear system that has a reaction coefficient consisting of coupled partial derivatives. This coupled nonlinear structure does not feature in the BSLM for a standard advection-diffusion equation and is the main difficulty to overcome in the BSLM for the model problem (1).

The primary goal of this paper is to develop an efficient numerical method for solving the BSLM system in such a

way that our solution retains the advantage of conventional temporal second-order BSLMs.

## II. Linearization method for (4)

In this section, we develop a linearization method to split the coupled nonlinear diffusion system (4) into two linear diffusion-reaction equations. To do this, let us introduce the notation $u^m(x) := u(t_m, x)$ for the function $u$. After evaluating (4) at the time $t = t_{m+1}$ by setting $s = t_{m+1}$ and then applying the second-order backward differentiation formula (BDF2) to approximate the total derivatives, the system (4) becomes a steady state system of coupled diffusion equations with an asymptotic term that is $\mathcal{O}(h^2)$.

$$\begin{aligned}
&\frac{3}{2h}u_1^{m+1}(x) - \delta_1\frac{\partial^2 u_1^{m+1}(x)}{\partial x^2} + \alpha_{1,2}\frac{\partial u_2^{m+1}(x)}{\partial x}u_1^{m+1}(x)\\
&\qquad = \mathrm{r}_1^{m+1}(x) + \mathcal{O}(h^2),\\
&\frac{3}{2h}u_2^{m+1}(x) - \delta_2\frac{\partial^2 u_2^{m+1}(x)}{\partial x^2} + \alpha_{2,2}\frac{\partial u_1^{m+1}(x)}{\partial x}u_2^{m+1}(x)\\
&\qquad = \mathrm{r}_2^{m+1}(x) + \mathcal{O}(h^2),
\end{aligned} \tag{5}$$

where

$$\mathrm{r}_\kappa^{m+1}(x) = \frac{4u_\kappa^m(\pi_\kappa(t_{m+1}, x; t_m)) - u_\kappa^{m-1}(\pi_\kappa(t_{m+1}, x; t_{m-1}))}{2h}. \tag{6}$$

Notice that the diffusion-reaction system (5) still has a nonlinear term as the reaction coefficients couple partial derivatives with one another. To deal with this difficulty while maintaining a coupling property between $u_1$ and $u_2$,we apply an extrapolation scheme for the function $u_2^{m+1}$ at time $t_{m+1}$ in the first equation of (5) as follows:

$$u_2^{m+1}(x) = \tilde{u}_2^{m+1}(x) + \mathcal{O}(h^2),\ \tilde{u}_2^{m+1}(x) := 2u_2^m(x) - u_2^{m-1}(x). \tag{7}$$

And then, we solve the linear diffusion-reaction equation for $u_1^{m+1}$ described by

$$\begin{cases}
\dfrac{3}{2h}u_1^{m+1}(x) - \delta_1\dfrac{\partial^2 u_1^{m+1}(x)}{\partial x^2} + \alpha_{1,2}\dfrac{\partial \tilde{u}_2^{m+1}(x)}{\partial x}u_1^{m+1}(x)\\
\qquad = \mathrm{r}_1^{m+1}(x) + \mathcal{O}(h^2),\\
u_1^{m+1}(x_L) = g_1(t_{m+1}), \quad u_1^{m+1}(x_R) = g_2(t_{m+1}).
\end{cases} \tag{8}$$

Then, by using the solution $u_1^{m+1}$, we sequentially solve the linear diffusion-reaction equation for $u_2^{m+1}$ described by

$$\begin{cases}
\dfrac{3}{2h}u_2^{m+1}(x) - \delta_2\dfrac{\partial^2 u_2^{m+1}(x)}{\partial x^2} + \alpha_{2,2}\dfrac{\partial u_1^{m+1}(x)}{\partial x}u_2^{m+1}(x)\\
\qquad = \mathrm{r}_2^{m+1}(x) + \mathcal{O}(h^2),\\
u_2^{m+1}(x_L) = g_3(t_{m+1}), \quad u_2^{m+1}(x_R) = g_4(t_{m+1}).
\end{cases} \tag{9}$$

Consequently, (4) is split into the two linear diffusion-reaction boundary value problems (BVPs) given by (8) and (9). We highlight that for a discretization of (8) and (9), the function $\mathrm{r}_\kappa^{m+1}$ defined by (8) must be evaluated at the departure points of the curve satisfying the IVPs (3), and whose velocities are given by a linear combination of the unknown solutions of (8) and (9) in each time integration. Furthermore, from BVPs (8) and (9), we can find approximate values for the concentrations $u_\kappa^{m+1}$ at the grid points only. However, since the departure points $\pi_\kappa(t_{m+1}, x; t_{m-l})$ $(l = 0, 1)$, do not coincide with grid points, we will use the Hermite cubic interpolation $\mathcal{H}$ introduced in [10] to make a fully discretized system for both problems (8) and (9).

## III. Numerical methods

In this section, we present on the implementation used to find the approximate solutions $v_{\kappa,j}^{m+1}$ for the solutions $u_{\kappa,j}^{m+1} := u_\kappa^{m+1}(x_j)$ of the diffusion-reaction problems (8) and (9), respectively. To do this, we use the fourth-order finite difference scheme, and the ECM (for more detailed, see [4]) with the previous approximations $v_{\kappa,j}^k$ $(k \le m,\ 1 \le j \le \bar{N})$ which are assumed to have already been calculated. For convenience, let us introduce a vector

$$\boldsymbol{v}_\kappa^k := \left[v_{\kappa,0}^k, v_{\kappa,1}^k, \ldots, v_{\kappa,N}^k\right]^T, \quad \kappa = 1, 2.$$

We begin with an approximation scheme for the IVPs (3) based on the ECM. To apply the ECM, at each interior grid point $x_j$, we take an Euler's polygon $y_{\kappa,j}(t)$ defined by

$$\begin{aligned}
y_{\kappa,j}(t) &:= \min\{\max\{\hat{y}_{\kappa,j}(t), x_0\}, x_N\},\\
\hat{y}_{\kappa,j}(t) &:= x_j + (t - t_{m+1})\big(\alpha_{\kappa,1}v_{1,j}^m + \alpha_{\kappa,2}v_{2,j}^m\big),
\end{aligned} \tag{10}$$

where we used a min-max technique so that the Euler's polygon $y_{\kappa,j}(t)$ only includes points inside the computational domain. Then, we can find the approximation of the departure points $\pi_{\kappa,j}(t_{m-l}) := \pi_\kappa(t_{m+1}, x_j; t_{m-l})$, $l = 0, 1$ as follows:

$$\pi_{\kappa,j}(t_{m-1}) \approx y_{\kappa,j}(t_{m-1}) + 2h\frac{y'_{\kappa,j}(t_m) - f(t_m, y_{\kappa,j}(t_m))}{1 + hf_x(t_m, y_{\kappa,j}(t_m))}. \tag{11}$$

Since the values $y_{\kappa,j}(t_m)$ given by (10) do not typically align with the grid points, approximate values for the functions $f$ and $f_x$ are required in (11). To find suitable approximations, we use the fourth-order finite difference and the Hermite cubic interpolation (see [10]). More precisely, we approximate $f(t_m, y_{\kappa,j}(t_m))$ and $f_x(t_m, y_{\kappa,j}(t_m))$ in (11) by

$$\begin{aligned}
&f(t_m, y_{\kappa,j}(t_m)) \approx \hat{f}_{\kappa,j}^m, \quad f_x(t_m, y_{\kappa,j}(t_m)) \approx \check{f}_{\kappa,j}^m,\\
&\hat{f}_{\kappa,j}^m := \alpha_{\kappa,1}\mathcal{H}\boldsymbol{v}_1^m(y_{\kappa,j}(t_m)) + \alpha_{\kappa,2}\mathcal{H}\boldsymbol{v}_2^m(y_{\kappa,j}(t_m)),\\
&\check{f}_{\kappa,j}^m := \alpha_{\kappa,1}\mathcal{H}\Big(D_1\boldsymbol{v}_1^m\Big)(y_{\kappa,j}(t_m))\\
&\qquad + \alpha_{\kappa,2}, \mathcal{H}\Big(D_1\boldsymbol{v}_2^m\Big)(y_{\kappa,j}(t_m)),
\end{aligned} \tag{12}$$

respectively, and numerically approximate $\pi_{\kappa,j}(t_{m-1})$ by

$$\pi_{\kappa,j}(t_{m-1}) \approx y_{\kappa,j}(t_{m-1}) + 2h\Big(1 + h\check{f}_{\kappa,j}^m\Big)^{-1}\Big(y'_{\kappa,j}(t_m) - \hat{f}_{\kappa,j}^m\Big). \tag{13}$$

Recall that an approximation of $\pi_{\kappa,j}(t_m)$ is also required to approximate the values $\mathrm{r}_\kappa^{m+1}$ $(\kappa = 1, 2)$ defined by (6). To find such an approximation, we use the Taylor expansion and $\pi_{\kappa,j}(t_{m-1})$ as given by (13), the Hermite interpolation $\mathcal{H}$,

together with (13), the approximation of $\pi_{\kappa,j}(t_m)$ is given as follows:

$$\pi_{\kappa,j}(t_m) \approx \frac{1}{4}\Big(x_j + 3\pi_{\kappa,j}^{m-1} + 2h\sum_{l=1}^{2}\alpha_{\kappa,l}\mathcal{H}\boldsymbol{v}_l^{m-1}(\pi_{\kappa,j}^{m-1})\Big). \tag{14}$$

We are now ready to introduce a fully discretized system for (8) and (9). Using the approximations of $\pi_{\kappa,j}(t_{m-l})$ $(l = 0, 1)$ defined by (13) and (14) with the interpolation $\mathcal{H}$, we first approximate $\mathrm{r}_\kappa^{m+1}$ defined in (6) at the grid point $x_j$ as follows:

$$\mathrm{r}_{\kappa,j}^{m+1} := \mathrm{r}_\kappa^{m+1}(x_j) \approx \frac{1}{2h}\Big(4\mathcal{H}\boldsymbol{v}_\kappa^m(\pi_{\kappa,j}^m) - \mathcal{H}\boldsymbol{v}_\kappa^{m-1}(\pi_{\kappa,j}^{m-1})\Big). \tag{15}$$

Now, we introduce the vector labeled $\tilde{\boldsymbol{w}}$, constructed from the vector $\boldsymbol{w}$ by taking only the interior element, given by

$$\tilde{\boldsymbol{w}} := \Big[w_1, w_2, \ldots, w_{\bar{N}}\Big]^T.$$

After evaluating (8) and (9) at the interior grid point $x_j$ and then applying the finite difference weight matrices $\widetilde{D}_k$, $(k = 1, 2)$ to the result, we can obtain the linear system

$$\begin{aligned}\Big(\frac{3}{2h}\mathcal{I} + \alpha_{1,2}\mathrm{diag}(\mathcal{C}) - \delta_1\widetilde{D}_2\Big)\tilde{\boldsymbol{u}}_1^{m+1} \\ = \tilde{\mathbf{r}}_1^{m+1} + \boldsymbol{b}_1^{m+1} + \mathcal{O}\Big(h^2 + \triangle x^4\Big),\end{aligned} \tag{16}$$

where $\boldsymbol{u}_\kappa^k$ $(k = m, m \pm 1)$ are the vectors defined by

$$\boldsymbol{u}_\kappa^k := \Big[u_{\kappa,0}^k, u_{\kappa,1}^k, \ldots, u_{\kappa,N}^k,\Big]^T$$

and $\Big(\boldsymbol{w}\Big)_j$ denotes the $j$-th component of the vector $\boldsymbol{w}$. Additionally, $\mathcal{I}$ is the identity matrix of size $\bar{N}$,

$$\mathcal{C} := \widetilde{D}_1\Big(2\tilde{\boldsymbol{u}}_2^m - \tilde{\boldsymbol{u}}_2^{m-1}\Big), \quad \tilde{\mathbf{r}}_1^{m+1} := \Big[\mathrm{r}_{1,1}^{m+1}, \mathrm{r}_{1,2}^{m+1}, \ldots, \mathrm{r}_{1,\bar{N}}^{m+1}\Big]^T$$

and $\boldsymbol{b}_1^{m+1}$ is a boundary vector defined as:

$$\begin{aligned}\boldsymbol{b}_1^{m+1} := \frac{\delta_1}{12\Delta x^2}\Big[10g_1^{m+1}, -g_1^{m+1}, 0, \ldots, 0, -g_2^{m+1}, 10g_2^{m+1}\Big]^T \\ -\frac{\alpha_{1,2}}{12\Delta x}\Big[-3\chi_1 g_1^{m+1}, \chi_1 g_1^{m+1}, 0, \ldots, 0, \chi_2 g_2^{m+1}, \\ -3\chi_2 g_2^{m+1}\Big]^T,\end{aligned}$$

where $\chi_1 := 2g_3^m - g_3^{m-1}$ and $\chi_2 := 2g_4^m - g_4^{m-1}$. Finally, after truncating the asymptotic term which is $\mathcal{O}(h^2 + \triangle x^4)$ of (16), and replacing $\tilde{\boldsymbol{u}}_\kappa^m$ and $\tilde{\mathbf{r}}_1^{m+1}$ by $\tilde{\boldsymbol{v}}_\kappa^m$ and $\tilde{\boldsymbol{d}}_1^{m+1}$, respectively, we get the fully discretized system of $\tilde{\boldsymbol{v}}_1^{m+1}$ that approximates $\tilde{\boldsymbol{u}}_1^{m+1}$, and solves the system

$$\Big(\frac{3}{2h}\mathcal{I} + \alpha_{1,2}\mathrm{diag}\big(\widetilde{\mathcal{C}}\big) - \delta_1\widetilde{D}_2\Big)\tilde{\boldsymbol{v}}_1^{m+1} = \tilde{\boldsymbol{d}}_1^{m+1} + \boldsymbol{b}_1^{m+1}, \tag{17}$$

where

$$\widetilde{\mathcal{C}} := \widetilde{D}_1\Big(2\tilde{\boldsymbol{v}}_2^m - \tilde{\boldsymbol{v}}_2^{m-1}\Big), \ \tilde{\boldsymbol{d}}_1^{m+1} := \Big[\mathrm{d}_{1,1}^{m+1}, \mathrm{d}_{1,2}^{m+1}, \ldots, \mathrm{d}_{1,\bar{N}}^{m+1}\Big]^T.$$

Using the approximate solution $\tilde{\boldsymbol{v}}_1^{m+1}$ and the above same discretization procedure for (8), one can get an approximation $\tilde{\boldsymbol{v}}_2^{m+1}$ for $\tilde{\boldsymbol{u}}_2^{m+1}$ by solving the following system

$$\Big(\frac{3}{2h}\mathcal{I} + \alpha_{2,2}\mathrm{diag}\big(\widetilde{D}_1\tilde{\boldsymbol{v}}_1^{m+1}\big) - \delta_2\widetilde{D}_2\Big)\tilde{\boldsymbol{v}}_2^{m+1} = \tilde{\boldsymbol{d}}_2^{m+1} + \boldsymbol{b}_2^{m+1}, \tag{18}$$

where

$$\begin{aligned}\tilde{\boldsymbol{d}}_2^{m+1} := \Big[\mathrm{d}_{2,1}^{m+1}, \mathrm{d}_{2,2}^{m+1}, \ldots, \mathrm{d}_{2,\bar{N}}^{m+1}\Big]^T, \\ \boldsymbol{b}_2^{m+1} := \frac{\delta_2}{12\Delta x^2}\Big[10g_3^{m+1}, -g_3^{m+1}, 0, \ldots, 0, -g_4^{m+1}, 10g_4^{m+1}\Big]^T \\ -\frac{\alpha_{2,2}}{12\Delta x}\Big[-3g_1^{m+1}g_3^{m+1}, g_1^{m+1}g_3^{m+1}, 0, \ldots, 0, \\ g_2^{m+1}g_4^{m+1}, -3g_3^{m+1}g_4^{m+1}\Big]^T.\end{aligned}$$

## IV. Numerical experiments

In this section, in order to gain insight into the performance of the proposed method and to compare their accuracy with those of other existing schemes, one example is tested. The accuracy of the numerical solutions is measured using the maximum norm error $E_\infty(t)$ and the relative $L_2$ norm error $E_2(t)$ defined by

$$\begin{aligned}E_\infty(t_m) &= \max_j |u_j^m - v_j^m|, \\ E_2(t_m) &= \Big(\sum_j |u_j^m - v_j^m|^2\Big)^{1/2} \Big/ \Big(\sum_j |u_j^m|^2\Big)^{1/2},\end{aligned} \tag{19}$$

where $u^k$ is the exact solution at time $t = t_m$ $(m = 1, ..., M)$ on the grid point $x_j$ and $v_j^m$ is its approximation for each $0 \le j \le N$.

**Example** We consider the coupled Burgers' equations (1) over a domain $[-20, 20]$, with $\alpha_{1,1} = \alpha_{2,1} = -2$, $\alpha_{1,2} = \alpha_{2,2} = 2.5$ and $\delta_1 = \delta_2 = 1$. The boundary and initial conditions are taken from the analytic solution given by [3]

$$u_1(t, x) = u_2(t, x) = a_0\Big(1 - \tanh\big(\frac{3}{2}a_0(x - 3a_0t)\big)\Big),$$

where $a_0$ is an arbitrary constant. For a different parameter $a_0$, the solution represents a different anti-kink wave.

TABLE I
Numerical temporal convergence rates of Example with $(a_0, N) = (0.1, 1000)$ at time $t = 1.0$.

| $h$ | $u_1$ | | | | $u_2$ | | | |
|---|---|---|---|---|---|---|---|---|
| | $E_\infty(t)$ | *Rate* | $E_2(t)$ | *Rate* | $E_\infty(t)$ | *Rate* | $E_2(t)$ | *Rate* |
| 1/10 | 1.06E-7 | - | 3.09E-7 | - | 2.95E-8 | - | 8.23E-8 | - |
| 1/20 | 2.86E-8 | 1.89 | 8.35E-8 | 1.89 | 7.92E-9 | 1.90 | 2.22E-8 | 1.89 |
| 1/40 | 7.41E-9 | 1.95 | 2.16E-8 | 1.95 | 2.05E-9 | 1.95 | 5.76E-9 | 1.95 |
| 1/80 | 1.88E-9 | 1.98 | 5.51E-9 | 1.97 | 5.22E-10 | 1.97 | 1.47E-9 | 1.97 |
| 1/160 | 4.74E-10 | 1.99 | 1.39E-9 | 1.97 | 1.33E-10 | 1.97 | 3.74E-10 | 1.97 |

To demonstrate the temporal and spatial convergence orders, and to show the superiority of our proposed method, a number of sets for numerical experiments using this example were conducted. In the first set of experiments, we numerically

TABLE II
NUMERICAL SPATIAL CONVERGENCE RATES OF EXAMPLE WITH $(a_0, h) = (0.1, 0.001)$ AT TIME $t = 1.0$.

| $N$ | $u_1$ | | | | $u_2$ | | | |
|---|---|---|---|---|---|---|---|---|
| | $E_\infty(t)$ | $Rate$ | $E_2(t)$ | $Rate$ | $E_\infty(t)$ | $Rate$ | $E_2(t)$ | $Rate$ |
| 40 | 1.11E-6 | - | 2.98E-6 | - | 1.11E-6 | - | 2.98E-6 | - |
| 80 | 7.12E-8 | 3.96 | 1.92E-7 | 3.96 | 7.12E-8 | 3.96 | 1.92E-7 | 3.96 |
| 160 | 4.48E-9 | 3.99 | 1.21E-8 | 3.99 | 4.49E-9 | 3.99 | 1.21E-8 | 3.99 |
| 320 | 2.73E-10 | 4.03 | 7.48E-10 | 4.02 | 2.84E-10 | 3.98 | 7.67E-10 | 3.98 |
| 640 | 1.75E-11 | 3.97 | 4.73E-11 | 3.98 | 2.08E-11 | 3.77 | 5.42E-11 | 3.82 |

TABLE III
COMPARISON OF THE ERRORS OF EXAMPLE WITH RESULTS FROM LAI ET AL. [3] FOR $u_1$.

| $a_0 = 0.1$ | | | | | | |
|---|---|---|---|---|---|---|
| | Lai [3] | | | Present | | |
| t | $h = 0.001,\ N = 320$ | | | $h = 0.1,\ N = 160$ | | |
| | $E_\infty(t)$ | $E_2(t)$ | CPU | $E_\infty(t)$ | $E_2(t)$ | CPU |
| 1.0 | 5.5555E-7 | 1.4884E-6 | 0.0720 | 1.0480E-7 | 3.0404E-7 | 0.0242 |
| 2.0 | 1.0359E-6 | 2.7891E-6 | 0.1297 | 2.0827E-7 | 6.1703E-7 | 0.0407 |
| 3.0 | 1.4588E-6 | 3.9361E-6 | 0.1897 | 2.9491E-7 | 8.8782E-7 | 0.0555 |
| 4.0 | 1.8346E-6 | 4.9545E-6 | 0.2451 | 3.6827E-7 | 1.1220E-6 | 0.0675 |
| 5.0 | 2.1723E-6 | 5.8640E-6 | 0.3070 | 4.3088E-7 | 1.3244E-6 | 0.0867 |
| 10.0 | 3.4584E-6 | 9.2232E-6 | 0.6021 | 6.3692E-7 | 1.9802E-6 | 0.1559 |
| $a_0 = 0.5$ | | | | | | |
| | Lai [3] | | | Present | | |
| t | $h = 0.001,\ N = 320$ | | | $h = 0.04,\ N = 160$ | | |
| | $E_\infty(t)$ | $E_2(t)$ | CPU | $E_\infty(t)$ | $E_2(t)$ | CPU |
| 1.0 | 6.7508E-4 | 1.6361E-4 | 0.0719 | 4.3197E-4 | 1.1858E-4 | 0.0435 |
| 2.0 | 8.1707E-4 | 1.9746E-4 | 0.1301 | 4.4659E-4 | 1.2263E-4 | 0.0716 |
| 3.0 | 8.6378E-4 | 2.0557E-4 | 0.1884 | 4.4615E-4 | 1.2385E-4 | 0.1001 |
| 4.0 | 8.8157E-4 | 2.0542E-4 | 0.2480 | 4.4181E-4 | 1.2340E-4 | 0.1193 |
| 5.0 | 8.9059E-4 | 2.0231E-4 | 0.3066 | 4.3749E-4 | 1.2146E-4 | 0.1526 |
| 10.0 | 8.9808E-4 | 1.8109E-4 | 0.6024 | 4.2159E-4 | 1.0524E-4 | 0.2757 |

estimated the temporal convergence rate of the proposed method for $u_1$ and $u_2$ at time $t = 1.0$ by ranging the temporal step size $h$ from $1/10$ to $1/160$, with a fixed spatial grid resolution of $N = 1000$. The computed results are listed in Table I, where it is clear that the proposed numerical method gives the expected second-order temporal convergence rate. Additionally, in order to examine the spatial convergence rate of the proposed method, the simulation was run until the time $t = 1.0$, and the errors were calculated for a fixed temporal step size of $h = 0.001$ by varying the spatial grid $\Delta x$ from $\frac{1}{40}$ to $\frac{1}{640}$. Our results are displayed in Table II, where they show that the spatial convergence rate of the proposed numerical method is approximately of the fourth-order. Finally, we simulated Example with $a_0 = 0.1$ and $0.5$. The errors and the computational time cost (CPU) were measured for $u_1$ and the results are captured in Table III. We also compared our results to those of Lai et al. [3] for $u_1$ at $a_0 = 0.1,\ 0.5$, for a variety of time durations, $0.1, 0.5, 1.0, 2.0$ and $10.0$. The results of Tables III shows that our method yields a more accurate numerical solution in less cputime, and with larger temporal and spatial step sizes than the method of Lai et al. [3]. The numerical results guarantee that by allowing use of the maximum time step size, our BSLM is better suited for long time simulations than a comparative method, which is one of the salient features of BSLM.

## V. CONCLUSION

We have proposed a backward semi-Lagrangian method for solving coupled viscous Burgers' equations. For the present scheme, we used the second-order backward differentiation formula for the total time derivative, fourth-order finite differences for partial derivatives, the error correction method for the highly nonlinear IVP and Hermite cubic interpolation for function evaluations at the non-grid points. In order to resolve the nonlinear problem in the coupled diffusion-reaction equations, we provided an extrapolation technique to split the nonlinearity into two diffusion-reaction boundary value problems, which are sequentially solved. We demonstrated the numerical accuracy and efficiency of the present method by comparing the numerical results with analytical solutions, or other existing numerical solutions that use alternate methods. In addition, we numerically proved that the proposed method exhibits second-order temporal convergence and fourth-order spatial convergence.

## ACKNOWLEDGEMENTS

This research was supported by the basic science research program through the National Research Foundation of Korea (NRF), and funded by the Ministry of Education, Science and Technology (grant number 2016R1A2B2011326).

## REFERENCES

[1] S.E. ESIPOV, *Coupled Burgers equations: A model of polydispersive sedimentation*, J. Comput. Appl. Math., 128, (2001) pp. 423–445.
[2] J-.M. KWON, D. YI, X. PIAO AND P. KIM, *Development of semi-Lagrangian gyrokinetic code for full-f turbulence simulation in general tokamak geometry*, J. Comput. Phys., 283 (2015) pp. 518–540.
[3] H. LAI AND C. MA, *A new lattice Boltzmann model for solving the coupled viscous Burgers' equation*, Physica A, 395 (2014) pp. 445–457.
[4] X. PIAO, S. BU, S. BAK AND P. KIM, *An iteration free backward semi-Lagrangian scheme for solving incompressible Navier-Stokes equations*, J. Comput. Phys., 283 (2015) pp. 189–204.
[5] M. TOLSTYKH AND V.V. SHASHKIN, *Vorticity-divergence mass-conserving semi-Lagrangian shallow-water model using the reduced grid on the sphere*, J. Comput. Phys., 231(11) (2012) pp. 4205–4233.
[6] A. ROBERT, *A stable numerical integration scheme for the primitive meteorological equations*, Atmosphere-Ocean, 19 (1) pp. 35–46 (1981).
[7] J. WANG AND A. LAYTON, *New numerical methods for Burgers' equation based on semi-Lagrangian and modified equation approaches*, Appl. Math. Comput., 60(6) (2010) pp. 645–657.
[8] D. XIU AND G.E. KARNIADAKIS, *A semi-Lagrangian high-order method for Navier-Stokes equations*, J. Comput Phys., 172 (2001) pp. 658–684.
[9] D. XIU, S.J. SHERWIN, S. DONG AND G.E. KARNIADAKIS, *Strong and auxiliary forms of the semi-Lagrangian method for incompressible flows*, Journal of Scientific Computing, 25(1) (2005) pp. 323–346.
[10] T. YABE AND T. AOKI, *A universal solver for hyperbolic equations by cubic-polynomial interpolation I. One-dimensional solvers*, Comput. Phys. Commun., 66(2-3) (1991) pp. 219–232.