



Simulations of nonlinear advection-diffusion models through various finite element techniques

H. Tunc and M. Sari*

Department of Mathematics, Faculty of Arts and Science, Yildiz Technical University, Istanbul 34220, Turkey.

Received 8 July 2018; received in revised form 3 June 2019; accepted 12 October 2019

KEYWORDS

Finite element method;
 α -family of approximation;
 Advection-diffusion model;
 Burgers equation;
 Taylor-Galerkin method;
 Collocation method.

Abstract. In this study, the Burgers equation is analyzed both numerically and mathematically by considering various finite element based techniques including Galerkin, Taylor-Galerkin and collocation methods for spatial variation of the equation. The obtained time dependent ordinary differential equation system is approximately solved by α -family of time approximation. All these methods are theoretically explained using cubic B-spline basis and weight functions for a strong form of the model equation. Von Neumann matrix stability analysis is performed for each of these methods and stability criteria are determined in terms of the problem parameters. Some challenging examples of the Burgers equation are numerically solved and compared with the literature and exact solutions. Also, the proposed techniques have been compared with each other in terms of their advantages and disadvantages depending on the problem types. The more advantageous method of the three, in comparison to the other two, has been found for the special cases of the present problem in detail.

© 2020 Sharif University of Technology. All rights reserved.

1. Introduction

Almost all physical processes faced in nature are described by partial differential equations. Many mathematical models are used to represent physical flows in various fields of sciences such as wave propagation, shallow water waves, reaction-diffusion models, biomechanical waves etc. One of such models is the Burgers equation attracting much attention in analysing evolution equations representing various physical processes [1]. Its computation is a natural first step towards developing methods. The existence and uniqueness of classical solutions to the Burgers equation were shown under certain conditions [2].

In recent decades, producing a solution to the Burgers equation has attracted great attention. Since

exact solutions, infinite series solutions, and fail for small viscosity values [3], $\varepsilon < 0.01$, plenty of researchers [4–7] have suggested various numerical techniques based on different methods such as finite difference, finite element and boundary element methods in solving the model equation.

Behaviours of many physical processes encountered in models of advection mechanisms, and diffusion transports lead to the Burgers equation. Thus, the equation arising in various physical areas of science are considered:

$$u_t + uu_x = \varepsilon u_{xx}, \quad a \leq x \leq b, \quad (1)$$

with boundary conditions:

$$\begin{aligned} u(a, t) &= f_1(t), & t > 0, \\ u(b, t) &= f_2(t), & t > 0, \end{aligned} \quad (2)$$

and initial condition:

$$u(x, 0) = g(x), \quad a < x < b, \quad (3)$$

*. Corresponding author. Tel.: +90 212 383 43 60
 E-mail address: sarim@yildiz.edu.tr (M. Sari)

where ε is the kinematic viscosity constant for $\varepsilon > 0$, and f_1, f_2 and g are known functions. The subscripts, x and t , represent differentiations with respect to space x and time t , respectively.

The Burgers equation is the nonlinear model equation for diffusive waves in fluid dynamics. The corresponding equation has also many application areas including theory of shock waves, sound waves in a viscous medium, mathematical modeling of turbulent fluid and so on. The Burgers equation can be solved exactly by using Hopf-Cole transformation [8,9]. So, the Burgers equation can be regarded as a test equation for the numerical methods. So far, lots of numerical methods have been used to solve the equation.

Spline in tension approximation [10], a hybrid numerical scheme including wavelets and finite differences [11], high order splitting methods [12], polynomial based differential quadrature method [13], boundary element method [14], integral equation method [15], explicit exponential finite difference method [16], implicit fourth-order Compact Finite Difference (CFD) scheme [17], higher order time integration formulae [18], sixth order CFD scheme [1] are some of the papers dealing with the numerical solutions of the Burgers equation. Various versions of finite element methods are the main interest of many researchers to find out effective numerical solutions of various types of partial differential equations [19–21].

Nowadays, B-spline basis functions are the main interest of many researchers to find out effective numerical solutions of partial differential equations [7,22–25]. Different finite element based techniques have profoundly been analyzed in the literature. For instance; least-squares B-spline finite element method was used by Kutluay et al. [22], a quadratic B-spline Galerkin method was introduced by Sari and Tunc [23], B-spline lumped Galerkin finite element method was organized by Kutluay and Esen [24], quadratic and cubic B-spline Galerkin methods were used in references [25,26] in dealing nonlinear advection-diffusion processes. In a recent study of Sari et al. [7], a cubic B-spline Galerkin method with higher order splitting approaches has also been considered for the Burgers equation. B-spline based numerical methods were also applied to various nonlinear equations such as generalized Benjamin-Bona-Mahony-Burgers, Gilson-Pickering and Cahn-Hilliard equations [27–29].

The current study discovers some finite element based hybrid techniques involving Galerkin, Taylor-Galerkin and collocation methods based on cubic B-spline basis functions to analyze the nonlinear advection-diffusion processes. To integrate the resulted system of Ordinary Differential Equations (ODE), α -family of time approximation is performed, and fully discrete algebraic equations are obtained in terms of the parameters. Note that the strong form of the Burgers

equation (Eq. (1)) is accepted, as opposed to the weak form commonly used in the literature [22,24,25], since the strong form leads to computationally more economic and more accurate results. Stability analysis of the considered methods are performed using the von Neumann approach. Thus, the effect of the general explicit-implicit α -family of time approximation on the stability of the complete discrete system has been analyzed. The currently produced quantitative and qualitative results revealed that each of the considered finite element methods has its own advantageous and disadvantageous. Depending on the problem parameters, the present techniques are discussed in a comparative way in terms of accuracy, computational efficiency and physical reliability. Four challenging examples representing various nonlinear advection-diffusion processes at shock stages have been taken into consideration. Current numerical techniques have been proven to have a high ability to analyze the nonlinear advection-diffusion process in terms of accuracy, stability and computational efficiency.

2. Numerical methods

The implementation of the Galerkin, collocation and Taylor-Galerkin methods and the α -family approximation in time for the equation of interest are discussed in this section. Generally, use of finite element techniques to approximate the spatial derivatives of the unsteady partial differential equations leads to a system of nonlinear ODEs. Thus, in order to cope with the system, a suitable time integration method should be used. We consider the α -family approximation to deal with the nonlinear ODE system, and then the discrete system is shown to be stable for all considered techniques.

2.1. The Galerkin method

To solve Eq. (1) with given boundary conditions (2) and initial condition (3), the Galerkin cubic B-spline finite element method is used for spatial approximation. The selection of these type basis functions is very suitable and has advantages [7].

The interval $[a, b]$ is partitioned into N finite elements. Each element has equal length h and element nodes are defined as $a = x_0 < x_1 < \dots < x_N = b$, where $x_{m+1} = x_m + h$ ($m = 0, 1, \dots, N - 1$). Let φ_m be the cubic B-spline basis functions and it is given [7] as:

$$\varphi_m(x) = \frac{1}{h^3} \begin{cases} (x - x_{m-2})^3 \\ h^3 + 3h^2(x - x_{m-1}) + 3h(x - x_{m-1})^2 \\ - 3(x - x_{m-1})^3 \\ h^3 + 3h^2(x_{m+1} - x) + 3h(x_{m+1} - x)^2 \\ - 3(x_{m+1} - x)^3 \\ (x_{(m+2)} - x)^3 \\ 0 \end{cases}$$

$$x \in \begin{cases} [x_{m-2}, x_{m-1}] \\ [x_{m-1}, x_m] \\ [x_m, x_{m+1}] \\ [x_{m+1}, x_{m+2}] \\ \text{Otherwise} \end{cases} \quad (4)$$

The corresponding cubic B-spline basis functions include the set of splines $\{\varphi_{-1}, \varphi_0, \dots, \varphi_{N+1}\}$ and the global approximation function $\tilde{u}_N(x, t)$ can be expressed as:

$$\tilde{u}_N(x, t) = \sum_{m=-1}^{N+1} \beta_m(t) \varphi_m(x), \quad (5)$$

where $\beta_m(t)$ are the time part of approximation function $\tilde{u}_N(x, t)$ and will be determined from the time approximation.

To compute element matrices easily, it is required to use a local coordinate system considering Eq. (4) and $\sigma = x - x_m$ where $0 \leq \sigma \leq h$. The basis functions will be in the form:

$$\varphi_m(\sigma) = \frac{1}{h^3} \begin{cases} (h - \sigma)^3 \\ h^3 + 3h^2(h - \sigma) + 3h(h - \sigma)^2 - 3(h - \sigma)^3 \\ h^3 + 3h^2\sigma + 3h\sigma^2 - 3\sigma^3 \\ \sigma^3 \end{cases} \quad (6)$$

Each finite element $[x_m, x_{m+1}]$ is covered by the set of four cubic B-splines $\{\varphi_{m-1}, \varphi_m, \varphi_{m+1}, \varphi_{m+2}\}$. Table 1 shows the values of φ_m, φ'_m and φ''_m at the end points of element $[x_m, x_{m+1}]$. Local approximation function on the element $[x_m, x_{m+1}]$ is defined as follows:

$$\tilde{u}_N(x, t) = \sum_{i=m-1}^{m+2} \beta_i(t) \varphi_i(x). \quad (7)$$

Values of the local approximation function $\tilde{u}_N(x, t)$ and its first two derivatives at the end points of the interval $[x_m, x_{m+1}]$ are defined in terms of time dependent quantities $\beta_m(t)$ using Eq. (7) and Table 1. The corresponding values will then be:

$$\tilde{u}_N(x_m, t) = \beta_{m-1} + 4\beta_m + \beta_{m+1},$$

$$\tilde{u}_N(x_{m+1}, t) = \beta_m + 4\beta_{m+1} + \beta_{m+2},$$

$$\tilde{u}'_N(x_m, t) = \frac{3}{h}(\beta_{m+1} - \beta_{m-1}),$$

$$\tilde{u}'_N(x_{m+1}, t) = \frac{3}{h}(\beta_{m+2} - \beta_m),$$

Table 1. Values of approximate function and its derivatives at the end points of the element.

x	x_{m-2}	x_{m-1}	x_m	x_{m+1}	x_{m+2}
φ_m	0	1	4	1	0
φ'_m	0	$-3/h$	0	$3/h$	0
φ''_m	0	$6/h^2$	$-12/h^2$	$6/h^2$	0

$$\tilde{u}''_N(x_m, t) = \frac{6}{h^2}(\beta_{m+1} - 2\beta_m + \beta_{m-1}),$$

$$\tilde{u}''_N(x_{m+1}, t) = \frac{6}{h^2}(\beta_{m+2} - 2\beta_{m+1} + \beta_m). \quad (8)$$

Now it is time to apply the Galerkin approach. By considering element $[x_m, x_{m+1}]$, let us multiply Eq. (1) by a test function v and integrate over the interval $[x_m, x_{m+1}]$. One can then write:

$$\int_{x_m}^{x_{m+1}} (u_t + uu_x - \varepsilon u_{xx}) v dx = 0. \quad (9)$$

The test function v is selected to be equal to the cubic B-spline basis functions. This type of procedure is known as the Galerkin approach in the finite element method. Using Eq. (7) and local coordinate system Eq. (6), Eq. (9) yields the following relation:

$$\begin{aligned} & \sum_{j=m-1}^{m+2} \left[\int_0^h \varphi_i \varphi_j d\sigma \right] \frac{d\beta_j^e}{dt} \\ & + \sum_{j=m-1}^{m+2} \sum_{k=m-1}^{m+2} \left[\int_0^h \varphi_i \varphi'_j \varphi_k d\sigma \right] \beta_k^e \beta_j^e \\ & - \varepsilon \sum_{j=m-1}^{m+2} \left[\int_0^h \varphi_i \varphi''_j d\sigma \right] \beta_j^e = 0, \end{aligned}$$

or, in a matrix notation:

$$M^e \frac{d\beta^e}{dt} + \beta^{eT} L^e \beta^e - \varepsilon K^e \beta^e = 0, \quad (10)$$

where:

$$M^e_{ij} = \int_0^h \varphi_i \varphi_j d\sigma, \quad K^e_{ij} = \int_0^h \varphi_i \varphi''_j d\sigma,$$

$$L^e_{ijk} = \int_0^h \varphi_i \varphi''_j \varphi_k d\sigma, \quad \beta^e = (\beta_{l-1}, \beta_l, \beta_{l+1}, \beta_{l+2})^T, \quad (11)$$

where $i, j, k = m - 1, m, m + 1, m + 2$ for the element $[x_m, x_{m+1}]$. In Eq. (10), M^e and K^e are (4×4) matrices and are independent of time. L^e is a $(4 \times 4 \times 4)$ tensor and L can be transformed to a time dependent matrix R by using:

$$R^e_{ij} = \sum_{k=m-1}^{m+2} L^e_{ijk} \beta_k^e. \quad (12)$$

After the assembling process of each element the matrix form will finally be:

$$\mathbf{M}^* \frac{d\beta}{dt} + \mathbf{R}^* \beta - \varepsilon \mathbf{K}^* \beta = 0, \quad (13)$$

where \mathbf{M}^* , \mathbf{R}^* and \mathbf{K}^* are $(N + 3) \times (N + 3)$ matrices and $\beta = (\beta_{-1}, \beta_0, \dots, \beta_{N+1})^T$ is the unknown time approximation vector.

2.2. The collocation method

Expressions (4)–(8) are reconsidered as stated above. Assume that the collocation points have the following form:

$$x_m = a + m * h, \quad m = 0, 1, \dots, N, \tag{14}$$

where $\omega^m = [x_m, x_{m+1}]$ is the m th element of the domain, h is the element size and N is the number of elements. Substitution of local approximate solution (7) into Eq. (1) yields:

$$\frac{\partial \tilde{u}_N(x, t)}{\partial t} + \tilde{u}_N(x, t) \frac{\partial \tilde{u}_N(x, t)}{\partial x} = \epsilon \frac{\partial^2 \tilde{u}_N(x, t)}{\partial x^2}. \tag{15}$$

Use of Eq. (8) in Eq. (15) at $x = x_m$ leads to:

$$\begin{aligned} &\dot{\beta}_{m-1} + 4\dot{\beta}_m + \dot{\beta}_{m+1} \\ &+ \frac{3}{h} (\beta_{m-1} + 4\beta_m + \beta_{m+1}) (-\beta_{m-1} + \beta_{m+1}) \\ &= \epsilon \frac{6}{h^2} (\beta_{m-1} - 2\beta_m + \beta_{m+1}), \end{aligned} \tag{16}$$

where $\dot{\beta}$ stands for the time differentiation. For all $m = 0, \dots, N$, one can obtain $N + 1$ differential equations. This system can be rewritten in a more compact form as follows:

$$M \frac{d\beta}{dt} + R(\beta)\beta - \epsilon K\beta = 0, \tag{17}$$

where M and K are $(N + 1) \times (N + 3)$ time independent matrices, $R(\beta)$ is $(N + 1) \times (N + 3)$ time dependent matrix and $\beta = (\beta_{-1}, \beta_0, \dots, \beta_{N+1})^T$ is $((N + 3) \times 1)$ unknown time approximation vector. Assembling each element and using the boundary conditions lead to:

$$\mathbf{M}^* \frac{d\beta}{dt} + \mathbf{R}^* \beta - \epsilon \mathbf{K}^* \beta = 0, \tag{18}$$

where \mathbf{M}^* , \mathbf{R}^* and \mathbf{K}^* are $(N + 1) \times (N + 1)$ dimensional matrices, and $\beta = (\beta_0, \dots, \beta_N)^T$ is the unknown time approximation vector.

2.3. The Taylor-Galerkin method

In the Taylor-Galerkin method, the time discretization is performed before the spatial discretization. If we use the Taylor series expansion, one can write:

$$u_t^n = \frac{u^{n+1} - u^n}{dt} - \frac{dt}{2} u_{tt}^n - O(dt^2). \tag{19}$$

Taking the time derivative of Eq. (1) and using forward Euler time stepping, one can write:

$$u_{tt} = (-uu_x + \epsilon u_{xx})_t^n = -u^n (u_t^n)_x - u_x^n u_t^n + \epsilon (u_t^n)_{xx}, \tag{20}$$

$$\begin{aligned} u_{tt} = &-u^n \left(\frac{u^{n+1} - u^n}{dt} \right)_x - u_x^n \left(\frac{u^{n+1} - u^n}{dt} \right) \\ &+ \epsilon \left(\frac{u^{n+1} - u^n}{dt} \right)_{xx}. \end{aligned} \tag{21}$$

If we substitute Eqs. (1) and (21) into Eq. (19), then we reach:

$$\begin{aligned} -u^n u_x^n + \epsilon u_{xx}^n = &\frac{u^{n+1} - u^n}{dt} - \frac{dt}{2} \left[-u^n \left(\frac{u^{n+1} - u^n}{dt} \right)_x \right. \\ &\left. - u_x^n \left(\frac{u^{n+1} - u^n}{dt} \right) + \epsilon \left(\frac{u^{n+1} - u^n}{dt} \right)_{xx} \right]. \end{aligned} \tag{22}$$

Organization of Eq. (22) in terms of u^n and u^{n+1} leads to the following semi-discrete form:

$$\begin{aligned} &\left(1 + \frac{dt}{2} u^n \partial_x + \frac{dt}{2} u_x^n - \frac{dt}{2} \epsilon \partial^2_x \right) u^{n+1} \\ &= \left(1 + \frac{dt}{2} \epsilon \partial^2_x \right) u^n. \end{aligned} \tag{23}$$

By considering the Galerkin approach as stated in Section 2.1, we get the following matrix equation for the element $e = [x_m, x_{m+1}]$:

$$\begin{aligned} &\left[\mathbf{A}^e + \frac{dt}{2} \mathbf{C}^e(\beta) + \frac{dt}{2} \mathbf{D}^e(\beta) - \epsilon \frac{dt}{2} \mathbf{B}^e \right] \{\beta\}_{n+1} \\ &= \left[\mathbf{A}^e + \epsilon \frac{dt}{2} \mathbf{B}^e \right] \{\beta\}_n, \end{aligned} \tag{24}$$

where:

$$\begin{aligned} \mathbf{A}_{ij}^e = &\int_0^h \varphi_i \varphi_j d\sigma, \quad \mathbf{B}_{ij}^e = \int_0^h \varphi_i \varphi_j'' d\sigma, \\ i, j = &m - 1, m, m + 1, m + 2, \end{aligned} \tag{25}$$

$$\mathbf{C}_{ij}^e = \sum_{k=m-1}^{m+2} L_{ijk}^e \beta_k, \quad \mathbf{D}_{ij}^e = \sum_{k=m-1}^{m+2} N_{ijk}^e \beta_k, \tag{26}$$

and:

$$\begin{aligned} \mathbf{L}_{ijk}^e = &\int_0^h \varphi_i \varphi_j \varphi_k' d\sigma, \quad \mathbf{N}_{ijk}^e = \int_0^h \varphi_i \varphi_j' \varphi_k d\sigma, \\ i, j, k = &m - 1, m, m + 1, m + 2. \end{aligned} \tag{27}$$

After the assembling process of each element $e = 1, 2, \dots, N$ and imposing boundary conditions, the following global matrix system can then be obtained:

$$\begin{aligned} &\left[\mathbf{A}^* + \frac{dt}{2} (\mathbf{C}^*(\beta) + \mathbf{D}^*(\beta)) - \epsilon \frac{dt}{2} \mathbf{B}^* \right] \{\beta\}_{s+1} \\ &= \left[\mathbf{A}^* + \epsilon \frac{dt}{2} \mathbf{B}^* \right] \{\beta\}_s, \end{aligned} \tag{28}$$

where \mathbf{A}^* and \mathbf{B}^* are $(N + 1) \times (N + 1)$ time independent matrices, while $\mathbf{C}^*(\beta)$ and $\mathbf{D}^*(\beta)$ are $(N + 1) \times (N + 1)$ time dependent matrices. In Eq. (28), β is $(N + 1) \times 1$ unknown vector to be determined for each time step. By obtaining $\{\beta\}_0$ in the first step and correcting $\mathbf{C}^*(\beta)$ and $\mathbf{D}^*(\beta)$ in each step, all unknown vectors $\{\beta\}_s$ can be calculated.

2.4. α -family of time variation

In Sections 2.1 and 2.2 we have obtained system of differential equations in the same form with different entries of the matrices. From the final matrix form Eqs. (13) and (18), one can write:

$$\mathbf{M}^* \frac{d\beta}{dt} + \mathbf{R}^* \beta - \varepsilon \mathbf{K}^* \beta = 0. \tag{29}$$

As stated in [23] the α -family of approximation can be defined as:

$$\{\beta\}_{s+1} = \{\beta\}_s + dt\{\dot{\beta}\}_{s+\alpha}, \tag{30}$$

$$\{\beta\}_{s+\alpha} = (1 - \alpha) \left\{ \dot{\beta} \right\}_s + \alpha \left\{ \dot{\beta} \right\}_{s+1}, \tag{31}$$

or:

$$dt \left[(1 - \alpha) \left\{ \dot{\beta} \right\}_s + \alpha \left\{ \dot{\beta} \right\}_{s+1} \right] = \{\beta\}_{s+1} - \{\beta\}_s, \tag{32}$$

where $0 \leq \alpha \leq 1$ and $t_{s+1} - t_s = dt$. $\dot{\beta}$ stands for the time differentiation. Using the same steps of the procedure given in [23], Eq. (29) becomes:

$$\begin{aligned} & [\mathbf{M}^* + \alpha dt (R_{s+1}^* - \varepsilon \mathbf{K}^*)] \{\beta\}_{s+1} \\ & = [\mathbf{M}^* - (1 - \alpha) dt (R_s^* - \varepsilon \mathbf{K}^*)] \{\beta\}_s, \end{aligned} \tag{33}$$

where matrices \mathbf{M}^* and \mathbf{K}^* are independent of time while \mathbf{R}^* depends on time. The initial approximation $\{\beta\}_0$ can be found from the initial condition (3). With consideration of the initial condition and two additional conditions, $(N + 1)$ equations can be found using the first derivative of the approximate function at both ends as follows:

$$\begin{aligned} \tilde{u}(x_l, 0) &= g(x_l) = \{\beta\}_{l-1}^0 + 4\{\beta\}_l^0 + \{\beta\}_{l+1}^0, \\ l &= 0, 1, \dots, N, \end{aligned}$$

$$\tilde{u}'_N(x_0, 0) = g'(x_0) = \frac{3}{h} (\{\beta\}_1^0 - \{\beta\}_{-1}^0),$$

$$\tilde{u}'_N(x_N, 0) = g'(x_N) = \frac{3}{h} (\{\beta\}_{N+1}^0 - \{\beta\}_{N-1}^0).$$

Hence, the system will be $(N + 3) \times (N + 3)$ and can be solved using any suitable linear system solver to obtain initial vector $\{\beta\}_0$.

3. Stability analyses of the hybrid approximations

In Section 2, we constructed various finite element techniques for the solution of the Burgers equation, such as the Galerkin, collocation and Taylor-Galerkin methods. We analyze the stability of these iterations by considering the von Neumann stability analysis. Stability criteria of the Galerkin and collocation methods

are obtained in terms of the parameter α . Stability conditions of the Taylor-Galerkin method are analyzed by considering various parameters used in the formula. It is useful to mention that, as we evaluate a typical row of the obtained iterations, the values of β_{s+1} and β_s in the time dependent matrices R_{s+1}^* and R_s^* are locally constant and equal to p .

3.1. Stability of the Galerkin method

Let us consider the iteration (33) with the typical rows of the matrices as follows:

$$\mathbf{M}^* = \frac{h}{140} (1, 120, 1191, 2416, 1191, 120, 1),$$

$$\mathbf{R}^* = \frac{p}{40} (-3, -168, -735, 0, 735, 168, 3),$$

$$\mathbf{K}^* = \frac{1}{10h} (3, 72, 45, -240, 45, 72, 3). \tag{34}$$

Consider the von Neumann theory with the Fourier growth factor defined by:

$$\beta_s^n = \tilde{\beta}^n e^{iskh}, \tag{35}$$

where k and h stand for the mode number and the element size, respectively. We aim to determine how much β_s^{n+1} grows under the consideration of β_s^n by using Eqs. (33) and (35).

A typical row of the septa-diagonal system (Eq. (33)) can be written as:

$$\begin{aligned} & c_1 \beta_{s-3}^{n+1} + c_2 \beta_{s-2}^{n+1} + c_3 \beta_{s-1}^{n+1} + c_4 \beta_s^{n+1} + c_5 \beta_{s+1}^{n+1} \\ & + c_6 \beta_{s+2}^{n+1} + c_7 \beta_{s+3}^{n+1} = c_8 \beta_{s-3}^n + c_9 \beta_{s-2}^n \\ & + c_{10} \beta_{s-1}^n + c_{11} \beta_s^n + c_{12} \beta_{s+1}^n + c_{13} \beta_{s+2}^n \\ & + c_{14} \beta_{s+3}^n, \end{aligned} \tag{36}$$

where:

$$c_1 = r_1 - 3r_2 - 3r_3,$$

$$c_2 = 120r_1 - 168r_2 - 72r_3,$$

$$c_3 = 1191r_1 - 735r_2 - 45r_3,$$

$$c_4 = 2416r_1 + 240r_3,$$

$$c_5 = 1191r_1 + 735r_2 - 45r_3,$$

$$c_6 = 120r_1 + 168r_2 - 72r_3,$$

$$c_7 = r_1 + 3r_2 - 3r_3,$$

$$c_8 = r_1 + 3d_2 + 3d_3,$$

$$c_9 = 120r_1 + 168d_2 + 72d_3,$$

$$c_{10} = 1191r_1 + 735d_2 + 45d_3,$$

$$\begin{aligned}
 c_{11} &= 2416r_1 - 240d_3, \\
 c_{12} &= 1191r_1 - 735d_2 + 45d_3, \\
 c_{13} &= 120r_1 - 168d_2 + 72d_3, \\
 c_{14} &= r_1 - 3d_2 + 3d_3, \\
 r_1 &= \frac{h}{140}, \quad r_2 = \frac{p\alpha dt}{40}, \quad r_3 = \frac{\varepsilon\alpha dt}{40h}, \\
 d_2 &= \frac{p(1-\alpha)dt}{40}, \quad d_3 = \frac{\varepsilon(1-\alpha)dt}{40h}. \tag{37}
 \end{aligned}$$

Substituting Eq. (35) into Eq. (36) and using the Euler expansion for exponential terms, one can write:

$$(g + 120r_3 + iv)\tilde{\beta}^{n+1} = (g - 120d_3 - iv')\tilde{\beta}^n, \tag{38}$$

where:

$$\begin{aligned}
 g &= (r_1 - 3r_3)\cos(3kh) + (120r_1 - 72r_3)\cos(2kh) \\
 &\quad + (1191r_1 - 45r_3)\cos(kh) + 1208r_1, \\
 g^* &= (r_1 + 3d_3)\cos(3kh) + (120r_1 + 72d_3)\cos(2kh) \\
 &\quad + (1191r_1 + 45d_3)\cos(kh) + 1208r_1, \\
 v &= 3r_2\sin(3kh) + 168r_2\sin(2kh) + 735r_2\sin(kh), \\
 v^* &= 3d_2\sin(3kh) + 168d_2\sin(2kh) + 735d_2\sin(kh).
 \end{aligned}$$

By defining the amplification factor, Eq. (38) can be rewritten as follows:

$$\tilde{\beta}^{n+1} = \frac{(g^* - 120d_3 - iv^*)}{(g + 120r_3 + iv)}\tilde{\beta}^n = z\tilde{\beta}^n. \tag{39}$$

Scheme (33) is stable if and only if $|z| < 1$. The modulus of the amplification factor is expressed as:

$$|z| = \sqrt{z\bar{z}} = \sqrt{\frac{(g^* - 120d_3)^2 + (v^*)^2}{(g + 120r_3)^2 + v^2}}. \tag{40}$$

Reasonable selection of the parameter h leads to sufficiently small values of r_1 , i.e. $r_1 \ll 1$. Then, whenever $\alpha \geq 0.5$, the inequalities $d_2 \leq r_2$, $d_3 \leq r_3$ and $|(g^* - 120d_3)| \leq |(g + 120r_3)|$ and $v^* < v$ are satisfied. Use of these inequalities gives rise to:

$$\begin{aligned}
 |z| &= \sqrt{\frac{(g^* - 120d_3)^2 + (v^*)^2}{(g + 120r_3)^2 + v^2}} \\
 &\leq \sqrt{\frac{(g + 120r_3)^2 + v^2}{(g + 120r_3)^2 + v^2}} \leq 1. \tag{41}
 \end{aligned}$$

Thus, one can say that whenever $\alpha \geq 0.5$, the proposed hybrid approximation (Eq. (33)) is unconditionally stable. When $\alpha < 0.5$, one can estimate that the stability of the present approach depends on the selection of critical time step dt and spatial step h . Thus, as $\alpha < 0.5$, Scheme (33) is conditionally stable.

3.2. Stability of the collocation method

In Section 2.2, the collocation method is introduced to solve the Burgers equation and an iterative matrix equation is obtained as in Eq. (33) with the typical rows of the matrices as follows:

$$\begin{aligned}
 \mathbf{M}^* &= (1, 4, 1), \\
 \mathbf{R}^* &= \frac{3}{h}((- \beta_{s-1} + \beta_{s+1}), 4(- \beta_{s-1} + \beta_{s+1}), \\
 &\quad (- \beta_{s-1} + \beta_{s+1})), \\
 \mathbf{K}^* &= \frac{6}{h^2}(1, -2, 1). \tag{42}
 \end{aligned}$$

By assuming Eq. (33) with the entries of the obtained matrices (Eq. (42)) and considering $\beta_{s+1} - \beta_{s-1} = p$ in matrix \mathbf{R}^* , the following equation can be utilized:

$$c_1\beta_{s-1}^{n+1} + c_2\beta_s^{n+1} + c_3\beta_{s+1}^{n+1} = c_4\beta_{s-1}^n + c_5\beta_s^n + c_6\beta_{s+1}^n, \tag{43}$$

where:

$$\begin{aligned}
 c_1 &= 1 + p_1 - p_2, & c_2 &= 4 + 4p_1 + 2p_2, \\
 c_3 &= 1 + p_1 - p_2, & c_4 &= 1 - k_1 + k_2, \\
 c_5 &= 4 - 4k_1 - 2k_2, & c_6 &= 1 - k_1 + k_2, \\
 p_1 &= \frac{3p\alpha dt}{h}, & p_2 &= \frac{6\varepsilon\alpha dt}{h^2}, \\
 k_1 &= \frac{3p(1-\alpha)dt}{h}, & k_2 &= \frac{6\varepsilon(1-\alpha)dt}{h^2}. \tag{44}
 \end{aligned}$$

Substituting Eq. (35) into Eq. (43) and using the Euler expansion lead to the following expression:

$$(g + 2p_1 + p_2)\tilde{\beta}^{n+1} = (g^* - 2k_1 - k_2)\tilde{\beta}^n, \tag{45}$$

where:

$$\begin{aligned}
 g &= (1 + p_1 - p_2)\cos(kh) + 2, \\
 g^* &= (1 - k_1 + k_2)\cos(kh) + 2.
 \end{aligned}$$

The amplification factor must satisfy the following inequality for the stability of the considered iteration and can be defined as follows:

$$|z| = \sqrt{\frac{(g^* - 2k_1 - k_2)^2}{(g + 2p_1 + p_2)^2}} \leq 1. \tag{46}$$

Whenever $k_1 \leq p_1 \ll 1$ and $k_2 \leq p_2$, i.e. $\alpha \geq 0.5$, the stability condition (Eq. (46)) holds. As a consequence, the collocation scheme for the Burgers equation (Eq. (1)) is unconditionally stable if the parameter α is taken to be $\alpha \geq 0.5$, otherwise the considered scheme is conditionally stable.

3.3. Stability of the Taylor-Galerkin method

In Section 2.3, the Taylor-Galerkin method is analyzed for the numerical solution of the Burgers equation (Eq. (1)) and an iterative approach (Eq. (28)) is constructed. For the stability analysis of this approach, it is required to define typical rows of the matrices in Eq. (28) as follows:

$$\begin{aligned}
 A^* &= \frac{h}{140}(1, 120, 1191, 2416, 1191, 120, 1), \\
 B^* &= \frac{1}{10h}(3, 72, 45, -240, 45, 72, 3), \\
 C^* &= \frac{p}{40}(-3, -168, -735, 0, 735, 168, 3), \\
 D^* &= \frac{p}{40}(-3, -168, -735, 0, 735, 168, 3). \tag{47}
 \end{aligned}$$

A typical row of the septa-diagonal system (Eq. (28)) can be written as:

$$\begin{aligned}
 c_1\beta_{s-3}^{n+1} + c_2\beta_{s-2}^{n+1} + c_3\beta_{s-1}^{n+1} + c_4\beta_s^{n+1} + c_5\beta_{s+1}^{n+1} \\
 + c_6\beta_{s+2}^{n+1} + c_7\beta_{s+3}^{n+1} = c_8\beta_{s-3}^n + c_9\beta_{s-2}^n \\
 + c_{10}\beta_{s-1}^n + c_{11}\beta_s^n + c_{12}\beta_{s+1}^n + c_{13}\beta_{s+2}^n \\
 + c_{14}\beta_{s+3}^n, \tag{48}
 \end{aligned}$$

where:

$$\begin{aligned}
 c_1 &= r_1 - 3r_2 - 3r_3, & c_2 &= 120r_1 - 168r_2 - 72r_3, \\
 c_3 &= 1191r_1 - 735r_2 - 45r_3, & c_4 &= 2416r_1 + 240r_3, \\
 c_5 &= 1191r_1 + 735r_2 - 45r_3, & c_6 &= 120r_1 + 168r_2 - 72r_3, \\
 c_7 &= r_1 + 3r_2 - 3r_3, & c_8 &= r_1 + 3r_3, \\
 c_9 &= 120r_1 + 72r_3, & c_{10} &= 1191r_1 + 45r_3, \\
 c_{11} &= 2416r_1 - 240r_3, & c_{12} &= 1191r_1 + 45r_3, \\
 c_{13} &= 120r_1 + 72r_3, & c_{14} &= r_1 + 3r_3, \\
 r_1 &= \frac{h}{140}, & r_2 &= \frac{pdt}{40}, & r_3 &= \frac{\varepsilon dt}{80h}. \tag{49}
 \end{aligned}$$

Placing Expression (35) into Expression (48) and considering the Euler expansion for exponential terms yield:

$$(g + 120r_3 + iv)\tilde{\beta}^{n+1} = (g^* - 120r_3)\tilde{\beta}^n, \tag{50}$$

where:

$$\begin{aligned}
 g &= (r_1 - 3r_3) \cos(3kh) + (120r_1 - 72r_3) \cos(2kh) \\
 &+ (1191r_1 - 45r_3) \cos(kh) + 1208r_1,
 \end{aligned}$$

$$\begin{aligned}
 g^* &= (r_1 + 3r_3) \cos(3kh) + (120r_1 + 72r_3) \cos(2kh) \\
 &+ (1191r_1 + 45r_3) \cos(kh) + 1208r_1, \\
 v &= 3r_2 \sin(3kh) + 168r_2 \sin(2kh) + 735r_2 \sin(kh).
 \end{aligned}$$

Using Eq. (50), the amplification factor of the obtained iteration (Eq. (28)) can be expressed as:

$$\tilde{\beta}^{n+1} = \frac{(g^* - 120r_3)}{(g + 120r_3 + iv)}\tilde{\beta}^n = z\tilde{\beta}^n. \tag{51}$$

Since $r_1 \ll 1$ in Expression (51), it is obvious that $|z| < 1$. Therefore, the obtained iteration (Eq. (28)) is unconditionally stable.

4. Numerical illustrations

The Galerkin, collocation and Taylor-Galerkin methods are compared and discussed in terms of accuracy and stability for various problems and their parameters. Note that parameter $\alpha = 0.5$, i.e. the Crank-Nicolson scheme, is preferred since the scheme is more accurate than other selections and satisfies the stability conditions in the Galerkin and collocation methods.

Example 1 [23]. Consider the Burgers equation (Eq. (1)) with initial condition:

$$u(x, 0) = \sin \pi x, \quad 0 < x < 1, \tag{52}$$

and the homogenous boundary conditions:

$$u(0, t) = 0, \quad t > 0, \tag{53}$$

$$u(1, t) = 0, \quad t > 0. \tag{54}$$

The exact solution of Eq. (1) under the consideration of cases (Eqs. (52)–(54)) given by Cole [9] is:

$$u(x, t) = 2\pi\varepsilon \frac{\sum_{n=1}^{\infty} a_n \exp(-n^2 \pi^2 \varepsilon t) n \sin(n\pi x)}{a_0 + \sum_{n=1}^{\infty} a_n \exp(-n^2 \pi^2 \varepsilon t) \cos(n\pi x)}, \tag{55}$$

with the following Fourier coefficients:

$$a_0 = \int_0^1 \exp\left\{-(2\pi\varepsilon)^{-1} [1 - \cos(\pi x)]\right\} dx, \tag{56}$$

$$a_n = 2 \int_0^1 \exp\left\{-(2\pi\varepsilon)^{-1} [1 - \cos(\pi x)]\right\} \cos(n\pi x) dx. \tag{57}$$

In this example, comparison of the currently produced results with the results of the literature [30] and exact solutions has been carried out in Table 2, for various spatial points at $t = 0.5$. As clearly seen in the table, the present methods are more accurate and more economical, even with fewer elements, in comparison to their results. Note also that Taylor-Galerkin Finite Element Method (TGFEM) and Galerkin Finite

Element Method (GFEM) have more accurate results than the Collocation Finite Element Method (CFEM) as demonstrated in Table 2. The presently produced solutions of those techniques have been compared with the literature [30] and the exact solution as seen in Table 3. The results have been computed for the parameter values $dt = 0.01$, $h = 0.01$, and $\varepsilon = 0.1$. The present numerical techniques have been realized to produce far more accurate results than the literature [30]. Among current methods, the TGFEM has been seen to give more acceptable results than for the others.

The results presented in Table 4 are compared with the literature [1,14] and the exact solution. Even with the use of fewer time elements, the comparison revealed that the suggested techniques generally produce more accurate results than the corresponding literature [1,14]. In the comparison, various time steps

are used for the elapsed time of $t = 4$ at different positions. All results are produced for $\varepsilon = 0.01$ and $h = 0.01$ in Table 4.

Now it is time to deal with the smaller kinematic viscosity constants. Comparison of the currently produced solutions has been carried out with the literature and the exact solution for two different viscosity values, in Tables 5 and 6, i.e., $\varepsilon = 0.004$ and $\varepsilon = 0.003$, respectively. Jiwari [11] used a hybrid approximation to also solve the problem. The present study revealed that, even with the use of fewer time elements, one can find similar or sometimes more accurate results than in the literature [11].

In the work of Aksan [31], the governing equation is solved using the quadratic B-spline FEM in the weak form with the Newton iteration for non-linear systems. The computed results in Table 7 reveal that the present method is more accurate with fewer time elements than

Table 2. Comparison of the results produced with $\varepsilon = 1$, $dt = 0.001$, and $t = 0.5$ in Example 1.

x	CFEM $N = 40$	TGFEM $N = 40$	GFEM $N = 40$	BDF3 [30] $N = 100$	Exact
$x = 0.1$	0.002207	0.002213	0.002213	0.002213	0.002213
$x = 0.2$	0.004199	0.004210	0.004210	0.004209	0.004210
$x = 0.3$	0.005781	0.005796	0.005796	0.005795	0.005796
$x = 0.4$	0.006798	0.006816	0.006816	0.006815	0.006816
$x = 0.5$	0.007151	0.007169	0.007169	0.007168	0.007169
$x = 0.6$	0.006803	0.006820	0.006820	0.006820	0.006821
$x = 0.7$	0.005789	0.005804	0.005804	0.005803	0.005804
$x = 0.8$	0.004207	0.004218	0.004218	0.004217	0.004218
$x = 0.9$	0.002212	0.002218	0.002218	0.002218	0.002218

Table 3. Comparison of the results produced with $\varepsilon = 0.1$, $dt = 0.01$, and $t = 2.3$ in Example 1.

x	CFEM $N = 100$	TGFEM $N = 100$	GFEM $N = 100$	BDF3 [30] $N = 100$	Exact
$x = 0.1$	0.0221378	0.0221395	0.0221389	0.02253	0.0221396
$x = 0.2$	0.0427917	0.0427953	0.0427942	0.04357	0.0427956
$x = 0.3$	0.0604248	0.0604307	0.0604292	0.06155	0.0604313
$x = 0.4$	0.0734337	0.0734423	0.0734405	0.07485	0.0734431
$x = 0.5$	0.0802187	0.0802298	0.0802280	0.08182	0.0802310
$x = 0.6$	0.0793843	0.0793974	0.0793957	0.08104	0.0793988
$x = 0.7$	0.0700918	0.0701053	0.0701039	0.07161	0.0701068
$x = 0.8$	0.0525071	0.0525185	0.0525175	0.05368	0.0525198
$x = 0.9$	0.0281666	0.0281733	0.0281728	0.02881	0.0281740

Table 4. Comparison of the results produced with $\varepsilon = 0.01$ and $h = 0.01$ at various times in Example 1.

x	t	CFEM	TGFEM	GFEM	Ref. [1]	Ref. [14]	Exact
		$dt = 0.004$	$dt = 0.004$	$dt = 0.004$	$dt = 0.001$	$dt = 0.001$	
$x = 0.1$	$t = 0.50$	0.1211626	0.1211435	0.1211416	0.12079	0.12114	0.1211435
	$t = 2.00$	0.0429683	0.0429638	0.0429634	0.04300	0.04295	0.0429638
	$t = 4.00$	0.0231057	0.0231042	0.0231041	0.02324	0.02310	0.0231042
$x = 0.3$	$t = 0.50$	0.3602699	0.3602713	0.3602668	0.36113	0.36027	0.3602711
	$t = 2.00$	0.1288446	0.1288399	0.1288389	0.12887	0.12882	0.1288399
	$t = 4.00$	0.0693104	0.0693083	0.0693080	0.06935	0.06930	0.0693083
$x = 0.5$	$t = 0.50$	0.5886937	0.5886975	0.5886941	0.59559	0.58870	0.5886958
	$t = 2.00$	0.2145611	0.2145581	0.2145565	0.21468	0.21455	0.2145581
	$t = 4.00$	0.1154973	0.1154948	0.1154943	0.11550	0.11549	0.1154948
$x = 0.7$	$t = 0.50$	0.7935009	0.7934995	0.7935022	0.81257	0.79354	0.7934934
	$t = 2.00$	0.2999991	0.2999979	0.2999958	0.30075	0.29999	0.2999978
	$t = 4.00$	0.1612261	0.1612147	0.1612140	0.16125	0.16121	0.1612147
$x = 0.9$	$t = 0.50$	0.9382118	0.9381275	0.9381339	0.97184	0.93822	0.9381083
	$t = 2.00$	0.3734554	0.3732775	0.3732749	0.37452	0.37328	0.3732776
	$t = 4.00$	0.1659905	0.1660587	0.1660577	0.16515	0.16605	0.1660587

Table 5. Comparison of the results produced with $\varepsilon = 0.004$ and $h = 0.01$ for various values of t in Example 1.

x	t	GFEM	TGFEM	CFEM	Ref. [11]	Exact
		$dt = 0.0125$	$dt = 0.0125$	$dt = 0.0125$	$dt = 0.001$	
$x = 0.25$	$t = 1$	0.18889	0.18890	0.18893	0.18891	0.18889
	$t = 5$	0.04698	0.04697	0.04698	0.04697	0.04697
	$t = 10$	0.02422	0.02422	0.02422	0.02422	0.02421
	$t = 15$	0.01632	0.01632	0.01632	0.01632	0.01631
$x = 0.5$	$t = 1$	0.37594	0.37598	0.37603	0.37598	0.37596
	$t = 5$	0.09394	0.09394	0.09395	0.09394	0.09393
	$t = 10$	0.04844	0.04844	0.04844	0.04843	0.04843
	$t = 15$	0.03260	0.03259	0.03260	0.03259	0.03259
$x = 0.75$	$t = 1$	0.55880	0.55885	0.55892	0.55883	0.55881
	$t = 5$	0.14089	0.14089	0.14090	0.14089	0.14089
	$t = 10$	0.07221	0.07220	0.07221	0.07221	0.07220
	$t = 15$	0.04678	0.04678	0.04678	0.04678	0.04677

the ones in references [25,31]. In the related references, they used the weak form of the governing equation, the trigonometric cubic B-spline basis approach and the splitting approach, respectively. For the same problem, the suggested methods here have produced

more accurate results in comparison to their work (see Table 7).

In Figures 1–3, the absolute error norms of the considered numerical approaches for Example 1 are compared for various values of the viscosity constant

Table 6. Comparison of the results produced with $\varepsilon = 0.003$ and $h = 0.01$ for various values of t in Example 1.

x	t	CFEM	TGFEM	GFEM	Ref. [11]	Exact
		$dt = 0.01$	$dt = 0.01$	$dt = 0.01$	$dt = 0.001$	
$x = 0.25$	$t = 1$	0.18902	0.18902	0.18904	0.18902	0.18901
	$t = 5$	0.04699	0.04698	0.04698	0.04698	0.04698
	$t = 10$	0.02422	0.02422	0.02422	0.02422	0.02422
	$t = 15$	0.01632	0.01632	0.01632	0.01631	0.01631
$x = 0.5$	$t = 1$	0.37620	0.37623	0.37628	0.37623	0.37619
	$t = 5$	0.09396	0.09396	0.09396	0.09396	0.09395
	$t = 10$	0.04845	0.04844	0.04845	0.04844	0.04843
	$t = 15$	0.03263	0.03263	0.03263	0.03263	0.03263
$x = 0.75$	$t = 1$	0.55926	0.55929	0.55935	0.55928	0.55924
	$t = 5$	0.14092	0.14092	0.14093	0.14092	0.14095
	$t = 10$	0.07261	0.07260	0.07261	0.07261	0.07260
	$t = 15$	0.04839	0.04839	0.04839	0.04839	0.04841

Table 7. Comparison of the results produced with $\varepsilon = 0.1$ and $h = 0.0125$ for various values of t in Example 1.

x	t	CFEM	TGFEM	GFEM	Ref. [25]	Ref. [31]	Exact
		$dt = 0.001$	$dt = 0.001$	$dt = 0.001$	$dt = 0.0001$	$dt = 0.0001$	
$x = 0.25$	$t = 0.4$	0.30890	0.30889	0.30889	0.30890	0.30891	0.30889
	$t = 0.6$	0.24075	0.24074	0.24074	0.24074	0.24075	0.24074
	$t = 0.8$	0.19569	0.19568	0.19568	0.19568	0.19568	0.19568
	$t = 1.0$	0.16258	0.16256	0.16256	0.16257	0.16257	0.16256
	$t = 3.0$	0.02720	0.02720	0.02720	0.02720	0.02721	0.02720
$x = 0.50$	$t = 0.4$	0.56965	0.56963	0.56963	0.56964	0.56969	0.56963
	$t = 0.6$	0.44723	0.44721	0.44721	0.44721	0.44723	0.44721
	$t = 0.8$	0.35925	0.35924	0.35924	0.35924	0.35926	0.35924
	$t = 1.0$	0.29192	0.29192	0.29192	0.29191	0.29193	0.29192
	$t = 3.0$	0.04019	0.04020	0.04020	0.04020	0.04021	0.04021
$x = 0.75$	$t = 0.4$	0.62537	0.62544	0.62544	0.62541	0.62543	0.62544
	$t = 0.6$	0.48713	0.48722	0.48722	0.48719	0.48723	0.48721
	$t = 0.8$	0.37384	0.37392	0.37392	0.37390	0.37394	0.37392
	$t = 1.0$	0.28741	0.28747	0.28747	0.28746	0.28750	0.28747
	$t = 3.0$	0.02976	0.02977	0.02977	0.02977	0.02978	0.02977

ε , $dt = 0.001$ and $h = 0.025$ at $t = 1$. As seen in the figures, the Taylor-Galerkin method produces less error than the other present methods. Under the same parameters, it can be concluded that the selection of the viscosity constant affects the amount of errors of the present schemes with inverse proportionally.

Note that the behaviour of the solution of the Burgers equation for various values of viscosity constant ε has been discussed in the literature, e.g. Sari and Gurarslan [1]. The physical behaviour of the process calculated by the GFEM and TGFEM in terms of the viscosity constant exhibits the expected

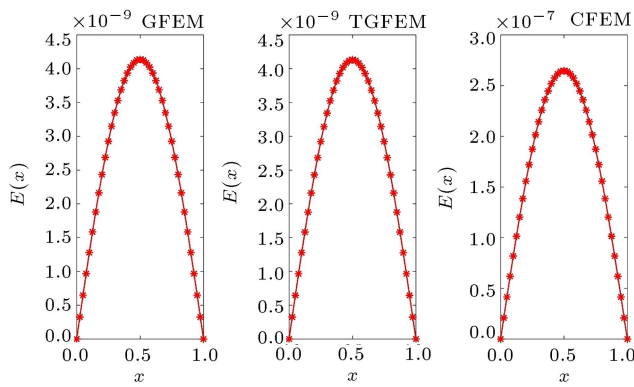


Figure 1. Comparison of the present methods in terms of absolute errors produced with $\varepsilon = 1$, $dt = 0.001$ $h = 0.025$, and $t = 1$ in Example 1.

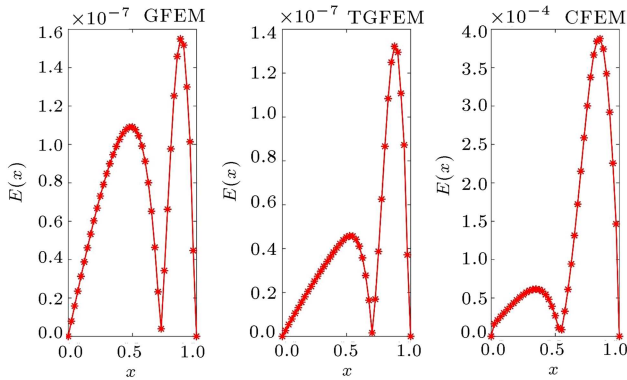


Figure 2. Comparison of the present methods in terms of absolute errors produced with $\varepsilon = 0.1$, $dt = 0.001$, $h = 0.025$, and $t = 1$ in Example 1.

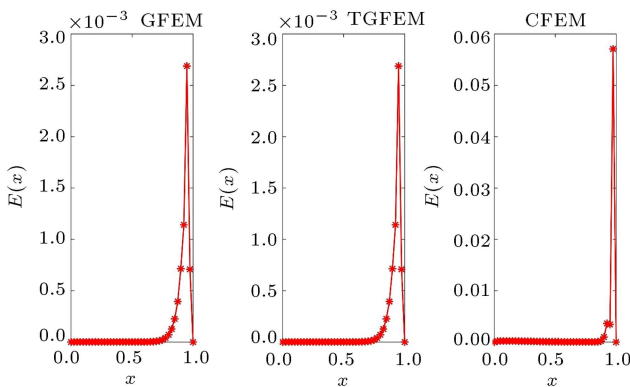


Figure 3. Comparison of the present methods in terms of absolute errors with $\varepsilon = 0.01$, $dt = 0.001$, $h = 0.025$, and $t = 1$ in Example 1.

physical characteristics of the problem as demonstrated in Figures 4–6. Even though the TGFEM generally produces more accurate results, it may produce small unwanted oscillations in advection dominated cases as in Figure 5.

Example 2 [32]. Let us now consider the Burgers equation (Eq. (1)) with the initial condition:

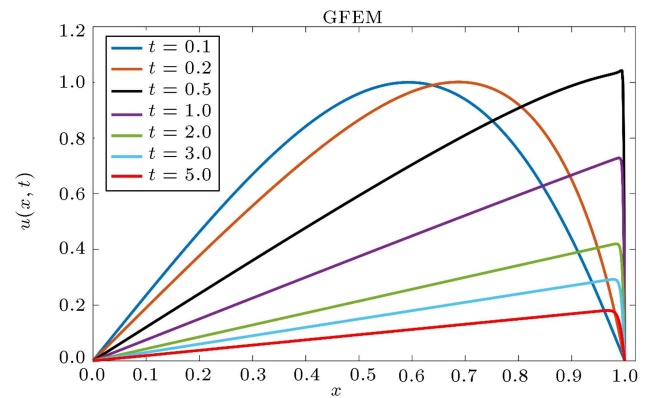


Figure 4. The Galerkin Finite Element Method (GFEM) solution of Example 1 at different times with $\varepsilon = 0.001$, $h = 0.0016$, and $dt = 0.1$.

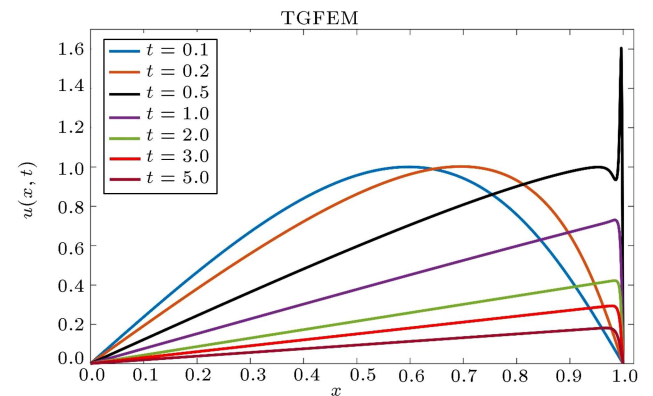


Figure 5. The Taylor-Galerkin Finite Element Method (TGFEM) solution of Example 1 at different times with $\varepsilon = 0.001$, $h = 0.0016$, and $dt = 0.1$.

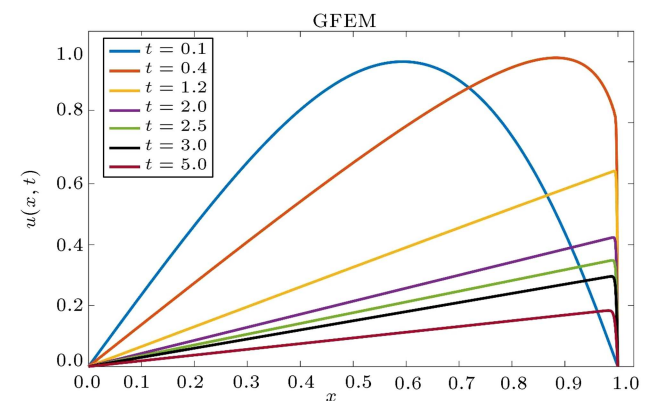


Figure 6. The Galerkin Finite Element Method (GFEM) solution of Example 1 at different times with $\varepsilon = 0.0005$, $h = 0.0015$, and $dt = 0.1$.

$$u(x, 0) = g(x) = 4x(1 - x), \quad 0 < x < 1, \quad (58)$$

and the homogenous Dirichlet boundary conditions:

$$u(0, t) = 0, \quad t > 0, \quad (59)$$

$$u(1, t) = 0, \quad t > 0. \quad (60)$$

Table 8. Comparison of the produced results with $\varepsilon = 1$ for various times in Example 2.

x	t	CFEM	TGFEM	GFEM	Ref. [4]	Ref. [33]	Exact
		$h = 0.01,$ $dt = 0.0002$	$h = 0.01,$ $dt = 0.0002$	$h = 0.01,$ $dt = 0.0002$	$h = 0.0125,$ $dt = 0.0001$	$h = 0.25$ $dt = 0.0001$	
$x = 0.25$	$t = 0.05$	0.4262723	0.4262855	0.4262854	0.42629	0.4262864	0.4262855
	$t = 0.10$	0.2614633	0.2614797	0.2614797	0.26149	0.2614801	0.2614797
	$t = 0.15$	0.1614605	0.1614775	0.1614775	0.16148	0.1614777	0.1614776
	$t = 0.25$	0.0610757	0.0610875	0.0610875	0.06109	0.0610875	0.0610875
$x = 0.50$	$t = 0.05$	0.6280605	0.6280837	0.6280837	0.62809	0.6280846	0.6280837
	$t = 0.10$	0.3833919	0.3834223	0.3834223	0.38343	0.3834228	0.3834224
	$t = 0.15$	0.2340271	0.2340552	0.2340552	0.23406	0.2340554	0.2340553
	$t = 0.25$	0.0872150	0.0872326	0.0872326	0.08724	0.0872327	0.0872327
$x = 0.75$	$t = 0.05$	0.4652268	0.4652526	0.4652526	0.46526	0.4652528	0.4652526
	$t = 0.10$	0.2815447	0.2815726	0.2815726	0.28158	0.2815727	0.2815726
	$t = 0.15$	0.1697150	0.1697382	0.1697382	0.16974	0.1697383	0.1697382
	$t = 0.25$	0.0622768	0.0622898	0.0622898	0.06229	0.0622898	0.0622898

Exact solution of Eq. (1) under the consideration of the cases (Eqs. (58)–(60)) given by Cole [9] has the form of Eq. (55) with the Fourier coefficients:

$$a_0 = \int_0^1 \exp \{ -x^2(3\varepsilon)^{-1}(3 - 2x) \} dx, \tag{61}$$

$$a_n = 2 \int_0^1 \exp \{ -x^2(3\varepsilon)^{-1}(3 - 2x) \} \cos(n\pi x) dx. \tag{62}$$

Table 8 includes the comparison of numerical solutions and exact solution with kinematic viscosity $\varepsilon = 1$. The produced results in Table 8 are seen to be more accurate than the literature [4,33]. To produce those results, even less elements, in comparison to the corresponding references, have been used both in time and in space. The computed results have also been compared with the literature [22,24] and exact solution as seen in Table 9. The current numerical solutions are more accurate than the corresponding literature when the advection is relatively more dominant than the diffusion, $\varepsilon = 0.01$, and $h = 0.0125$. The presently calculated solutions are seen to require less effort in time comparison to the references.

Table 10 is organized to present the numerical results for $\varepsilon = 0.01$, $dt = 0.002$ and $h = 0.01$ with various numbers of spatial and time nodes. The present solutions have been compared with the exact and other numerical solutions, based on various numerical methods, which are finite difference [1] and boundary element [6] methods. The present solutions revealed

that less time effort and less computational time are needed to catch high accuracy as compared to the previously mentioned effective methods.

As for the comparison of the present numerical results with the exact solution and reference [11] with small viscosity values $\varepsilon = 0.004$ and $\varepsilon = 0.003$ in Tables 11 and 12, respectively, the current results are seen to agree with some other numerical solutions [11]. As underlined for various times, an even less number of time elements are seen to be enough to have highly accurate solutions.

In Figures 7 and 8, the present numerical solutions are compared with each other for the parameters $\varepsilon = 1$, $\varepsilon = 0.1$, $h = 0.025$ and $dt = 0.001$ at $t = 1$. As seen in the figures, the GFEM and TGFEM produce less absolute error than the CFEM.

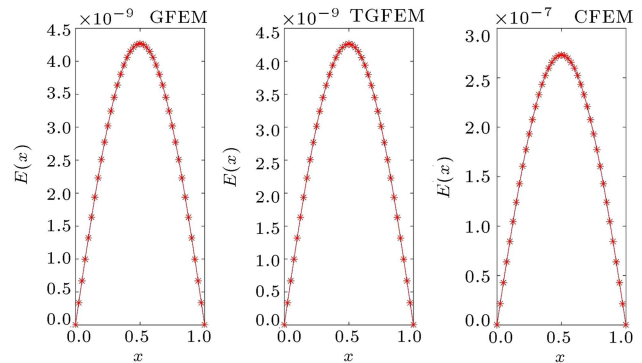


Figure 7. Comparison of the present methods in terms of absolute errors produced for $\varepsilon = 1$, $dt = 0.001$, $h = 0.025$, and $t = 1$ in Example 2.

Table 9. Comparison of the results produced with $\varepsilon = 0.1$ and $h = 0.0125$ for various times in Example 2.

x	t	CFEM	TGFEM	GFEM	Ref. [22]	Ref. [24]	Exact
		$dt = 0.001$	$dt = 0.001$	$dt = 0.001$	$dt = 0.0001$	$dt = 0.0001$	
$x = 0.25$	$t = 0.4$	0.3175299	0.3175228	0.3175226	0.32091	0.31749	0.3175229
	$t = 0.6$	0.2461491	0.2461384	0.2461383	0.24910	0.24612	0.2461385
	$t = 0.8$	0.1995677	0.1995553	0.1995552	0.20211	0.19954	0.1995553
	$t = 1.0$	0.1656106	0.1655986	0.1655986	0.16782	0.16559	0.1655986
	$t = 3.0$	0.0277529	0.0277587	0.0277587	0.02828	0.02776	0.0277587
$x = 0.50$	$t = 0.4$	0.5845555	0.5845372	0.5845372	0.58788	0.58448	0.5845373
	$t = 0.6$	0.4579982	0.4579764	0.4579763	0.46174	0.45793	0.4579764
	$t = 0.8$	0.3674150	0.3673982	0.3673981	0.37111	0.36736	0.3673982
	$t = 1.0$	0.2983514	0.2983431	0.2983430	0.30183	0.29831	0.2983431
	$t = 3.0$	0.0410543	0.0410650	0.0410650	0.04185	0.04106	0.0410650
$x = 0.75$	$t = 0.4$	0.6455490	0.6456157	0.6456160	0.65054	0.64547	0.6456155
	$t = 0.6$	0.5025960	0.5026758	0.5026760	0.50825	0.50255	0.5026758
	$t = 0.8$	0.3852565	0.3853355	0.3853356	0.39068	0.38523	0.3853355
	$t = 1.0$	0.2957873	0.2958567	0.2958567	0.30057	0.29578	0.2958567
	$t = 3.0$	0.0304300	0.0304396	0.0304396	0.03106	0.03044	0.0304396

Table 10. Comparison of the results produced with $\varepsilon = 0.01$ and $h = 0.01$ for various times in Example 2.

x	t	CFEM	TGFEM	GFEM	Ref. [6]	Ref. [1]	Exact
		$dt = 0.002$	$dt = 0.002$	$dt = 0.002$	$dt = 0.001$	$dt = 0.001$	
$x = 0.10$	$t = 0.50$	0.128486	0.128462	0.128462	0.12808	0.12846	0.128462
	$t = 2.00$	0.043819	0.043814	0.043814	0.04388	0.04379	0.043814
	$t = 4.00$	0,023347	0,023345	0,023345	0.02351	0.02334	0,023345
$x = 0.30$	$t = 0.50$	0,378492	0,378489	0,378488	0.37956	0.37849	0,378489
	$t = 2.00$	0,131351	0,131345	0,131345	0.13129	0.13131	0,131345
	$t = 4.00$	0,070030	0,070027	0,070027	0.07009	0.07002	0,070027
$x = 0.50$	$t = 0.50$	0,609886	0,609887	0,609886	0.61768	0.60991	0,609886
	$t = 2.00$	0.218593	0.218588	0.218588	0.21873	0.21858	0.218588
	$t = 4.00$	0.116685	0.116682	0.116682	0.11671	0.11667	0.116682
$x = 0.70$	$t = 0.50$	0.809785	0.809783	0.809784	0.83022	0.80986	0.809782
	$t = 2.00$	0.305352	0.305348	0.305348	0.30614	0.30534	0.305348
	$t = 4.00$	0.162890	0.162878	0.162878	0.16293	0.16287	0.162878
$x = 0.90$	$t = 0.50$	0.946090	0.946017	0.946019	0.98068	0.94615	0.946014
	$t = 2.00$	0.380461	0.380273	0.380273	0.38163	0.38027	0.380274
	$t = 4.00$	0.168511	0.168577	0.168577	0.16766	0.16857	0.168577

The parameter α also affects the accuracy of the GFEM and the CFEM as stated in Section 2. In Figure 9, we use various values of parameter α to compare L_2 errors produced by the GFEM using $\varepsilon = 1$ and $h = 0.0125$. As seen in Figure 9, for small

values of the parameter dt , the case $\alpha = 0.5$ is more acceptable.

As seen in Figure 10, the numerical solution of Example 2 varies with the values of the viscosity constant ε . The numerical solution has a steep behavior

Table 11. Comparison of the results produced with $\varepsilon = 0.004$ and $h = 0.01$ for various times in Example 2.

x	t	CFEM	TGFEM	GFEM	Ref. [11]	Exact
		$dt = 0.01$	$dt = 0.01$	$dt = 0.01$	$dt = 0.001$	
$x = 0.25$	$t = 1$	0.19639	0.19639	0.19638	0.19636	0.19641
	$t = 5$	0.04744	0.04744	0.04744	0.04744	0.04744
	$t = 10$	0.02434	0.02434	0.02434	0.02434	0.02434
	$t = 15$	0.01637	0.01637	0.01637	0.01637	0.01637
$x = 0.50$	$t = 1$	0.38847	0.38849	0.38847	0.38842	0.38846
	$t = 5$	0.09486	0.09486	0.09486	0.09491	0.09493
	$t = 10$	0.04869	0.04868	0.04868	0.04868	0.04869
	$t = 15$	0.03271	0.03271	0.03271	0.03270	0.03270
$x = 0.75$	$t = 1$	0.57318	0.57321	0.57319	0.57312	0.57315
	$t = 5$	0.14225	0.14225	0.14224	0.14224	0.14225
	$t = 10$	0.07259	0.07258	0.07258	0.07258	0.07257
	$t = 15$	0.04697	0.04696	0.04696	0.04696	0.04695

Table 12. Comparison of the results produced with $\varepsilon = 0.003$ and $h = 0.01$ for various times in Example 2.

x	t	CFEM	TGFEM	GFEM	Ref. [11]	Exact
		$dt = 0.01$	$dt = 0.01$	$dt = 0.01$	$dt = 0.001$	
$x = 0.25$	$t = 1$	0.19673	0.19672	0.19672	0.19668	0.19673
	$t = 5$	0.04747	0.04746	0.04746	0.04746	0.04748
	$t = 10$	0.02435	0.02435	0.02435	0.02434	0.02434
	$t = 15$	0.01638	0.01638	0.01638	0.01637	0.01637
$x = 0.50$	$t = 1$	0.38896	0.38897	0.38896	0.38890	0.38894
	$t = 5$	0.09492	0.09491	0.09491	0.09491	0.09494
	$t = 10$	0.04870	0.04870	0.04870	0.04870	0.04871
	$t = 15$	0.03275	0.03275	0.03275	0.03274	0.03274
$x = 0.75$	$t = 1$	0.57383	0.57383	0.57383	0.57375	0.57378
	$t = 5$	0.14233	0.14232	0.14232	0.14232	0.14234
	$t = 10$	0.07299	0.07299	0.07299	0.07298	0.07297
	$t = 15$	0.04858	0.04857	0.04857	0.04857	0.04857

for the small values of parameter ε . Figures 11 and 12 show the solutions of the GFEM and the TGFEM of Example 2 for $\varepsilon = 0.001$. The GFEM has the ability to capture the steep behaviour for the considered advection dominated case while the TGFEM solution has small unwanted oscillations. Even though the TGFEM seems to have more accurate results than the GFEM, the unwanted oscillations come out in the GFEM later than the TGFEM.

Example 3 [34]. Let us consider the Burgers equation (Eq. (1)) with the initial condition:

$$u(x, 0) = 1 - \tanh\left(\frac{x}{2\varepsilon}\right), \tag{63}$$

and the nonhomogeneous and time dependent boundary conditions:

$$u(0.5, t) = 1 - \tanh\left(\frac{0.5 - t}{2\varepsilon}\right), \tag{64}$$

$$u(1.5, t) = 1 - \tanh\left(\frac{1.5 - t}{2\varepsilon}\right). \tag{65}$$

The exact solution to the problem is:

$$u(x, t) = 1 - \tanh\left(\frac{x - t}{2\varepsilon}\right). \tag{66}$$

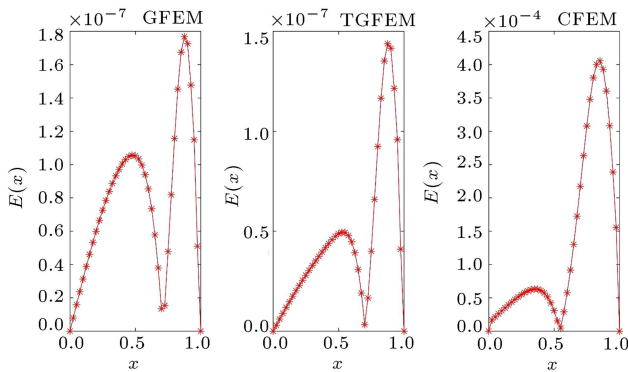


Figure 8. Comparison of the present methods for $\varepsilon = 0.1$, $dt = 0.001$, $h = 0.025$, and $t = 1$ in Example 2.

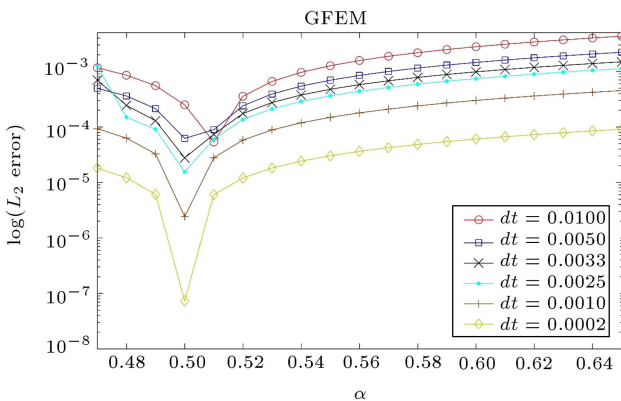


Figure 9. Comparison of L_2 error norms of the GFEM solutions for various values of parameter α .

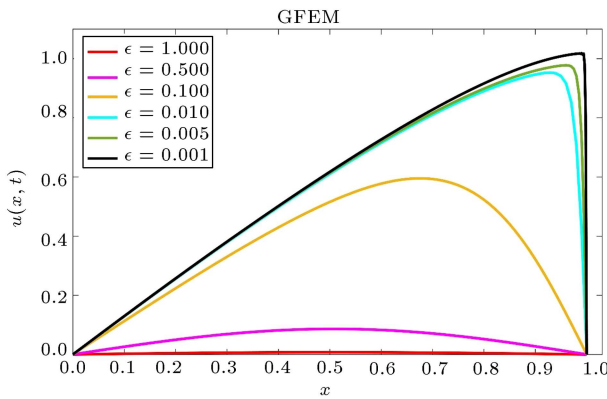


Figure 10. The Galerkin Finite Element Method (GFEM) solution of the problem at $t = 0.5$ with various kinematic viscosity constants for: $\varepsilon = 1$, $\varepsilon = 0.5$, $\varepsilon = 0.1$, $\varepsilon = 0.01$, $\varepsilon = 0.005$, and $\varepsilon = 0.001$.

The numerical capabilities of the present techniques have been demonstrated through their pointwise errors, for various kinematic viscosity values, as seen in Figures 13–16. Unlike the previous examples, even when time dependent boundary conditions are accepted, the computed results still show high order of accuracy. In all figures, the parameter values $dt = 0.0005$, $h = 0.01$ and $t = 0.5$ are considered to evaluate absolute errors.

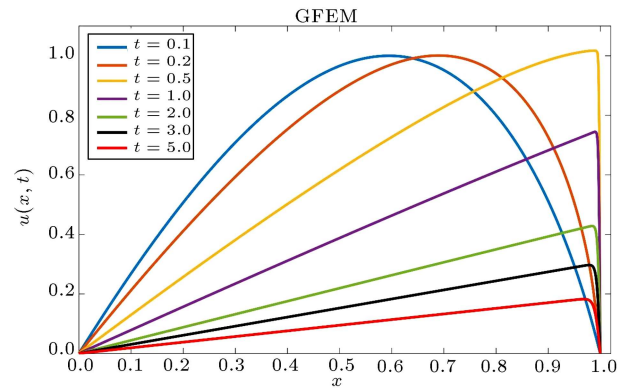


Figure 11. The Galerkin Finite Element Method (GFEM) solution of Example 2 at different times produced for $\varepsilon = 0.001$, $h = 0.0016$, and $dt = 0.1$.

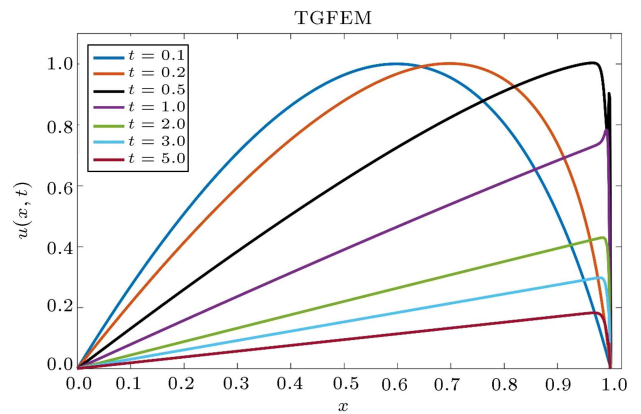


Figure 12. The Taylor-Galerkin Finite Element Method (TGFEM) solution of Example 2 at different times produced for $\varepsilon = 0.001$, $h = 0.0016$, and $dt = 0.1$.

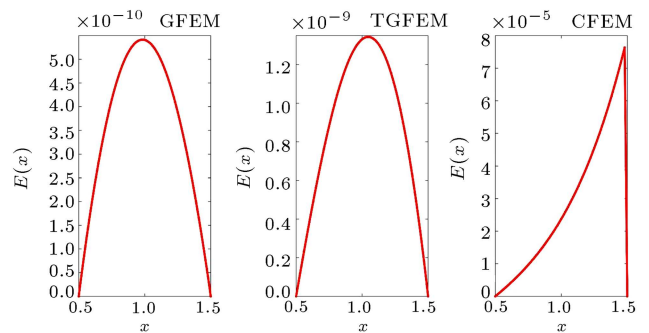


Figure 13. Comparison of the present methods in terms of absolute errors produced for $\varepsilon = 1$, $dt = 0.0005$, $h = 0.01$, and $t = 0.5$ in Example 3.

Example 4. Let us consider the Burgers equation with the following discontinuous initial condition:

$$u(x, 0) = \begin{cases} 2, & x < 0 \\ 1, & x \geq 0 \end{cases} \quad (67)$$

and the nonhomogeneous boundary conditions:

$$u(-1, t) = 2, \quad \text{and} \quad u(1, t) = 1. \quad (68)$$

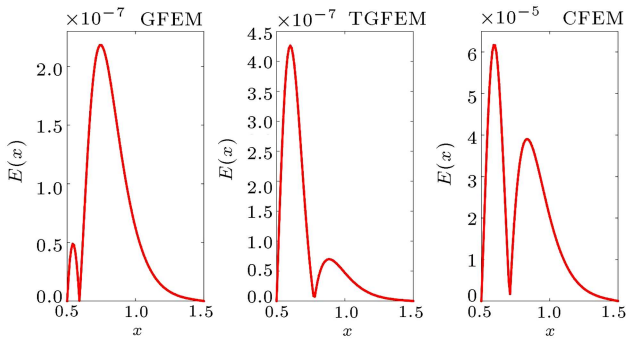


Figure 14. Comparison of the present methods in terms of absolute errors produced for $\varepsilon = 0.1$, $dt = 0.0005$, $h = 0.01$, and $t = 0.5$ in Example 3.

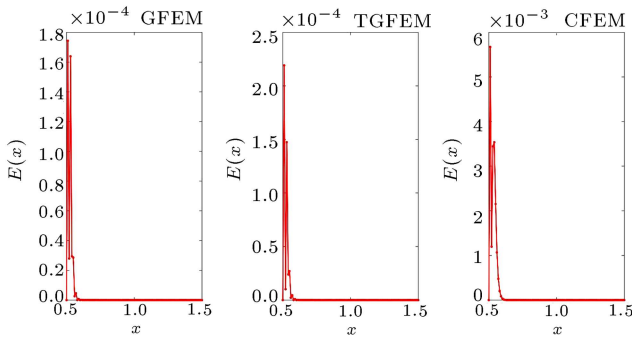


Figure 15. Comparison of the present methods in terms of absolute errors produced for $\varepsilon = 0.01$, $dt = 0.0005$, $h = 0.01$, and $t = 0.5$ in Example 3.

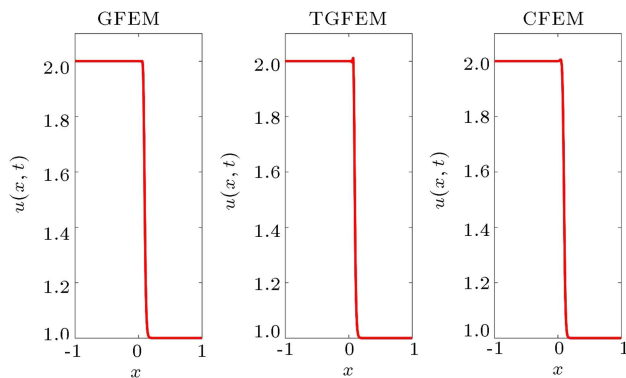


Figure 16. Nonlinear advection-diffusion process solved by present methods for $\varepsilon = 0.005$, $dt = 0.01$, $h = 0.005$, and $t = 0.5$ in Example 4.

In Figure 16, the physical behaviour of the problem given in Example 4 is exhibited by using the current numerical techniques with the parameter values $\varepsilon = 0.005$, $dt = 0.01$, $h = 0.005$ and $t = 0.5$. As seen in the figure, shock behavior is physically captured by the present techniques, except for a very small deviation of the TGFEM around the discontinuity. The advection-dominated process of Example 4 is illustrated with the use of the GFEM in Figure 17. Propagation of the shock behaviours is shown for various kinematic

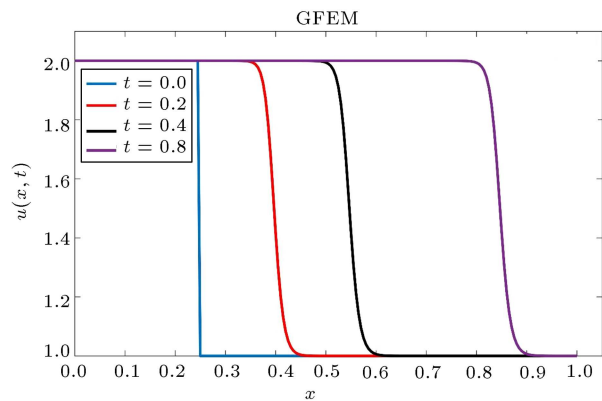


Figure 17. Nonlinear advection-diffusion process solved by the Galerkin Finite Element Method (GFEM) for $\varepsilon = 0.01$, $dt = 0.008$, and $h = 0.005$ in Example 4.

viscosity values as seen in Figure 18. To achieve this, the GFEM is used with the parameter values $dt = 0.006$, $h = 0.005$ and $t = 0.6$. As realized from the figure, the shock behaviors are clearly captured for the cases of $\varepsilon = 0.01$ and $\varepsilon = 0.005$, but little oscillations take place for the case of $\varepsilon = 0.001$. Thus, the present methods are seen to be capable of producing oscillation free results for the discontinuous initial condition, in the case of $\varepsilon > 0.001$.

5. Concluding remarks

Finite element based hybrid methods have been proposed to numerically analyze the nonlinear advection diffusion equation in this article. It has been shown that the Galerkin and the collocation methods are unconditionally stable for $\alpha \geq 0.5$, as well as showing that the Taylor-Galerkin method is unconditionally stable for any case. Note that the present approaches have been illustrated to be capable of solving the nonlinear advection diffusion models and also be capable of producing highly accurate solutions with fewer numbers of elements used in both time and space for the advection-dominant case. Notice that the proposed hybrid approaches give convergent approximations, and is seen to be a very good choice to achieve a high degree of accuracy while dealing with the nonlinear advection-diffusion problems. The produced results have also been seen to be more accurate than some available results in the literature.

The Taylor-Galerkin Finite Element Method (TGFEM) has found to be more accurate than both the Collocation Finite Element Method (CFEM) and the Galerkin Finite Element Method (GFEM) for almost all cases of the Burgers equation problem. Even though the CFEM is not as accurate as the GFEM and the TGFEM, the CFEM is of computationally low-cost. It is illustrated that the GFEM is more capable of capturing the behavior of the advection-

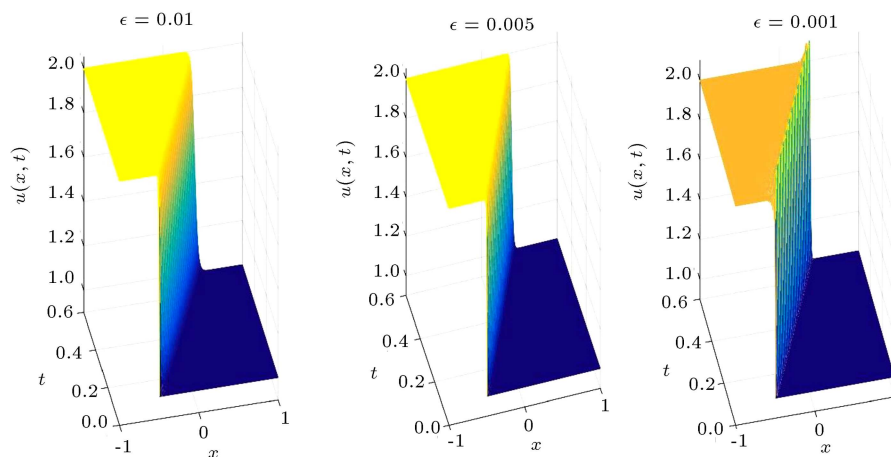


Figure 18. Comparison of various cases of the nonlinear advection-diffusion process solved by the Galerkin Finite Element Method (GFEM) for $dt = 0.006$, $h = 0.005$, and $t = 0.6$ in Example 4.

dominated cases of the model equation. Further studies can investigate the generalization for designing of the current methods to physical processes described by any nonlinear time-dependent partial differential equation.

Acknowledgment

The first author would like to thank the Science Fellowships and Grant Programmes Department of TUBITAK (TUBITAK BİDEB) for their support of this academic research.

References

- Sari, M. and Gurarslan, G. "A sixth-order compact finite difference scheme to the numerical solutions of Burgers' equation", *Appl Math Comput.*, **208**, pp. 475–483 (2009).
- Wang, J. and Warnecke, G. "Existence and uniqueness of solutions for a non-uniformly parabolic equation", *J. Differ. Equations*, **189**, pp. 1–16 (2003).
- Miller, E.L. "Predictor-corrector studies of Burger's model of turbulent flow", MS Thesis, University of Delaware, Newark, Delaware (1966).
- Kutluay, S., Bahadir, A.R., and Ozdes, A. "Numerical solution of one-dimensional Burgers' equation: explicit and exact-explicit finite difference methods", *J. Comput. Appl. Math.*, **103**, pp. 251–261 (1999).
- Seydaoglu, M. "An accurate approximation algorithm for Burgers' equation in the presence of small viscosity", *J. Comput. Appl. Math.*, **344**, pp. 473–481 (2018).
- Bahadir, A.R. and Saglam, M. "A mixed finite difference and boundary element approach to one-dimensional Burgers' equation", *Appl. Math. Comput.*, **160**, pp. 663–673 (2005).
- Sari, M., Tunc, H., and Seydaoglu M. "Higher order splitting approaches in analysis of the Burgers equation", *Kuwait J. Sci.*, **46**(1), pp. 1–14 (2019).
- Hopf, E. "The partial differential equation", *Commun. Pur. Appl. Math.*, **9**, pp. 201–230 (1950).
- Cole, J.D. "On a quasi-linear parabolic equation in aerodynamics", *Q. Appl. Math.*, **9**, pp. 225–236 (1951).
- Talwar, J., Mohanty, R.K., and Singh, S. "A new algorithm based on spline in tension approximation for 1D parabolic quasi-linear equations on a variable mesh", *Int. J. Comput. Math.*, **93**, pp. 1771–1786 (2016).
- Jiwari, R. "A hybrid numerical scheme for the numerical solution of the Burgers' equation", *Comput. Phys. Commun.*, **188**, pp. 50–67 (2015).
- Seydaoglu, M., Erdogan, U., and Ozis, T. "Numerical solution of Burgers' equation with higher order splitting methods", *J. Comput. Appl. Math.*, **291**, pp. 410–421 (2016).
- Korkmaz, A. and Dag, I. "Polynomial based differential quadrature method for numerical solution of nonlinear Burgers' equation", *J. Franklin I.*, **248**, pp. 2863–2875 (2016).
- Bahadir, A.R. and Saglam, M. "A mixed finite difference and boundary element approach to one-dimensional Burgers' equation", *Appl. Math. Comput.*, **160**, pp. 663–673 (2005).
- Egidi, N., Maponi, P., and Quadrini, M. "An integral equation method for the numerical solution of the Burgers equation", *Comput. Math. Appl.*, **76**(1), pp. 35–44 (2018).
- Inan, B. and Bahadir, A.R. "An explicit exponential finite difference method for the Burgers' equation", *European International Journal of Science and Technology*, **2**(10), pp. 61–72 (2013).
- Zeytinoglu, A., Sari, M., and Pasaoglu, B.A. "Numerical simulations of shock wave propagating by a hybrid approximation based on high-order finite difference schemes", *Acta Phys. Pol. A*, **133**, pp. 369–386 (2017).
- Verma, A.K. and Verma, L. "Higher order time integration formula with application on Burgers' equation", *Int. J. Comput. Math.*, **92**, pp. 756–771 (2015).

19. Rouzegar, J. and Sharifpoor, R.A. “A finite element formulation for bending analysis of isotropic and orthotropic plates based on two-variable refined plate theory”, *Sci. Iran.*, **22**(1), pp. 196–207 (2015).
20. Ak, T., Karakoc, S.B.G., and Biswas, A. “Application of Petrov-Galerkin finite element method to shallow water waves model: modified Korteweg-de Vries equation”, *Sci. Iran.*, **24**(3), pp. 1148–1159 (2017).
21. Zendehebudi, G. “Combination of projection and Galerkin finite element methods to solve the problem of free convection in enclosures with complex geometries”, *Sci. Iran.*, **25**(3), pp. 1189–1196 (2018).
22. Kutluay, S., Esen, A., and Dag, I. “Numerical solutions of the Burgers’ equation by the least-squares quadratic B-spline finite element method”, *J. Comput. Appl. Math.*, **167**, pp. 21–33 (2004).
23. Sari, M. and Tunc, H. “An optimization technique in analyzing the Burgers equation”, *Sigma Journal of Engineering and Natural Sciences*, **35**(3), pp. 369–386 (2017).
24. Kutluay, S. and Esen, A. “A lumped Galerkin method for solving the Burgers equation”, *Int. J. Comput. Math.*, **81**(11), pp. 1433–1444 (2004).
25. Dag, I., Saka, B., and Boz, A. “B-spline Galerkin methods for numerical solutions of Burgers’ equation”, *Appl. Math. Comput.*, **166**, pp. 506–522 (2005).
26. Aksan, E.N. “An application of cubic B-spline finite element method for the Burgers’ equation”, *Therm. Sci.*, **22**(1), pp. 195–202 (2018).
27. Ak, T. “An application of Galerkin method to generalized Benjamin-Bona-Mahony-Burgers equation”, *Adiyaman University Journal of Science*, **8**(2), pp. 53–69 (2018).
28. Ak, T., Saha, A., and Dhawan, S. “Performance of a hybrid computational scheme on travelling waves and its dynamic transition for Gilson Pickering equation”, *Int. J. Mod. Phys. C*, **30**(4), pp. 1950028–1/17 (2019).
29. Mohammadi, R. “Numerical approximation for viscous Cahn-Hilliard equation via septic B-spline”, *Appl. Anal.* (2019). DOI: 10.1080/00036811.2019.1594200
30. Mukundan, V. and Awasthi, A. “Efficient numerical techniques for Burgers’ equation”, *Appl. Math. Comput.*, **262**, pp. 282–297 (2015).
31. Aksan, E.N. “Quadratic B-spline finite element method for numerical solution of the Burgers equation”, *Appl. Math. Comput.*, **174**, pp. 884–896 (2006).
32. Tunc, H. “Various finite element techniques for advection-diffusion-reaction processes”, MSc Thesis, Yildiz Technical University (2017).
33. Shao, L., Feng, X., and He, Y. “The local discontinuous Galerkin finite element method for Burgers equation”, *Math. Comput. Model.*, **54**, pp. 2943–2954 (2011).
34. Abazari, R. and Borhanifar, A. “Numerical study of the solution of the Burgers and coupled Burgers equations by a differential transformation method”, *Comput. Math. Appl.*, **59**, pp. 2711–2722 (2010).

Biographies

Huseyin Tunc is currently a PhD student at the Mathematics Department of Yildiz Technical University. He received his BSc degree from Ege University in Mathematics in 2015 and his MSc degree in Mathematical Engineering from Yildiz Technical University, Turkey in 2017. His research interests include numerical solutions to differential equations, fluid dynamics, dynamical systems, finite element method, and high-order time integration methods. He has 6 international scientific publications.

Murat Sari is a Professor of Mathematics at Yildiz Technical University, Istanbul, Turkey. He received a BSc degree from the Ondokuz Mayıs University in 1991, and MPhil (transferred) and PhD degrees in Mathematics from the University of South Wales, UK, in 1997 and 2000, respectively. His current research interests include numerical solutions of differential equations, computational methods, computational fluid dynamics, modelling of nonlinear behavior, economical modelling, and biomechanical/biomedical modelling. He has published about 60 high-quality scientific papers, over 40 conference proceedings, and written/translated a number of chapters in various books. He has supervised 15 graduate students and has been supervising 8 graduate students. He has carried out many talks in various universities in different countries such as the Petroleum-Gas University of Ploiești, Romania, Pardubice University, Czech Republic, Université Toulouse III-Paul Sabatier, France, Ningbo University in China, Pamukkale University, Bogazici University, Yildiz Technical University, Erzinçan University, and Dumlupınar University in Turkey. He is a reviewer for many international high-quality journals.