UDC 517.5

O. Ersoy, I. Dag, N. Adar (Eskisehir Osmangazi Univ., Turkey)

THE EXPONENTIAL TWICE CONTINUOUSLY DIFFERENTIABLE B-SPLINE ALGORITHM FOR BURGERS' EQUATION

ЕКСПОНЕНЦІАЛЬНИЙ ДВІЧІ НЕПЕРЕРВНО ДИФЕРЕНЦІЙОВНИЙ B-СПЛАЙНОВИЙ АЛГОРИТМ ДЛЯ РІВНЯННЯ БЮРГЕРСА

The exponential twice continuously differentiable B-spline functions known from the literature as the exponential are used to set up the collocation method for finding solutions of the Burgers' equation. The effect of the exponential cubic B-splines in the collocation method is sought by studying the text problems.

Експоненціальні двічі неперервно диференційовні B-сплайнові функції, що відомі з літератури як експоненціальні, застосовано для побудови методу колокацій для знаходження розв'язків рівняння Бюргерса. Ефект експоненціальних кубічних B-сплайнів у методі колокацій знайдено за допомогою аналізу текстових задач.

1. Introduction. This paper is concerned with adapting the exponential cubic B-spline function into the collocation method to develop a numerical method for finding numerical solutions of the Burgers' equation of the form

$$U_t + UU_x - \lambda U_{xx} = 0, \qquad a \le x \le b, \qquad t \ge 0, \tag{1}$$

with the initial condition and the boundary conditions

$$U(x,0) = f(x), \qquad a \le x \le b, \tag{2}$$

$$U(a,t) = \sigma_1, \qquad U(b,t) = \sigma_2 \tag{3}$$

where subscripts x and t denote differentiation, $\lambda \doteq \dfrac{1}{\mathrm{Re}} > 0$ and Re is the Reynolds number characterizing the strength of viscosity, σ_1, σ_2 are the constants $u = u(x,t)$ is a sufficiently differentiable unknown function and $f(x)$ is a bounded function. Initial condition and boundary conditions will be defined in the later section depending on the test problems.

Burgers' equation was first introduced by [2]. Solutions of the Burgers' equation were presented by using some numerical methods with splines. A cubic spline collocation procedure has been developed for the numerical solution of the Burgers' equation in the papers [3, 4]. A B-spline Galerkin method is described to solve the Burgers' equation over the both fixed and varied distribution of knots to define the B-splines in the studies of Davies [5, 6]. A numerical method [7 – 9] is developed for solving the Burgers' equation by using splitting method and cubic spline approximation method. In [14, 17, 18, 23], numerical solutions of the one-dimensional Burgers' equation are obtained by a methods based on collocation of quadratic, cubic and quintic B-splines over finite elements, in which approximate functions in the collocation method for the Burgers' equation are constructed by using the various degree B-splines. Galerkin methods based on various degree B-splines have been set up to find approximate solutions of the Burgers' equation in the studies [6, 11, 13]. The least square method is combined with the B-splines to form numerical methods for solving the Burgers' equation in the works [10, 22]. Numerical solutions of the Burgers' equation is presented based on the cubic B-spline quasiinterpolation and the compact finite difference method in [20]. Taylor-collocation and Taylor – Galerkin methods for the numerical solutions of the Burgers' equation are formed by using both cubic and quadratic B-splines in the study [19] respectively. Differential quadrature methods based on cubic and quartic B-splines are set up to solve the Burgers' equation in the works [21, 25, 26]. The hybrid spline difference method is developed to solve the Burgers' equation by Chi-Chang Wang et al. [24].

The exponential cubic B-spline function and its some properties are described in detail in the paper [27]. Since each exponential basis we use is twice continuously differentiable, we can form twice continuously approximate solution to the differential equations. There exist few articles which are used to form numerical methods to solve differential equations. The exponential cubic B-splines are used with the collocation method to find the numerical solution of the singular perturbation problem by Manabu Sakai et al. [28]. Another application of the collocation method using the cardinal exponential cubic B-splines was shown for finding the numerical solutions of the singularly perturbed boundary problem in the study of D. Radunuvic [29]. The exponential cubic B-spline collocation method is set up to obtain the numerical solutions of the self-adjoint singularly perturbed boundary-value problems in the work [30]. The only linear partial differential equation known as the convection-diffusion equation is solved by way of the exponential cubic B-spline collocation method in the study [31]. Also exponential cubic B-spline collocation method have been applied to obtain numerical solution of Equal width equation, Korteweg – de Vries equation, Fisher equation and Kuramoto – Sivashinsky equation [32 – 35] recently.

In this paper, we have compared results of the Burgers' equation with those obtained with both the cubic B-spline collocation method and cubic B-spline Galerkin finite element method [12, 13] since the B-spline and exponential cubic B-spline functions have almost the same properties. In Section 2, exponential cubic B-spline collocation method is described. In Section 3, three classical test examples are studied to show the versatility of proposed algorithm and finally the conclusion is included to discuss the outcomes of the algorithm.

2. Collocation method via exponential cubic B-spline. The problem domain $[a, b]$ is partitioned equally at the knots

$$\pi : a = x_0 < x_1 < \ldots < x_N = b$$

with distance $h = (b - a)/N$ between consecutive knots. The exponential cubic B-splines, $B_i\ (x)$, at the points of π, can be defined as

$$B_i(x) = \begin{cases} b_2\left((x_{i-2} - x) - \dfrac{1}{p}\left(\sinh(p\left(x_{i-2} - x\right))\right)\right), & [x_{i-2}, x_{i-1}]\,, \\ a_1 + b_1\left(x_i - x\right) + c_1 \exp\left(p\left(x_i - x\right)\right) + d_1 \exp\left(-p\left(x_i - x\right)\right), & [x_{i-1}, x_i]\,, \\ a_1 + b_1(x - x_i) + c_1 \exp\left(p\left(x - x_i\right)\right) + d_1 \exp\left(-p\left(x - x_i\right)\right), & [x_i, x_{i+1}]\,, \\ b_2\left((x - x_{i+2}) - \dfrac{1}{p}(\sinh\left(p\left(x - x_{i+2}\right)\right))\right), & [x_{i+1}, x_{i+2}]\,, \\ 0, & \text{otherwise}, \end{cases} \tag{4}$$

where

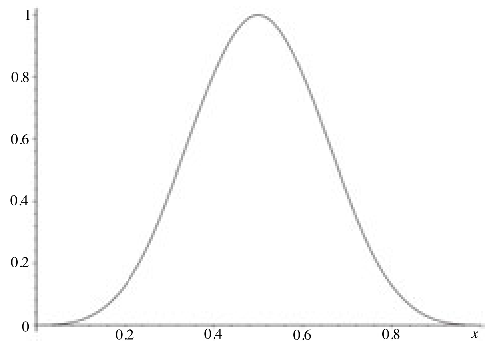

Fig. 1. Exponential cubic B-splines over the interval $[0,1]$.

$$a_1 = \frac{phc}{phc - s}, \qquad b_1 = \frac{p}{2}\left[\frac{c(c-1)+s^2}{(phc-s)(1-c)}\right], \qquad b_2 = \frac{p}{2(phc-s)},$$

$$c_1 = \frac{1}{4}\left[\frac{\exp(-ph)(1-c)+s(\exp(-ph)-1)}{(phc-s)(1-c)}\right],$$

$$d_1 = \frac{1}{4}\left[\frac{\exp(ph)(c-1)+s(\exp(ph)-1)}{(phc-s)(1-c)}\right]$$

and $c = \cosh(ph)$, $s = \sinh(ph)$, p is a free parameter. On the particular interval $[0,1]$, the exponential cubic B-spline function is depicted for $p = 1$ in Fig. 1.

$\{B_{-1}(x), B_0(x), \ldots, B_{N+1}(x)\}$ composes a basis, so that any function defined on the interval $[a,b]$ can be expressed as the linear combination of the element of the basis. Each basis function $B_i(x)$ has got the second derivatives. The values of $B_i(x)$, $B_i^{'}(x)$ and $B_i^{''}(x)$ at the knots x_i's can be computed from Eq. (4) shown Table 1.

Table 1. Values of $B_i(x)$ and its principle two derivatives at the knot points

x	x_{i-2}	x_{i-1}	x_i	x_{i+1}	x_{i+2}
B_i	0	$\dfrac{s-ph}{2(phc-s)}$	1	$\dfrac{s-ph}{2(phc-s)}$	0
$B_i^{'}$	0	$\dfrac{p(1-c)}{2(phc-s)}$	0	$\dfrac{p(c-1)}{2(phc-s)}$	0
$B_i^{''}$	0	$\dfrac{p^2 s}{2(phc-s)}$	$-\dfrac{p^2 s}{phc-s}$	$\dfrac{p^2 s}{2(phc-s)}$	0

Let U_N be approximate solution for U

$$U_N(x,t) = \sum_{i=-1}^{N+1} \delta_i B_i(x), \tag{5}$$

where δ_i are time dependent parameters. Nodal values $U(x,t)$ and its first and second derivatives at the knots can be computed from Eq. (5) with respect to the parameters as

$$U_i = U(x_i,t) = \frac{s-ph}{2(phc-s)}\delta_{i-1} + \delta_i + \frac{s-ph}{2(phc-s)}\delta_{i+1},$$

$$U_i' = U'(x_i,t) = \frac{p(1-c)}{2(phc-s)}\delta_{i-1} + \frac{p(c-1)}{2(phc-s)}\delta_{i+1}, \tag{6}$$

$$U_i'' = U''(x_i,t) = \frac{p^2 s}{2(phc-s)}\delta_{i-1} - \frac{p^2 s}{phc-s}\delta_i + \frac{p^2 s}{2(phc-s)}\delta_{i+1}.$$

Time discretization of unknown U is managed by way of the Grank – Nicolson scheme in the Burgers' equation to obtain following equation:

$$\frac{U^{n+1}-U^n}{\Delta t} + \frac{(UU_x)^{n+1} + (UU_x)^n}{2} - \lambda\frac{U_{xx}^{n+1} + U_{xx}^n}{2} = 0, \tag{7}$$

where $U^{n+1} = U(x,t)$ is the solution of the equation at the $(n+1)$th time level. Here $t^{n+1} = t^n + \Delta t$, and Δt is the time step, superscripts denote nth time level, $t^n = n\Delta t$.

The nonlinear term $(UU_x)^{n+1}$ in Eq. (7) is linearized by using the following form given by Rubin and Graves [3]:

$$(UUx)^{n+1} = U^{n+1}U_x^n + U^nU_x^{n+1} - U^nU_x^n, \tag{8}$$

is applied to Eq. (7) to obtain the time discretized Burgers' equation

$$U^{n+1} - U^n + \frac{\Delta t}{2}(U^{n+1}U_x^n + U^nU_x^{n+1}) - \lambda\frac{\Delta t}{2}(U_{xx}^{n+1} - U_{xx}^n) = 0. \tag{9}$$

Place Eq. (5) in (9) to have fully discretized system of equations

$$\left(\alpha_1 + \frac{\Delta t}{2}\left(\alpha_1 L_2 + \beta_1 L_1 - \lambda\gamma_1\right)\right)\delta_{m-1}^{n+1} + \left(\alpha_2 + \frac{\Delta t}{2}\left(\alpha_2 L_2 - \lambda\gamma_2\right)\right)\delta_m^{n+1} +$$
$$+ \left(\alpha_3 + \frac{\Delta t}{2}\left(\alpha_3 L_2 + \beta_2 L_1 - \lambda\gamma_3\right)\right)\delta_{m+1}^{n+1} = \left(\alpha_1 + \lambda\frac{\Delta t}{2}\gamma_1\right)\delta_{m-1}^n +$$
$$+ \left(\alpha_2 + \lambda\frac{\Delta t}{2}\gamma_2\right)\delta_m^n + \left(\alpha_3 + \lambda\frac{\Delta t}{2}\gamma_3\right)\delta_{m+1}^n, \tag{10}$$

where

$$L_1 = \alpha_1\delta_{i-1} + \alpha_2\delta_i + \alpha_3\delta_{i+1},$$

$$L_2 = \beta_1\delta_{i-1} + \beta_2\delta_{i+1},$$

$$\alpha_1 = \frac{s-ph}{2(phc-s)}, \quad \alpha_2 = 1, \quad \alpha_3 = \frac{s-ph}{2(phc-s)},$$

$$\beta_1 = \frac{p(1-c)}{2(phc-s)}, \quad \beta_2 = \frac{p(c-1)}{2(phc-s)},$$

$$\gamma_1 = \frac{p^2 s}{2(phc-s)}, \quad \gamma_2 = -\frac{p^2 s}{phc-s}, \quad \gamma_3 = \frac{p^2 s}{2(phc-s)}.$$

The system consist of $N+1$ linear equation in $N+3$ unknown parameters $\mathbf{d}^{n+1} = (\delta_{-1}^{n+1}, \delta_0^{n+1}, \ldots, \delta_{N+1}^{n+1})$. The boundary conditions $\sigma_1 = U_0,\ \sigma_2 = U_N$ are gives two additional linear equations

$$\delta_{-1} = \frac{1}{\alpha_1}\left(U_0 - \alpha_2\delta_0 - \alpha_3\delta_1\right),$$
$$\delta_{N+1} = \frac{1}{\alpha_3}\left(U_N - \alpha_1\delta_{N-1} - \alpha_2\delta_N\right). \tag{11}$$

Eqs. (11) can be used to eliminate $\delta_{-1},\ \delta_{N+1}$ from the system (10) which then becomes the solvable matrix equation for the unknown $\delta_0^{n+1}, \ldots, \delta_N^{n+1}$. A variant of Thomas algorithm is used to solve the system.

Use of the initial condition and first space derivative of the initial conditions at the boundaries allows to have a system:

$$U_N(x_i, 0) = U(x_i, 0), \quad i = 0, \ldots, N,$$
$$(U_x)_N(x_0, 0) = U'(x_0),$$
$$(U_x)_N(x_N, 0) = U'(x_N).$$

3. Computational examples. Solution of the system produces initial parameters $\delta_{-1}^0, \delta_0^0, \ldots$ $\ldots, \delta_{N+1}^0$, so that we can start solving the recursive system at times requested. Numerical method described in the previous section will be tested on three text problems for getting solutions of the Burgers' equation. Three kinds of examples are presented in order to demonstrate the versatility and the accuracy of the proposed method. The discrete L_2 and L_∞ error norm

$$L_2 = \sqrt{h\sum_{j=0}^{N}\left|(U_j^n - (U_N)_j^n)^2\right|},$$

$$L_\infty = \max_j \left|U_j^n - (U_N)_j^n\right|$$

are used to measure error between the analytical and numerical solutions.

(a) The Burger's equation, with the sine wave initial condition $U(x,0) = \sin(\pi x)$ and boundary conditions $U(0,t) = U(1,t) = 0$, has analytic solution in the form of the infinite series defined by [15] as

$$U(x,t) = \frac{4\pi\lambda \sum_{j=1}^{\infty} jI_j\left(\frac{1}{2\pi\lambda}\right)\sin(j\pi x)\exp\left(-j^2\pi^2\lambda t\right)}{I_0\left(\frac{1}{2\pi\lambda}\right) + 2\sum_{j=1}^{\infty} I_j\left(\frac{1}{2\pi\lambda}\right)\cos(j\pi x)\exp\left(-j^2\pi^2\lambda t\right)}, \tag{12}$$

where $\mathbf{I}_j$ are the modified Bessel functions. This problem gives the decay of sinusoidal disturbance. Numerical solutions at different times are depicted in Figs. 2–5 for the parameters $N = 40$ and $N = 80$, $\Delta t = 0.0001, \lambda = 1, 0.1, 0.01, 0.001$. From the figures we see that the smaller viscosities λ cause to develop the sharp front thorough the right boundary and amplitude of the sharp front starts to decay as time progress. These properties of solutions are in very good agreement with findings of Saka and Dağ [16, 17].

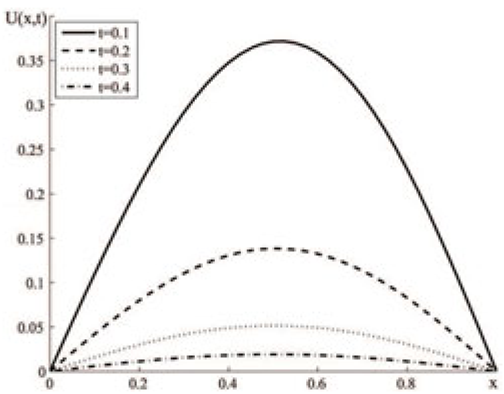

Fig. 2. Solutions for $\lambda = 1$, $N = 40$, $\Delta t = 0.0001$.

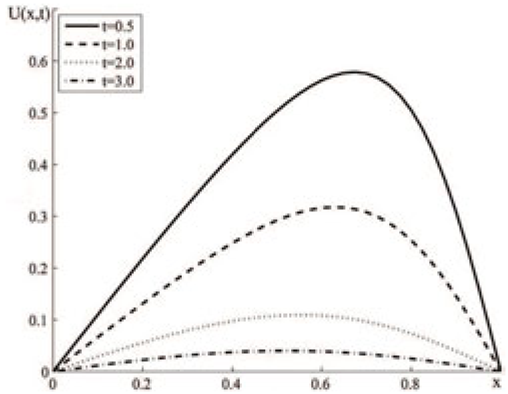

Fig. 3. Solutions for $\lambda = 0.1$, $N = 40$, $\Delta t = 0.0001$.

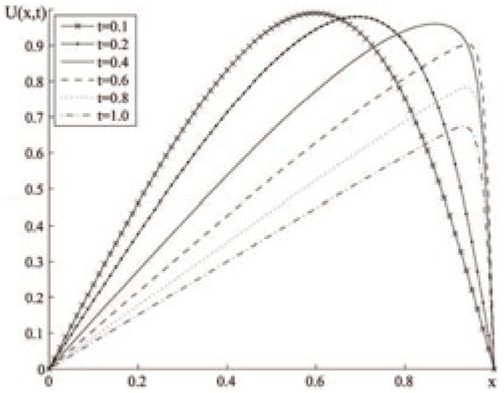

Fig. 4. Solutions for $\lambda = 0.01$, $N = 80$, $\Delta t = 0.0001$.

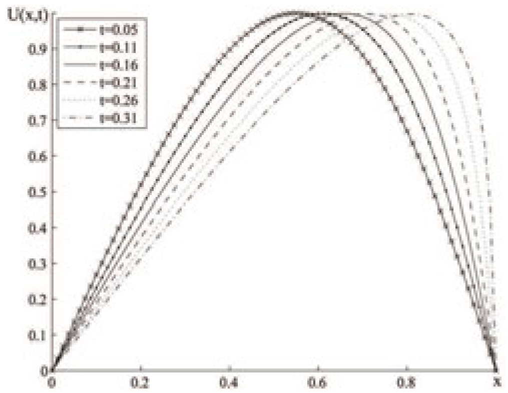

Fig. 5. Solutions for $\lambda = 0.001$, $N = 80$, $\Delta t = 0.0001$.

Two dimensional solutions are depicted from time $t = 0$ to $t = 1$ with time increment $\Delta t = 0.0001$ for space increment $h = 0.25$ and various λ in Figs. 6–9. When the smaller $\lambda = 0.001$ is taken, the solutions starts to decay after about time $t = 0.6$ when $N = 40$ is used. So to have acceptable solution with $\lambda = 0.001$, we decrease the space step to $h = 0.125$ and graph of the solution is shown in Figs. 8, 9.

A comparison has been made between the present collocation method and alternative approaches including the cubic B-spline collocation method and cubic B-spline Galerkin method for parameters of $\Delta t = 0.0001$, $N = 80$, $\lambda = 0.01$. Exact solutions for $\lambda > 10^{-2}$ are not practical because of the low convergence of the infinite series so that these results are not compared with the exact solutions. It can be seen from Tables 2 and 3 that accuracy of the presented solutions is much the same with both the cubic B-spline collocation method and cubic B-spline Galerkin method. When the size of the space variable is reduced, the error becomes less than that of the cubic B-spline collocation methods and is almost close to the cubic B-spline Galerkin method and solution values are documented in Table 3 at time $t = 0.1$.

(b) As the second example, we consider particular solution of Burgers' equation with initial condition

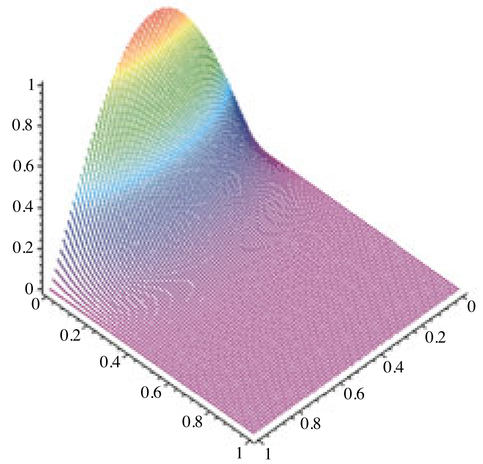

Fig. 6. Solutions for $\lambda = 1,\ N = 40,\ \Delta t = 0.01$.

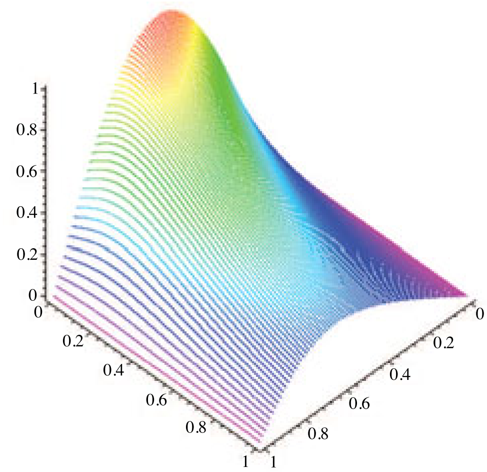

Fig. 7. Solutions for $\lambda = 0.1,\ N = 40,\ \Delta t = 0.01$.

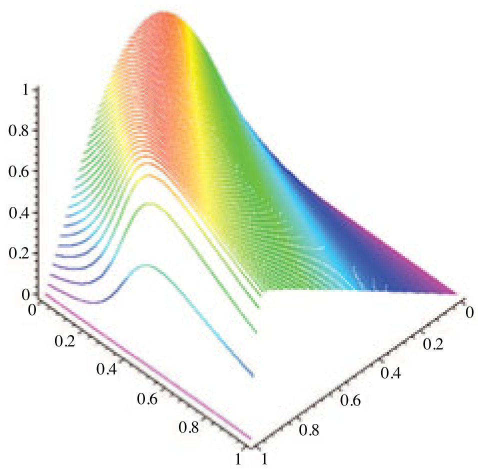

Fig. 8. Solutions for $\lambda = 0.01,\ N = 80,\ \Delta t = 0.01$.

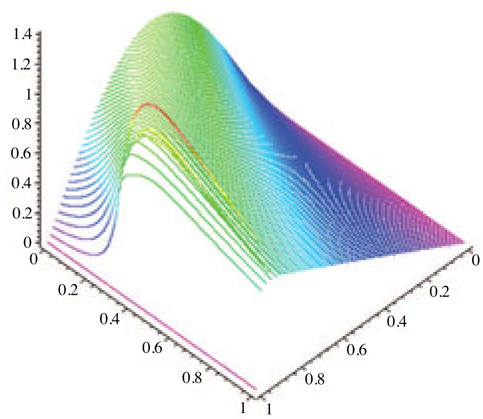

Fig. 9. Solutions for $\lambda = 0.001,\ N = 80,\ \Delta t = 0.001$.

$$U(x,1) = \exp\left(\frac{1}{8\lambda}\right), \quad 0 \le x \le 1,$$

and boundary conditions $U(0,t) = 0$ and $U(1,t) = 0$.

This problem has the following analytical solution:

$$U(x,t) = \frac{\dfrac{x}{t}}{1 + \sqrt{\dfrac{t}{t_0}}\exp\left(\dfrac{x^2}{4\lambda t}\right)}, \quad t \ge 1, \quad 0 \le x \le 1. \tag{13}$$

This solution represents the propagation of the shock and the selection of the smaller λ result in steep shock solution. So the success of the numerical method depends on dealing with the steep shock efficiently.

Table 2. Numerical results for $p = 1,\ \lambda = 0.01,$
$N = 40,\ \Delta t = 0.0001$ at different times

x	Time	Present	Ref. [12]	Ref. [13]	Exact
0.25	0.4	0.34192	0.34192	0.34192	0.34191
	0.6	0.26897	0.26897	0.26897	0.22896
	0.8	0.22148	0.22148	0.22148	0.22148
	1.0	0.18819	0.18819	0.18819	0.18819
	3.0	0.07511	0.07511	0.07511	0.07511
0.50	0.4	0.66071	0.66071	0.66071	0.66071
	0.6	0.52942	0.52942	0.52942	0.52942
	0.8	0.43914	0.43914	0.43914	0.43914
	1.0	0.37442	0.37442	0.37442	0.37442
	3.0	0.15018	0.15018	0.15018	0.15018
0.75	0.4	0.91027	0.91027	0.91027	0.91026
	0.6	0.76725	0.76725	0.76724	0.76724
	0.8	0.64740	0.64740	0.64740	0.64740
	1.0	0.55605	0.55605	0.55605	0.55605
	3.0	0.22483	0.22483	0.22481	0.22481

Table 3. Numerical results for $p = 1,\ t = 0.1,\ \lambda = 1,\ \Delta t = 0.0001$ at different sizes

x	h	Present	Ref. [12]	Ref. [13]	Exact
0.1	0.0125	0.10953	0.10952	0.10954	0.10954
0.2		0.20977	0.20975	0.20979	0.20979
0.3		0.29186	0.29184	0.29189	0.29190
0.4		0.34788	0.34788	0.34792	0.34792
0.5		0.37153	0.37153	0.37158	0.37158
0.6		0.35899	0.35896	0.35904	0.35905
0.7		0.30986	0.30983	0.30990	0.30991
0.8		0.22778	0.22776	0.22782	0.22782
0.9		0.12067	0.12065	0.12069	0.12069
0.1	h = 0.0625	0.10954	0.10953	0.10954	0.10954
0.2		0.20979	0.20977	0.20979	0.20979
0.3		0.29189	0.29186	0.29190	0.29190
0.4		0.34792	0.34788	0.34792	0.34792
0.5		0.37156	0.37153	0.37158	0.37158
0.6		0.35903	0.35900	0.35904	0.35905
0.7		0.30989	0.30986	0.30990	0.30991
0.8		0.22781	0.22778	0.22782	0.22782
0.9		0.12068	0.12067	0.12069	0.12069

Table 4. Numerical results for $p = 1, \lambda = 0.0005,\ h = 0.005,\ \Delta t = 0.01$ at different times

x	Times	Present	Ref. [12]	Exact
0.1	1.7	0.05882	0.05883	0.05882
0.2		0.11765	0.11765	0.11765
0.3		0.17647	0.17648	0.17647
0.4		0.23529	0.23531	0.23529
0.5		0.29412	0.29414	0.29412
0.6		0.35294	0.35296	0.35294
0.7		0.00000	0.00000	0.00000
0.8		0.00000	0.00000	0.00000
0.9		0.00000	0.00000	0.00000
0.1	2.5	0.04000	0.04000	0.04000
0.2		0.08000	0.08000	0.08000
0.3		0.12000	0.12001	0.12000
0.4		0.16000	0.16001	0.16000
0.5		0.20000	0.20001	0.20000
0.6		0.24000	0.24001	0.24000
0.7		0.28000	0.28001	0.28000
0.8		0.00828	0.00811	0.00977
0.9		0.00000	0.00000	0.00000
0.1	3.25	0.03077	0.03077	0.03077
0.2		0.06154	0.06154	0.06154
0.3		0.09231	0.09231	0.09231
0.4		0.12308	0.12308	0.12308
0.5		0.15385	0.15385	0.15385
0.6		0.18462	0.18462	0.18462
0.7		0.21538	0.21539	0.21538
0.8		0.24615	0.24616	0.24615
0.9		0.12394	0.12358	0.12435

The propogation of the shock is studied with parameters $\lambda = 0.005, 0.0005$. Numerical solutions obtained by exponential collocation method can be favorably compared with results reported in the papers [12, 13] at some times in the same Table 4. Figs. 10 and 11 show propagation of shock $\lambda = 0.005,\ h = 0.02,\ \Delta t = 0.1$ and $\lambda = 0.0005,\ h = 0.005,\ \Delta t = 0.01$ respectively. As time advances, the initial steep shock becomes smoother when the the larger viscosity is used but for the small viscosity it is steeper. These observations are in complete agreement with those reported in the papers [9].

(c) Travelling wave solution of the Burgers' equation has the form:

$$U(x,t) = \frac{\alpha + \mu + (\mu - \alpha)\exp\eta}{1 + \exp\eta}, \qquad 0 \le x \le 1, \quad t \ge 0, \tag{14}$$

where

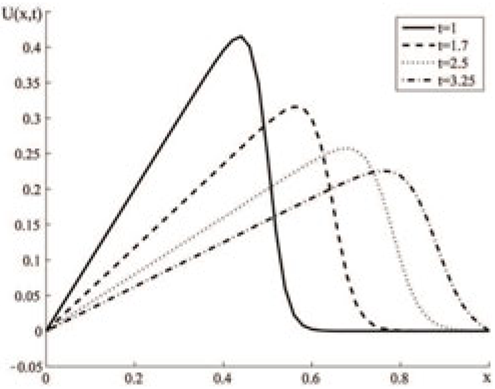

Fig. 10. Shock propagation, $\lambda = 0.005$.

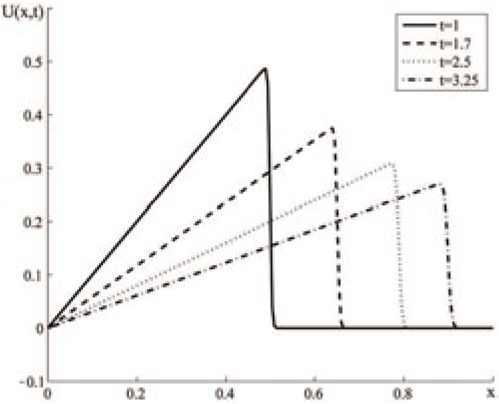

Fig. 11. Shock propagation, $\lambda = 0.0005$.

$$\eta = \frac{\alpha(x - \mu t - \gamma)}{\lambda},$$

and $\alpha,\ \mu$ and γ are arbitrary constants. The boundary conditions is

$$U(0,t) = 1, \qquad U(1,t) = 0.2$$

or

$$U_x(0,t) = 0, \qquad U_x(1,t) = 0 \quad \text{for} \quad t \geq 0$$

and initial condition is obtained from the analytical solution (14) when $t = 0$. Analytical solution takes values between 1 and 0.2 and the propagation of the wave front through the right will be observed with varying λ. The smaller λ we take for the Burgers' equation, the steeper the wave front propagates. The robustness of the algorithm will be shown by monitoring the motion of the wave front with smaller λ. The algorithm has run for the values $\alpha = 0.4,\ \mu = 0.6,\ \gamma = 0.125$ and $\lambda = 0.01,\ h = 1/36,\ \Delta t = 0.001,\ p = 1$. Visual motion of the wave front is depicted in Figs. 12 and 13 for the $\lambda = 0.01, 0.05$. The numerical results demonstrate the formation of the steep front and very steeper front. Error graphs of the numerical solutions are also shown in Figs. 14 and 15. From figures the maximum error occurs in the midle of the solution domain. Solutions from time $t = 0$ to $t = 1.2$ at some times are visualised in 3D graph to see the propogation of the sharp behaviours in Figs. 16 and 17 for $h = 1/80$ and $\Delta t = 0.0001$.

4. Conclusion. The exponential cubic B-spline collocation method for the numerical solutions of the Burges' equation is presented over the finite elements so that the continuity of the dependent variable and its first two derivatives is satisfied for the approximate solution throughout the solution range. The equation has been integrated into a system of the linearized iterative algebraic equations. The system of the iterative at each time step in which it has got a three-banded coefficients matrix is solved with the Thomas algorithm. Generally, comparative results show that results of our finding is better than the cubic B-spline collocation method and is much the same with that of the cubic B-spline Galerkin method. Since cost of the cubic B-spline Galerkin method is higher than the suggested method, that is advantages of the exponential cubic B-spline collocation method over the cubic B-spline Galerkin method. During all runs of the algorithm, the best result are found for the free parameter $p = 1$ for the exponential cubic B-spline functions.

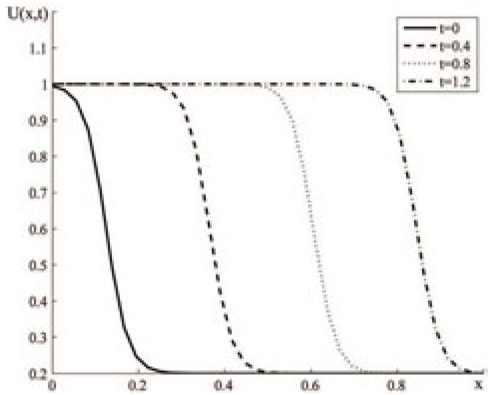

Fig. 12. Solutions for $\lambda = 0.01$.

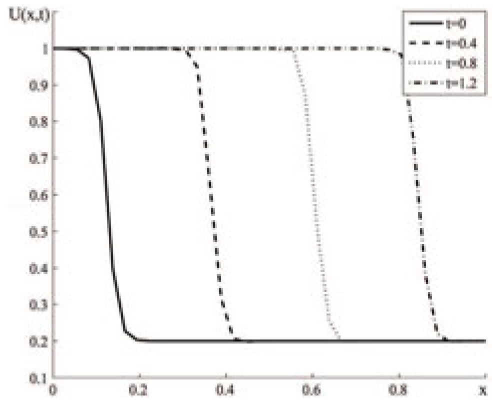

Fig. 13. Solutions for $\lambda = 0.005$.

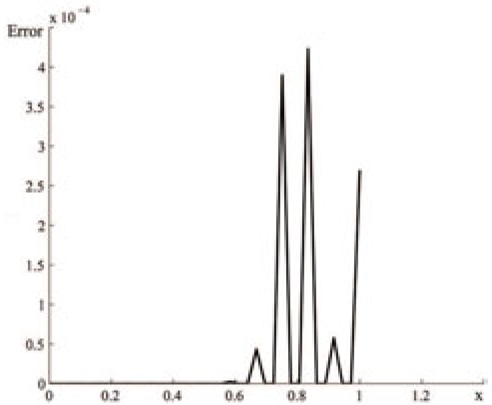

Fig. 14. L_2 error norm for $\lambda = 0.01,\ t = 1.2$.

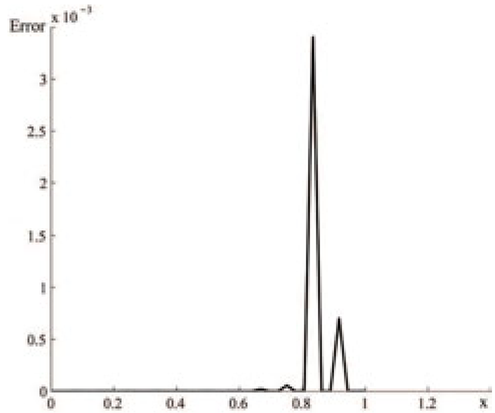

Fig. 15. L_2 error norm for $\lambda = 0.005,\ t = 1.2$.

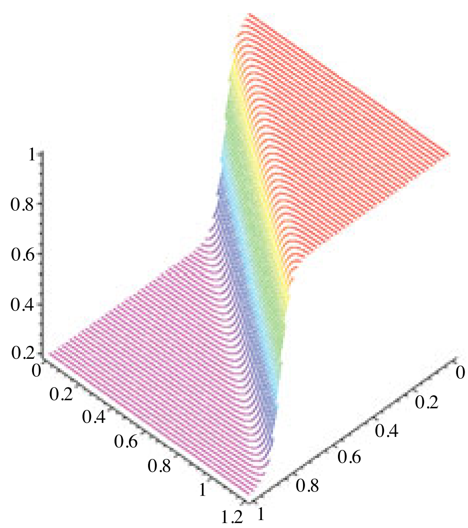

Fig. 16. Shock propagation, $\lambda = 0.01$.

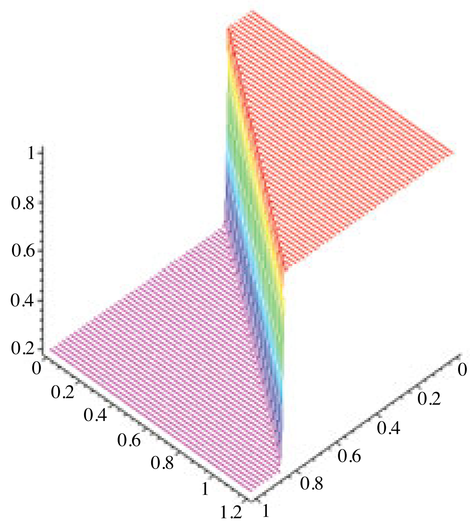

Fig. 17. Shock propagation, $\lambda = 0.005$.

References

1. *Hopf E.* The partial differential equation $U_t + UU_x - \nu U_{xx} = 0$ // Communs Pure and Appl. Math. – 1950. – **3**. – P. 201 – 230.
2. *Bateman H.* Some recent researches on the motion of fluids // Monthly Weather Rev. – 1915. – **43**. – P. 163 – 170.
3. *Rubin S. G., Graves R. A.* Cubic spline approximation for problems in fluid mechanics / Nasa TR R-436,Washington, DC. – 1975.
4. *Rubin S. G., Khosla P. K.* High-order numerical solutions using cubic splines // AIAA J. – 1976. – **14**, № 7. – P. 851 – 858.
5. *Davies A. M.* A Numerical investigation of errors arising in applying the Galerkin method of the solution of nonlinear partial differential equations // Comput. Methods Appl. Mech. – 1977. – **11**. – P. 341 – 350.
6. *Davies A. M.* Application of the Galerkin method to the solution of Burgers equation // Comput. Methods Appl. Mech. – 1978. – **14**. – P. 305 – 321.
7. *Jain P. C., Holla D. H.* Numerical solutions of coupled Burgers' equation // Int. J. Nonlinear Mech. – 1978. – **13**. – P. 213 – 222.
8. *Jain P. C., Lohar B. L.* Cubic spline technique for coupled nonlinear parabolic equations // Comput. Math. Appl. – 1979. – **5**. – P. 179 – 185.
9. *Lohar B. L., Jain P. C.* Variable mesh cubic spline technique for N-wave solution of Burgers' equation // J. Comput. Phys. – 1981. – **39**. – P. 433 – 442.
10. *Kutluay S., Esen A., Dağ I.* Numerical solutions of the Burgers' equation by the least squares quadratic B-spline finite element method // J. Comput. and Appl. Math. – 2004. – **167**. – P. 21 – 33.
11. *Öziş T., Esen A., Kutluay S.* Numerical solution of Burgers equation by quadratic B-spline finite elements // Appl. Math. and Comput. – 2005. – **165**. – P. 237 – 249.
12. *Dağ İ., Irk D., Saka B.* A numerical solution of the Burgers' equation using cubic B-splines // Appl. Math. and Comput. – 2005. – **163**. – P. 199 – 211.
13. *Dağ İ., Saka B., Boz A.* B-spline Galerkin methods for numerical solutions of the Burgers' equation // Appl. Math. and Comput. – 2005. – **166**. – P. 506 – 522.
14. *Aksan E. N.* Quadratic B-spline finite element method for numerical solution of the Burgers equation // Appl. Math. and Comput. – 2006. – **174**. – P. 884 – 896.
15. *Cole D.* On a quasilinear parabolic equation occurring in aerodynamics // Quart. Appl. Math. – 1951. – **9**. – P. 225 – 236.
16. *Saka B., Dağ İ.* Quartic B-spline collocation methods to the numerical solutions of the Burgers' equation // Chaos, Solitons and Fractals. – 2007. – **32**. – P. 1125 – 1137.
17. *Saka B., Dağ İ.* A numerical study of Burgers' equation // J. Frank. Inst. – 2008. – **345**. – P. 328 – 348.
18. *Sepehrian B., Lashani M.* A numerical solution of the Burgers equation using quintic B-splines // Proc. World Congr. Eng. – 2008. – **3**.
19. *Dağ İ., Canıvar A., Şahin A.* Taylor – Galerkin and Taylor-collocation methods for the numerical solutions of Burgers' equation using B-splines // Communs Nonlinear Sci. Numer. Simul. – 2011. – **16**. – P. 2696 – 2708.
20. *Wang R. H., Xi M., Zhang J. H., Fang Q.* A novel numerical scheme for solving Burgers' equation // Appl. Math. and Comput. – 2011. – **217**. – P. 4473 – 4482.
21. *Korkmaz A., Aksoy A. M., Dag I.* Quartic B-spline differential quadrature method // Int. J. Nonlinear Sci. – 2011. – **11**, № 4. – P. 403 – 411.
22. *Chapani H. V., Pradhan V. H., Mehta M. N.* Numerical simulation of Burger's equation using quadratic B-splines // Int. J. Appl. Math. and Mech. – 2012. – **8**, № 11. – P. 18 – 32.
23. *Mittal R. C., Jain R. K.* Numerical solutions of nonlinear Burgers' equation with modified cubic B-splines collocation method // Appl. Math. and Comput. – 2012. – **218**. – P. 7839 – 7855.
24. *Chi-Chang Wang, Wu-Jung Liao, Yin-Sung Hsu.* Hybrid spline difference method for the Burgers' equation // Appl. Math. and Comput. – 2012. – **219**. – P. 1031 – 1039.
25. *Korkmaz A., Dag I.* Cubic B-spline differential quadrature methods and stability for Burgers' equation // Eng. Comput. – 2013. – **30**, № 3. – P. 320 – 344.
26. *Geeta Arora, Brajesh Kumar Singh.* Numerical solution of Burgers' equation with modified cubic B-spline differential quadrature method // Appl. Math. and Comput. – 2013. – **224**. – P. 166 – 177.

27. *McCartin B. J.* Theory of exponential splines // J. Approxim. Theory. – 1991. – **66**. – P. 1 – 23.
28. *Sakai M., Usmani R. A.* A class of simple exponential B-splines and their application to numerical solution to singular perturbation problems // Numer. Math. – 1989. – **55**. – P. 493 – 500.
29. *Radunovic D.* Multiresolution exponential B-splines and singularly perturbed boundary problem // Numer. Algorithms. – 2008. – **47**. – P. 191 – 210.
30. *Chandra Sekhara Rao S., Mukesh Kumar.* Exponential B-spline collocation method for self-adjoint singularly perturbed boundary value problems // Appl. Numer. Math. – 2008. – **58**. – P. 1572 – 1581.
31. *Mohammadi R.* Exponential B-spline solution of convection-diffusion equations // Appl. Math. – 2013. – **4**. – P. 933 – 944.
32. *Ersoy O., Dag I.* The exponential cubic B-Spline algorithm for equal width equation // Adv. Stud. Contemp. Math. – 2015. – **25**, № 34. – P. 525 – 535.
33. *Ersoy O., Dag I.* The exponential cubic B-Spline algorithm for Korteweg – de Vries (KdV) equation // Adv. Numer. Anal. – 2015.
34. *Dag I., Ersoy O.* The exponential cubic B-spline algorithm for Fisher equation // Chaos, Solitons and Fractals. – 2016. – **86**. – P. 101 – 106.
35. *Ersoy O., Dag I.* The exponential cubic B-spline collocation method for the Kuramoto – Sivashinsky(KS) equation // Filomat. – 2016. – **30**, № 3. – P. 853 – 861.

Received 20.10.16