Journal of Physics: Conference Series **766** (2016) 012031 doi:10.1088/1742-6596/766/1/012031

Wave Propagation by Way of Exponential B-Spline Galerkin Method

M Zorsahin Gorgulu[1], I Dag[2] and D Irk[1]

[1] Department of Mathematics-Computer and [2] Department of Computer Engineering, Eskisehir Osmangazi University, 26480, Eskisehir, Turkey.

E-mail: mzorsahin@ogu.edu.tr, idag@ogu.edu.tr, dirk@ogu.edu.tr

Abstract. In this paper, the exponential B-spline Galerkin method is set up for getting the numerical solution of the Burgers' equation. Two numerical examples related to shock wave propagation and travelling wave are studied to illustrate the accuracy and the efficiency of the method. Obtained results are compared with some early studies.

1. Introduction

Burgers' equation in which convection and diffusion play an important role arises in applications such as meteorology, turbulent flows, modelling of the shallow water. Burgers' equation is considered to be useful model for many physical problems. Thus it is often studied for testing of both real life problems and computational techniques. Not only does exact solutions of nonlinear convective problem develops discontinuities in finite time, and might display complex structure near discontinuities. Efficient and accurate methods are in need to be tackled the complex solutions of the Burgers' equation. Though analytical solutions of the Burgers' equation exist for simple initial condition, the numerical techniques are of interest to meet requirement of the wide range of solutions of the Burgers' equation. Some variants of the spline methods have set up to find the numerical solutions of the Burgers' equation such as Galerkin finite element method [3, 8, 12], least square method [2], collocation method [4–7, 11], method based on the cubic B-spline quasi interpolant [9, 10], etc.

Finite element methods are mainly used methods to have good functional approximate solutions of the differential equations. The accuracy of the finite element solutions are increased by the selection of suitable basis function for the approximate function over the finite intervals. The exponential B-splines are suggested to interpolate data and function exhibiting sharp variations [1], since polynomial B-splines based interpolation cause unwanted osculation for interpolation. Some solutions of the Burgers' equation show sharpness. Thus we will construct the finite element method together with the exponential B-splines to have solutions of the Burgers' equation. A few exponential B-spline numerical methods have suggested for some partial differential equations such as [13, 14].

In this study, we will consider the Burgers' equation

$$u_t + uu_x - \nu u_{xx} = 0 \tag{1}$$

where subscripts x and t are space and time parameters, respectively and ν is the viscosity coefficient. Boundary conditions of the Eq. (1) are chosen from

$$u\left(a,t\right)=\beta_1,\ u\left(b,t\right)=\beta_2, \tag{2}$$

and initial condition is

$$u\left(x,0\right)=f\left(x\right),\ x\in\left[a,b\right]. \tag{3}$$

$f\left(x\right)$ and β_1, β_2 constants are described in the computational section.

2. Exponential B-splines Galerkin Finite Element Solution

Divide spatial interval $[a,b]$ in N subintervals of length $h=\frac{b-a}{N}$ and $x_i=x_0+ih$ at the knots x_i, $i=0,..,N$ and time interval $[0,T]$ in M interval of length Δt.

Let $\phi_i\left(x\right)$ be the exponential B-splines defined at the knots x_i, $i=0,\ldots,N$, together with fictitious knots x_i, $i=-3,-2,-1,N+1,N+2,N+3$ outside the interval $[a,b]$. The $\phi_i\left(x\right)$, $i=-1,\ldots,N+1$ can be defined as

$$\phi_i\left(x\right)=\begin{cases} b_2\left[\left(x_{i-2}-x\right)-\frac{1}{p}\left(\sinh\left(p\left(x_{i-2}-x\right)\right)\right)\right] & \text{if } x\in\left[x_{i-2},x_{i-1}\right]; \\ a_1+b_1\left(x_i-x\right)+c_1e^{p(x_i-x)}+d_1e^{-p(x_i-x)} & \text{if } x\in\left[x_{i-1},x_i\right]; \\ a_1+b_1\left(x-x_i\right)+c_1e^{p(x-x_i)}+d_1e^{-p(x-x_i)} & \text{if } x\in\left[x_i,x_{i+1}\right]; \\ b_2\left[\left(x-x_{i+2}\right)-\frac{1}{p}\left(\sinh\left(p\left(x-x_{i+2}\right)\right)\right)\right] & \text{if } x\in\left[x_{i+1},x_{i+2}\right]; \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

where

$$p=\max_{0\le i\le N}p_i,\ s=\sinh\left(ph\right),\ c=\cosh\left(ph\right),\ a_1=\frac{phc}{phc-s},\ b_1=\frac{p}{2}\left[\frac{c\left(c-1\right)+s^2}{\left(phc-s\right)\left(1-c\right)}\right],$$
$$b_2=\frac{p}{2\left(phc-s\right)},\ c_1=\frac{1}{4}\left[\frac{e^{-ph}\left(1-c\right)+s\left(e^{-ph}-1\right)}{\left(phc-s\right)\left(1-c\right)}\right],\ d_1=\frac{1}{4}\left[\frac{e^{ph}\left(c-1\right)+s\left(e^{ph}-1\right)}{\left(phc-s\right)\left(1-c\right)}\right].$$

The $\phi_i\left(x\right)$, $i=-1,\ldots,N+1$ forms a basis for functions defined on the interval $[a,b]$. The Galerkin method consists of seeking approximate solution in the following form:

$$u\left(x,t\right)\approx U\left(x,t\right)=\sum_{i=-1}^{N+1}\phi_i\left(x\right)\delta_i\left(t\right) \tag{5}$$

where $\delta_i\left(t\right)$ are time dependent unknowns to be determined from the boundary conditions and Galerkin approach to the Eq. (1). The approximate solution and the first two derivatives at the knots can be found from the (4-5) as

$$\begin{aligned} U_i&=U(x_i,t)=\alpha_1\delta_{i-1}+\delta_i+\alpha_1\delta_{i+1}, \\ U_i'&=U'(x_i,t)=\alpha_2\delta_{i-1}-\alpha_2\delta_{i+1}, \\ U_i''&=U''(x_i,t)=\alpha_3\delta_{i-1}-2\alpha_3\delta_i+\alpha_3\delta_{i+1} \end{aligned} \tag{6}$$

where $\alpha_1=\frac{s-ph}{2(phc-s)},\alpha_2=\frac{p(1-c)}{2(phc-s)},\alpha_3=\frac{p^2s}{2(phc-s)}$.

Over the sample interval $[x_m,x_{m+1}]$, applying Galerkin approach to Eq. (1) with the weight function $\phi_j\left(x\right)$ yields

$$\int_{x_m}^{x_{m+1}}\phi_j\left(x\right)\left(u_t+uu_x-\nu u_{xx}\right)dx. \tag{7}$$

Journal of Physics: Conference Series **766** (2016) 012031 doi:10.1088/1742-6596/766/1/012031

Substitution of the (6) into the integral equation lead to

$$\sum_{i=m-1}^{m+2}\left(\int_{x_m}^{x_{m+1}}\phi_j\phi_i dx\right)\overset{\bullet}{\delta}_i+\left(\int_{x_m}^{x_{m+1}}\phi_j\left(\sum_{k=m-1}^{m+2}\delta_k\phi_k\right)\phi_i' dx\right)\delta_i-\nu\left(\int_{x_m}^{x_{m+1}}\phi_j\phi_i'' dx\right)\delta_i, \tag{8}$$

where i, j and k take only the values $m-1, m, m+1, m+2$ for $m = 0, 1, \dots, N-1$ and the notation $\bullet$ denotes time derivative.

If we denote A^e_{ji}, $B^e_{jki}(\delta^e)$ and C^e_{ji} by

$$A^e_{ji}=\int_{x_m}^{x_{m+1}}\phi_j\phi_i dx,\quad B^e_{jki}(\delta)=\int_{x_m}^{x_{m+1}}\phi_j\left(\sum_{k=m-1}^{m+2}\delta_k\phi_k\right)\phi_i' dx,\quad C^e_{ji}=\int_{x_m}^{x_{m+1}}\phi_j\phi_i'' dx \tag{9}$$

where $\mathbf{A}^e$ and $\mathbf{C}^e$ are the element matrices of which dimensions are 4×4 and $\mathbf{B}^e(\delta^e)$ is the element matrix with the dimension $4\times 4\times 4$, (8) can be written in the matrix form as

$$\mathbf{A}^e\overset{\bullet}{\delta^e}+\left(\mathbf{B}^e\left(\delta^e\right)-\nu\mathbf{C}^e\right)\delta^e, \tag{10}$$

where $\delta^e = (\delta_{m-1}, ..., \delta_{m+2})^T$.

Gathering the systems (10) over all elements, we obtain global system

$$\mathbf{A}\overset{\bullet}{\delta}+\left(\mathbf{B}(\delta)-\nu\mathbf{C}\right)\delta=0 \tag{11}$$

where $\mathbf{A}, \mathbf{B}(\delta), \mathbf{C}$ are derived from the corresponding element matrices $\mathbf{A}^e, \mathbf{B}^e(\delta^e), \mathbf{C}^e$, respectively and $\delta = (\delta_{-1}, ..., \delta_{N+1})^T$ contains all elements parameters.

The unknown parameters δ are interpolated between two time levels n and $n+1$ with the Crank-Nicolson method

$$\delta=\frac{\delta^{n+1}+\delta^n}{2},\quad \overset{\bullet}{\delta}=\frac{\delta^{n+1}-\delta^n}{\Delta t}.$$

Then we obtain iterative formula for the time parameters δ^n

$$\left[\mathbf{A}+\frac{\Delta t}{2}\left(\mathbf{B}\left(\delta^{n+1}\right)-\nu\mathbf{C}\right)\right]\delta^{n+1}=\left[\mathbf{A}-\frac{\Delta t}{2}\left(\mathbf{B}\left(\delta^{n}\right)-\nu\mathbf{C}\right)\right]\delta^{n}. \tag{12}$$

The set of equations consist of $(N+3)$ equations with $(N+3)$ unknown parameters. Boundary conditions must be adapted into the system. Because of the this requirement, initially the first and last equations are eliminated from the (12) and parameters δ^{n+1}_{-1} and δ^{n+1}_{N+1} are substituted in the remaining system (12) by using following equations

$$u(a,t)=\alpha_1\delta^{n+1}_{-1}+\delta^{n+1}_0+\alpha_1\delta^{n+1}_1=\beta_1,\quad u(b,t)=\alpha_1\delta^{n+1}_{N-1}+\delta^{n+1}_N+\alpha_1\delta^{n+1}_{N+1}=\beta_2$$

which are obtained from the boundary conditions. Thus we obtain a septa-diagonal matrix with the dimension $(N+1)\times(N+1)$. Since the system (12) is an implicit system together with the nonlinear term $\mathbf{B}\left(\delta^{n+1}\right)$, we have used the following inner iteration at each time step $(n+1)\Delta t$ to work up solutions:

$$(\delta^*)^{n+1}=\delta^n+\frac{(\delta^n-\delta^{n-1})}{2}. \tag{13}$$

We use the above iteration three times to find the new approximation $(\delta^*)^{n+1}$ for the parameters δ^{n+1} to recover solutions at time step $(n+1)\Delta t$.

To start evolution of the iterative system for the unknown δ^n, the vector of initial parameters δ^0 must be determined by using the following initial and boundary conditions:

$$u'(x_0,0) = \frac{p(1-c)}{2(phc-s)}\delta_{-1} + \frac{p(c-1)}{2(phc-s)}\delta_1,\ u'(x_N,0) = \frac{p(1-c)}{2(phc-s)}\delta_{N-1} + \frac{p(c-1)}{2(phc-s)}\delta_{N+1}$$
$$u(x_m,0) = \frac{s-ph}{2(phc-s)}\delta_{m-1} + \delta_m + \frac{s-ph}{2(phc-s)}\delta_{m+1},\ m=0,...,N. \tag{14}$$

The solution of matrix equation (14) with the dimensions $(N+1)\times(N+1)$ is obtained by the way of Thomas algorithm. Once δ^0 is determined, we can start the iteration of the system to find the parameters δ^n at time $t^n = n\Delta t$. Approximate solutions at the knots are found from the (6).

3. Test Problems

The robustness of the algorithm is shown by studying two test problems. Error is measured by the maximum error norm;

$$L_\infty = \left\|u^{\text{exact}} - u^{\text{numeric}}\right\|_\infty = \max_{0\le j\le N}\left|u_j^{\text{exact}} - u_j^{\text{numeric}}\right|. \tag{15}$$

The free parameter p of the exponential B-spline is found by scanning the predetermined interval with very small increment.

(a) A shock propagation solution of the Burgers' equation is

$$u(x,t) = \frac{x/t}{1+\sqrt{t/t_0}\exp(x^2/(4\nu t))}, \quad t\ge 1, \tag{16}$$

where $t_0 = \exp(1/(8\nu))$. The sharpness of the solutions increase with selection of the smaller ν.

Substitution of the $t=1$ in (16) gives the initial condition. The boundary conditions $u(0,t)=0$ and $u(1,t)=0$ are used. Computations are performed with parameters $\nu = 0.0005$, 0.005, 0.01, $h = 0.02$, 0.005 and $\Delta t = 0.01$ over the solution domain $[0,1]$. As time increases, shock evaluation is observed and some graphical solutions are drawn in Figs. 1 and 2 for various viscosity values and space steps. For $\nu = 0.01$, algorithm produces smoother shock during run

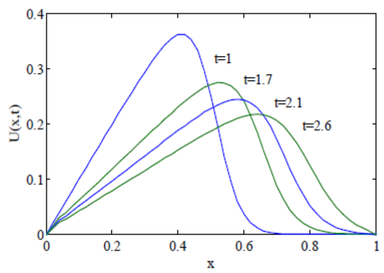

Figure 1. Solutions for $\nu = 0.01$, $h = 0.02$, $p = 0.005111$.

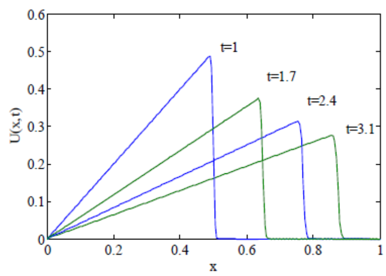

Figure 2. Solutions for $\nu = 0.0005$, $h = 0.005$, $p = 0.005941$.

time. With decreasing values of ν, as seen in the Fig. 2 the steepening occurs. For the smaller viscosity constant $\nu = 0.0005$, the sharper shock is observed and steepness of numerical solution is kept almost unchanged during the program run. The results obtained by present scheme can be compared with previous ones through the computation of error norm L_∞ at various times in the Table 1.

Table 1. Comparison of numerical results at different times.

	$L_\infty \times 10^3$	$L_\infty \times 10^3$	$L_\infty \times 10^3$
$h = 0.005,\ \nu = 0.005$	$t = 1.7$	$t = 2.4$	$t = 3.1$
Present ($p = 0.005941$)	3.15776	2.33757	4.79061
Ref. [3] (QBGM)	1.20755	0.80187	4.79061
Ref. [6] (QBCM1)	0.06192	0.05882	4.43469
Ref. [7] (QBCA1)	1.21175	0.80771	4.79061
Ref. [8]	0.04284	0.06464	4.79061
$h = 0.02,\ \nu = 0.01$	$t = 1.7$	$t = 2.1$	$t = 2.6$
Present ($p = 0.005111$)	8.08651	7.53518	8.06798
Ref. [5]	3.13476	2.66986	8.06798
Ref. [6] (QBCM1)	0.40431	0.86363	6.69425
Ref. [7] (QBCA1)	0.47456	1.14759	8.06798
Ref. [8]	0.09592	1.14760	8.06799

The absolute error distributions between the analytical and numerical solutions are drawn in Figs. 3 and 4 for various viscosity values and space steps.

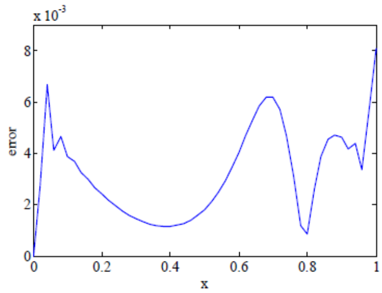

Figure 3. Absolute error for $\nu = 0.01$, $h = 0.02$, $p = 0.005111$.

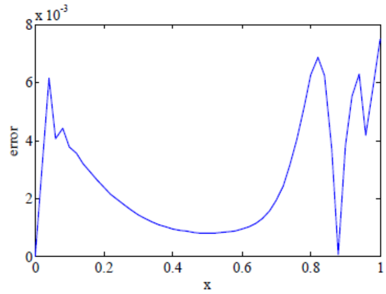

Figure 4. Absolute error for $\nu = 0.005$, $h = 0.02$, $p = 0.000739$.

(b) A well-known analytical solution of Burgers' equation is

$$u(x,t) = \frac{\alpha + \mu + (\mu - \alpha)\exp\eta}{1 + \exp\eta}, \quad 0 \le x \le 1,\ t \ge 0, \tag{17}$$

where $\eta = \dfrac{\alpha\,(x - \mu t - \gamma)}{\nu}$. α, μ and γ are constants. Parameters $\alpha = 0.4$, $\mu = 0.6$ and $\gamma = 0.125$ are used to coincide with the some previous studies. This solution involves a travelling wave and move to the right with speed μ. Initial condition is obtained from (17) when $t = 0$. The boundary conditions are $u(0,t) = 1$, $u(1,t) = 0.2$ for $t > 0$.

The calculation is performed with time step $\Delta t = 0.01$, space step $h = 1/36$ and viscosity coefficient $\nu = 0.01$. The program is run up to time $t = 0.5$. We have found $L_\infty = 6.73543978 \times 10^{-4}$ for the exponential B-spline Galerkin method at time $t = 0.5$ documented in Table 2 with results of the quadratic B-spline Galerkin method [3], the quartic B-spline collocation method [6], the quintic B-spline collocation method [7] and the quartic B-spline Galerkin method [8].

The numerical solution obtained by the present scheme gives better results than the others. The profiles of initial wave and solution at some times are depicted in Fig. 5. Error variations of the scheme are given in Fig. 6 at time $t = 0.5$.

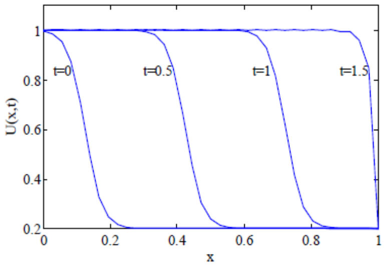

Figure 5. Solutions for $\nu = 0.01$.

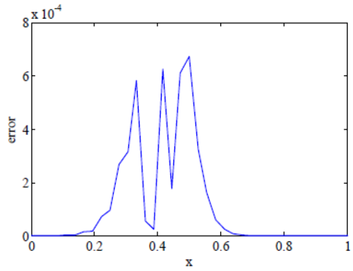

Figure 6. Absolute error for $\nu = 0.01$.

Table 2. Comparison of results at $t = 0.5$ for $h = 1/36$, $\nu = 0.01$.

	Present ($p = 0.002323$)	Ref. [3] (QBGM)	Ref. [6] (QBCM1)	Ref. [7] (QBCA1)	Ref. [8] (QBGM)
$L_\infty \times 10^3$	0.67354	6.35489	3.03817	5.78454	1.44

4. Conclusion

In this paper, we investigate the utility of the exponential B-spline in the Galerkin algorithm for solving the Burgers' equation. The efficiency of the method is tested for a shock propagation solution and a travelling solution of the Burgers' equation. For the first test problem, solutions found with the present methods are in good agreement with the results obtained by previous studies. In the second test problem, present method leads to accurate results than all of the others. In conclude, the numerical algorithm in which the exponential B-spline functions are used, performs well compared with other existing numerical methods for the solution of Burgers' equation.

Acknowledgments

The authors are grateful to The Scientific and Technological Research Council of Turkey for financial support given for the project 113F394. This paper is presented at the conference of ICQSA 2016.

References

[1] McCartin B J 1991 *J. Approx. Theory.* **66** 1–23
[2] Kutluay S, Esen A and Dag I 2004 *J. Comput. Appl. Math.* **167** 21–33
[3] Dag I, Saka B and Boz A 2005 *Appl. Math. Comput.* **166** 506–22
[4] Dag I, Irk D and Sahin A 2005 *Math. Probl. Eng.* **2005** 521–38
[5] Ramadan M A, El-Danaf T S and Abd Alaal F E I 2005 *Chaos. Soliton. Fract.* **26** 795–804
[6] Saka B and Dag I 2007 *Chaos. Soliton. Fract.* **32** 1125–37
[7] Saka B and Dag I 2008 *J. Franklin. I.* **345** 328–48
[8] Saka B and Dag I 2009 *Appl. Math. Comput.* **215** 746–58
[9] Zhu C and Wang R 2009 *Appl. Math. Comput.* **208** 260–72
[10] Jiang Z and Wang R 2010 *J. Inform. Comput. Sci.* **7** 1013–21
[11] Mittal R C and Jain R K 2012 *Appl. Math. Comput.* **218** 7839–55
[12] Chapani H V, Pradhan V H and Mehta M N 2012 *Int. J. Appl. Math. Mech.* **8** 18–32
[13] Mohammadi R 2013 *Appl. Math.* **4** 933–44
[14] Dag I and Ersoy O 2015 *AIP Conf. P.* **1648** 370008