Research Article

The Exponential Cubic B-Spline Algorithm for Korteweg-de Vries Equation

Ozlem Ersoy and Idris Dag

Department of Mathematics-Computer, Faculty of Science and Art, Eskişehir Osmangazi University, 26480 Eskişehir, Turkey

Correspondence should be addressed to Ozlem Ersoy; ozersoy@ogu.edu.tr

The exponential cubic B-spline algorithm is presented to find the numerical solutions of the Korteweg-de Vries (KdV) equation. The problem is reduced to a system of algebraic equations, which is solved by using a variant of Thomas algorithm. Numerical experiments are carried out to demonstrate the efficiency of the suggested algorithm.

1. Introduction

The splines consist of piecewise functions defined on the distributed knots on problem domain and have certain continuity inside problem subdomain and at the knots. Until now, some types of splines have been developed and especially polynomial splines. The exponential splines are defined as more general splines by McCartin [1–3]. The basis of the exponential splines known as the exponential B-splines is also given in the studies of McCartin. Existence of the free parameter in the exponential B-splines yields the different shapes of the splines functions. He has also showed a reliable algorithm by using the exponential spline functions to solve the hyperbolic conservation laws, McCartin and Jameson [4]. However, McCartin stated that application of the exponential spline/exponential B-spline functions has been neglected in the numerical analysis. So, use of the exponential spline in the numerical methods for finding solutions of the differential equations is not common and few papers exist in the literature. McCartin has shown that the exponential splines admit a basis known as the exponential B-splines. These B-splines have been started using to form approximates functions recently which are adapted to set up the numerical methods to find solutions of the differential equations recently. An application of the simple exponential splines is considered for setting up the collocation method to solve the numerical solution of singular perturbation problem [5].

Cardinal exponential B-splines are applied in solving singularly perturbed boundary problems [6]. A variant of B-spline exponential collocation method was also built up for computing numerical solutions of the singularly perturbed boundary value problem [7]. Very recently, the exponential B-spline collocation method has been applied to obtain the solutions of the one-dimensional linear convection-diffusion equation [8].

Types of spline functions are utilized to form approximate solutions for Korteweg-de Vries equation (KdVE). The standard Galerkin formulation using the smooth splines on uniform mesh is set up for 1-periodic solutions of KdVE by Baker and his coauthors [9]. The Galerkin finite element method together with the cubic B-splines is used to solve the KdVE in the paper [10]. The quadratic B-spline Galerkin method is built up to find solutions of the KdVE [11]. A collocation solution of the Korteweg-de Vries equation using septic B-splines is proposed by Soliman [12]. A variant of the Galerkin finite element method is designed for solving the KdVE by Aksan and Özdeş [13]. The collocation method using quintic B-splines is developed to solve the KdVE [14]. A numerical method is developed for the KdVE by using splitting finite difference technique and quintic B-spline functions [15]. The spline finite element method using quadratic polynomial spline for the numerical solution of the KdVE is given by G. Micula and M. Micula [16]. A cubic B-spline Taylor-Galerkin method is developed to find numerical solution of the KdVE

TABLE 1: Values of $B_i(x)$ and its first and second 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

by Canıvar et al. in [17]. A study based on cubic B-spline finite element method for the solution of the KdVE is suggested by Kapoor et al. [18]. A Bubnov-Galerkin finite element method with quintic B-spline functions taken as element shape and weight functions is presented for the solution of the KdVE [19]. The paper deals with the numerical solution of the KdVE using quartic B-splines Galerkin method as both shape and weight functions over the finite intervals [20]. A blended spline quasi-interpolation scheme is employed to solve the one-dimensional nonlinear KdVE [21]. A multilevel quartic spline quasi-interpolation scheme is fulfilled to exhibit a large number of physical phenomena for KdVE [22].

The aim of the present paper is to develop an approximate solution of KdVE by collocation method. In Section 2, the exponential B-spline collocation algorithm is defined for the KdVE. In Section 3, the three numerical experiments are constructed to demonstrate the efficiency of the proposed method and the results are documented in tables and graphs are depicted.

We will solve the KdVE:

$$U_t + \varepsilon U U_x + \mu U_{xxx} = 0, \quad a \le x \le b, \tag{1}$$

where ε, μ are positive parameters and the subscripts x and t denote differentiation. The boundary conditions will be chosen as

$$\begin{aligned} U(a,t) &= U(b,t) = 0, \\ U_x(a,t) &= U_x(b,t) = 0. \end{aligned} \tag{2}$$

KdVE is prototypical example of exactly solvable mathematical model of waves on shallow water surface. It arises for evolution, interaction of waves, and generation in physics. Due to the term U_t, (1) is called the evolution equation, the nonlinear term causes the steepness of the wave, and the dispersive term defines the spreading of the wave. It is known that the effect of the steepness and spreading results in soliton solutions for the KdVE.

2. Exponential B-Spline Collocation Method

The region $[a,b]$ is partitioned into equal subintervals by points x_i, $i = 0,\ldots,N$. On these points together with additional points x_i, $i = -3, -2, -1, N+1, N+2, N+3$ outside

the domain, the exponential B-splines, $B_i(x)$, can be defined as

$B_i(x)$

$$= \begin{cases} b_2\left((x_{i-2}-x) - \dfrac{1}{p}\left(\sinh\left(p\left(x_{i-2}-x\right)\right)\right)\right) & [x_{i-2}, x_{i-1}], \\ a_1 + b_1(x_i-x) + c_1 \exp\left(p\left(x_i-x\right)\right) \\ \quad + d_1 \exp\left(-p\left(x_i-x\right)\right) & [x_{i-1}, x_i], \\ a_1 + b_1(x-x_i) + c_1 \exp\left(p\left(x-x_i\right)\right) \\ \quad + d_1 \exp\left(-p\left(x-x_i\right)\right) & [x_i, x_{i+1}], \\ b_2\left((x-x_{i+2}) - \dfrac{1}{p}\left(\sinh\left(p\left(x-x_{i+2}\right)\right)\right)\right) & [x_{i+1}, x_{i+2}], \\ 0 & \text{otherwise,} \end{cases} \tag{3}$$

where

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

$$b_2 = \frac{p}{2\,(phc-s)},$$

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

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

$$s = \sinh\left(ph\right), \qquad c = \cosh\left(ph\right),$$

$$p \text{ is a free parameter}, \qquad h = \frac{b-a}{N}. \tag{4}$$

$\{B_{-1}(x), B_0(x), \ldots, B_{N+1}(x)\}$ forms a basis for the exponential spline space on the interval $[a,b]$. On the four consecutive subintervals, an exponential B-spline $B_i(x)$ is defined and it is second-order continuously differentiable functions.

In Table 1, the values of $B_i(x)$, $B_i'(x)$, and $B_i''(x)$ at the points x_i, which can be obtained from (3) are listed where $'$ denotes differentiation with respect to space variable x.

The global approximation $U_N(x,t)$ to the solution $U(x,t)$ will be searched in terms of the unknown parameters δ_i and exponential B-spline function defined on the problem domain:

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

Substitution of the points x_i in (5), in its first and its second derivatives, respectively, yields the numerical solution in terms of parameters:

$$U_i = U(x_i, t) = m_1 \delta_{i-1} + \delta_i + m_1 \delta_{i+1},$$

$$U_i' = U'(x_i, t) = m_2 \delta_{i-1} - m_2 \delta_{i+1}, \quad (6)$$

$$U_i'' = U''(x_i, t) = m_3 \delta_{i-1} - 2m_3 \delta_i + m_3 \delta_{i+1},$$

where $m_1 = (s - ph)/2(phc - s)$, $m_2 = p(1 - c)/2(phc - s)$, and $m_3 = p^2 s/2(phc - s)$.

Over the subregion $[x_i, x_{i+1}]$, the local approximation is given by

$$U_N^e(x,t) = \delta_{i-1} B_{i-1}(x) + \delta_i B_i(x)$$
$$+ \delta_{i+1} B_{i+1}(x) + \delta_{i+2} B_{i+2}(x), \quad (7)$$

where δ_{i-1}, δ_i, and δ_{i+1} act as subregion parameters and B_{i-1}, B_i, and B_{i+1} are known as the subregion shape parameters.

To be able to apply the collocation method formed with the exponential B-splines, KdV equation is space-splitted as

$$U_t + \varepsilon U U_x + \mu V_{xx} = 0,$$
$$U_x - V = 0. \quad (8)$$

This system includes the second-order derivatives so that smooth approximation can be done with the exponential B-splines. To integrate system (8) in time, discretize U_t by the usual finite difference scheme and U, U_x, and V_{xx} by Crank-Nicolson method and we get

$$\frac{U^{n+1} - U^n}{\Delta t} + \varepsilon \frac{(UU_x)^{n+1} + (UU_x)^n}{2} + \mu \frac{V_{xx}^{n+1} + V_{xx}^n}{2} = 0,$$

$$\frac{U_x^{n+1} + U_x^n}{2} - \frac{V^{n+1} + V^n}{2} = 0, \quad (9)$$

where $U^{n+1} = U(x, (n+1)\Delta t)$ represent the solution 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, $t^n = n\Delta t$.

One linearizes terms $(UU_x)^{n+1}$ and $(UUx)^n$ in (9) as

$$(UUx)^{n+1} = U^{n+1} U_x^n + U^n U_x^{n+1} - U^n U_x^n,$$
$$(UUx)^n = U^n U_x^n \quad (10)$$

to obtain the time-integrated KdVE:

$$U^{n+1} - U^n + \frac{\varepsilon \Delta t}{2} \left(U^{n+1} U_x^n + U^n U_x^{n+1} \right)$$

$$- \frac{\mu \Delta t}{2} \left(V_{xx}^{n+1} + V_{xx}^n \right) = 0, \quad (11)$$

$$\frac{U_x^{n+1} + U_x^n}{2} - \frac{V^{n+1} + V^n}{2} = 0.$$

We approximate U^n and V^n in terms of the element parameters and exponential B-splines separately as

$$U_N(x,t) = \sum_{i=-1}^{N+1} \delta_i B_i(x), \qquad V_N(x,t) = \sum_{i=-1}^{N+1} \phi_i B_i(x). \quad (12)$$

Putting the approximate solution (12) and its derivatives into (11) and evaluating the resulting equations at the points x_i, $i = 0, \ldots, N$ yield the following system of equations:

$$m_1 \delta_{m-1}^{n+1} - \frac{\mu \Delta t}{2} \gamma \phi_{m-1}^{n+1} + \delta_m^{n+1}$$

$$+ \mu \Delta t \gamma \phi_m^{n+1} + m_1 \delta_{m+1}^{n+1} - \frac{\mu \Delta t}{2} \gamma \phi_{m+1}^{n+1}$$

$$= m_1 \left(1 - \frac{\varepsilon \Delta t}{2} L \right) \delta_{m-1}^n - \frac{\Delta t}{2} \left(\mu \gamma + \varepsilon m_1 K \right) \phi_{m-1}^n$$

$$+ \left(1 - \frac{\varepsilon \Delta t}{2} L \right) \delta_{m-1}^n - \frac{\Delta t}{2} \left(-2\mu \gamma + \varepsilon K \right) \phi_{m-1}^n \quad (13)$$

$$+ m_1 \left(1 - \frac{\varepsilon \Delta t}{2} L \right) \delta_{m-1}^n - \frac{\Delta t}{2} \left(\mu \gamma + \varepsilon m_1 K \right) \phi_{m-1}^n,$$

$$\beta \delta_{m-1}^{n+1} - m_1 \phi_{m-1}^{n+1} - \phi_m^{n+1} - \beta \delta_{m+1}^{n+1} - m_1 \phi_{m+1}^{n+1}$$

$$= -\beta \delta_{m-1}^n + m_1 \phi_{m-1}^n + \phi_m^n + \beta \delta_{m+1}^n + m_1 \phi_{m+1}^n,$$

$$m = 0, \ldots, N, \quad n = 0, 1, \ldots,$$

where

$$K = m_1 \delta_{i-1} + \delta_i + m_1 \delta_{i+1},$$

$$L = m_1 \phi_{i-1} + \phi_i + m_1 \phi_{i+1}, \quad (14)$$

$$\beta = \frac{p(1-c)}{2(phc - s)}, \qquad \gamma = \frac{p^2 s}{2(phc - s)}.$$

The system consists of $2N + 2$ linear equation in $2N + 6$ unknown parameters $\mathbf{d}^{n+1} = (\delta_{-1}^{n+1}, \phi_{-1}^{n+1}, \delta_0^{n+1}, \phi_0^{n+1}, \ldots, \delta_{N+1}^{n+1}, \phi_{N+1}^{n+1})$. A unique solution of the system can be obtained by imposing the boundary conditions $U(a,t) = 0$, $U(b,t) = 0$, $V(a,t) = 0$, $V(b,t) = 0$ to have the following the equations:

$$m_1 \delta_{-1} + \delta_0 + m_1 \delta_1 = 0,$$

$$m_1 \phi_{-1} + \phi_0 + m_1 \phi_1 = 0,$$

$$m_1 \delta_{N-1} + \delta_N + m_1 \delta_{N+1} = 0, \quad (15)$$

$$m_1 \phi_{N-1} + \phi_N + m_1 \phi_{N+1} = 0.$$

Elimination of the parameters δ_{-1}, ϕ_{-1}, δ_{N+1}, ϕ_{N+1}, using (15) from system (13), gives a solvable system of $2N + 2$ linear equation including $2N + 2$ unknown parameters. Placing solution parameters in (12) when computed from the system via a variant of the Thomas algorithm gives the approximate solution over the subregion $[a, b]$. We need the initial parameter vectors $d_1 = (\delta_{-1}, \delta_0, \ldots, \delta_N, \delta_{N+1})$, $d_2 = (\phi_{-1}, \phi_0, \ldots, \phi_N, \phi_{N+1})$ to start the iteration process for system of (13). To do that, the following requirements help to determine the initial parameters:

$$
(U_N)_x (a, 0) = 0 = m_2 \delta_{-1}^0 + m_2 \delta_1^0,
$$

$$
(U_N)_x (x_i, 0) = m_1 \delta_{i-1}^0 + \delta_i^0 + m_1 \delta_{i+1}^0 = U(x_i, 0),
$$

$$
i = 0, \ldots, N,
$$

$$
(U_N)_x (b, 0) = 0 = m_2 \delta_{N-1}^0 + m_2 \delta_{N+1}^0,
$$

$$
(V_N)_x (a, 0) = 0 = m_2 \phi_{-1}^0 + m_2 \phi_1^0, \tag{16}
$$

$$
(V_N)_x (x_i, 0) = m_1 \phi_{i-1}^0 + \phi_i^0 + m_1 \phi_{i+1}^0 = V(x_i, 0),
$$

$$
i = 0, \ldots, N,
$$

$$
(V_N)_x (a, 0) = m_2 \phi_{N-1}^0 + m_2 \phi_{N+1}^0.
$$

3. Numerical Tests

Since the conservation laws remain constant at all time, first three numerical conservations are calculated using the rectangular rule for integrals:

$$
C_1 = \int_{-\infty}^{\infty} U \, dx \approx \int_a^b U \, dx \approx \frac{h}{2} \sum_{i=0}^{N-1} (U_i + U_{i+1}),
$$

$$
C_2 = \int_{-\infty}^{\infty} (U^2) \, dx \approx \int_a^b (U^2) \, dx \approx \frac{h}{2} \sum_{i=0}^{N-1} \left[(U^2)_i + (U^2)_{i+1} \right],
$$

$$
C_3 = \int_{-\infty}^{\infty} \left(U^3 - \frac{3\mu}{\varepsilon} U_x^2 \right) dx \approx \int_a^b \left(U^3 - \frac{3\mu}{\varepsilon} U_x^2 \right) dx
$$

$$
\approx \frac{h}{2} \sum_{i=0}^{N-1} \left[(U^3)_i - \frac{3\mu}{\varepsilon} (U_x^2)_i + (U^3)_{i+1} - \frac{3\mu}{\varepsilon} (U_x^2)_{i+1} \right]. \tag{17}
$$

The error norm L_∞

$$
L_\infty = |U - U_N|_\infty = \max_j \left| U_j - (U_N)_j^n \right| \tag{18}
$$

is calculated to show the error between analytical and numerical solutions.

(a) The soliton solution of the KdVE is

$$
U(x, t) = 3c \sec h^2 (Ax - Bt + D), \tag{19}
$$

where $A = (1/2)\sqrt{\varepsilon c/\mu}$ and $B = \varepsilon c A$. This solution represents propagation of single soliton, having velocity εc and amplitude $3c$.
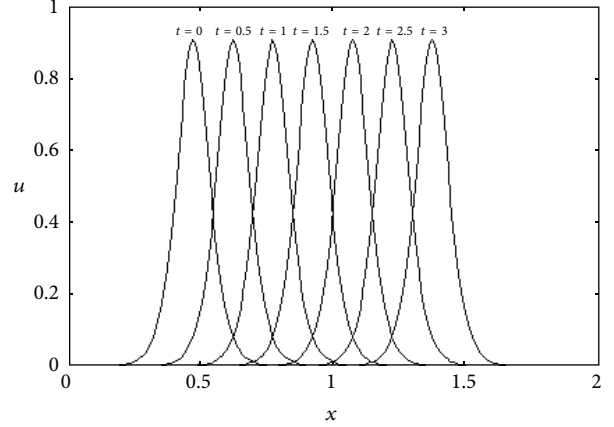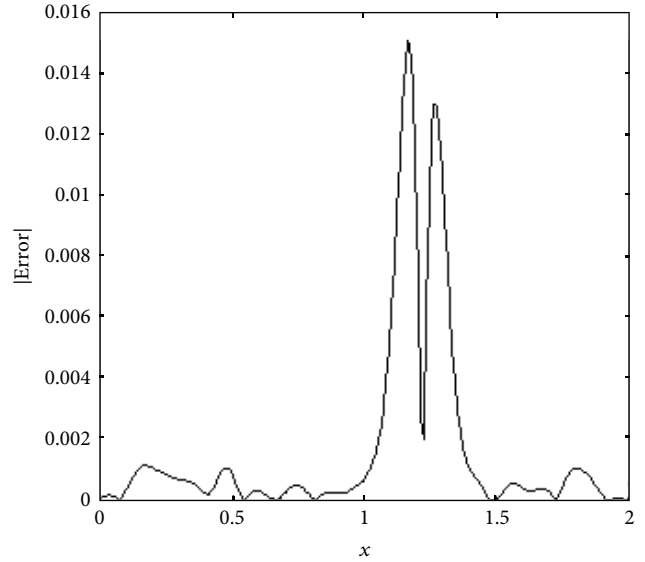

FIGURE 1: A single soliton at some times.

FIGURE 2: Error distribution.

The analytical solution (19) is used as the initial condition when $t = 0$. The Dirichlet boundary conditions $U(0, t) = V(0, t) = 0$ and $U(2, t) = V(2, t) = 0$ are adapted to the system to control numerical solutions at the boundaries. Parameters $\varepsilon = 1$, $\mu = 4.84 \times 10^{-4}$, $c = 0.3$, $D = -6$, space step $h = 0.01$, and time step $\Delta t = 0.005$ on the interval $[0, 2]$ from time $t = 0$ to $t = 3$ are chosen. At time $t = 3$, numerical magnitude of the single soliton is calculated as 0.9, so that the numerical amplitude is obtained to be almost the same as the analytical amplitude. Figure 1 illustrates the amplitudes at some times. The distribution of the absolute values of errors can be observed in Figure 2.

L_∞ error norms and invariants are presented at the selected times in Table 2, as seen from the table that L_2 error norm is found small enough and conservation invariants are excellent throughout the simulation. The method gives good results when the free parameter $p = 1.64 \times 10^{-5}$ is used. Invariants C_2 and C_3 remain constant during the run and C_1

TABLE 2: Error norms and invariants for single soliton.

Time	$L_\infty \times 10^3$ $p \times 10^5$	$L_\infty \times 10^3$ $p = 1$	C_1	C_2	C_3	Method
0.0	0	0	0.144597	0.086759	0.046849	
0.5	0.521 ($p = 1.37$)	0.51	0.144602	0.086759	0.046849	
1.0	0.589 ($p = 1.64$)	0.75	0.144593	0.086759	0.046849	
1.5	0.533 ($p = 1.64$)	1.03	0.144592	0.086759	0.046849	Pre.
2.0	0.595 ($p = 1.64$)	1.26	0.144591	0.086759	0.046849	
2.5	0.657 ($p = 1.64$)	1.50	0.144597	0.086759	0.046849	
3.0	0.740 ($p = 1.64$)	1.61	0.144597	0.086759	0.046849	
3.0	0.08		0.014460	0.08676	0.04685	[23]
3.0	1.04		0.014460	0.08675	0.04685	[24]
3.0	0.04		0.144597	0.086761	0.0468524	[17]
3.0	0.14		0.144601	0.086760	0.046850	[25]

TABLE 3: Invariant for Maxwellian.

	$\varepsilon = 1.0, \ \mu = 0.04$		
t	C_1	C_2	C_3
0	1.77245	1.25331	0.87292
2.5	1.77243	1.25332	0.87412
5.0	1.77248	1.25333	0.87437
7.5	1.77170	1.25332	0.87442
10.0	1.77203	1.25333	0.87442
12.5	1.77367	1.25330	0.87442

TABLE 4: Invariant for Maxwellian.

	$\varepsilon = 1.0, \ \mu = 0.01$		
t	C_1	C_2	C_3
0	1.77245	1.25331	0.98572
2.5	1.77244	1.25362	0.99277
5.0	1.77250	1.25374	0.99558
7.5	1.77243	1.25376	0.99576
10.0	1.77242	1.25377	0.99582
12.5	1.77222	1.25378	0.99582

TABLE 5: Invariant for Maxwellian.

	$\varepsilon = 1.0, \ \mu = 0.001$		
t	C_1	C_2	C_3
0	1.77245	1.25331	1.01956
2.5	1.77245	1.25356	1.02461
5.0	1.77245	1.25365	1.02644
7.5	1.77245	1.25365	1.02654
10.0	1.77244	1.25365	1.02656
12.5	1.77244	1.25365	1.02656

remains the constant up to the third decimal digits seen in Table 2.

(b) Wave generation is performed by using the Maxwellian initial condition:

$$U(x, 0) = \exp\left(-x^2\right) \qquad (20)$$

and boundary conditions:

$$U(-15, t) = U(15, t) = 0, \quad t > 0. \qquad (21)$$

$h = 0.1$ and $\Delta t = 0.01$ and $\varepsilon = 1.0$ are taken. We have verified the case in which μ_c is some critical parameter and, according to the parameters $\mu \ll \mu_c$, initial condition breaks up into a number of solitons and, for values $\mu \gg \mu_c$, soliton turns into exhibiting the rapidly oscillating wave packets. When $\mu \approx \mu_c$ together with parameters $\varepsilon = 1.0$, $\mu = 0.04$, $h = 0.1$, and $\Delta t = 0.01$, the solution takes the form of the leading soliton and an oscillating tail. This case is shown in Figure 3.
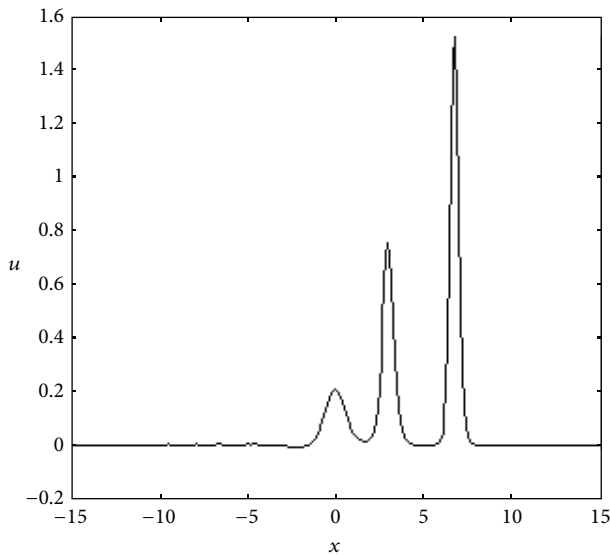
For $\mu = 0.04$, we observe a solitary wave plus an oscillating tail (Figure 3). The actual velocity of the wave V_n has been measured and also computed from the measured amplitude using the formula $V_a = a\varepsilon/3$. We find that $V_n = 0.4$ and $V_a = (1.1978 \times 1)/3 = 0.39926$, so the solitary waves are indeed solitons. In Table 3 invariant for Maxwellian $\varepsilon = 1.0$ and $\mu = 0.04$.

When $\mu = 0.01$, we find three solitons. We have measured the velocity of the largest solitary wave as $V_n = 0.52$ and calculated the expected velocity from the observed amplitude 1.54468 as $V_a = (1.5535 \times 1)/3 = 0.51783$. In Figure 4, Maxwellian initial condition is depicted for $t = 12.5$, $\varepsilon = 1.0$, $\mu = 0.01$, $h = 0.1$, and $\Delta t = 0.01$. The invariants are given in Tables 4 and 5 for $\varepsilon = 1.0$, $\mu = 0.01$ and $\varepsilon = 1.0$, $\mu = 0.001$, respectively.

For $\mu = 0.001$, we observed nine solitons moving to the right in Figure 5. The measured velocity of leading soliton is $V_n = 0.62$ and the corresponding velocities calculated from their measured amplitudes are $V_a = (1.7949 \times 1)/3 = 0.60$. The agreement is good. The initial perturbation breaks up into a number of solitons in the course of time depending on the value of μ chosen. So, if we decrease the value of μ, then the number of solitons, amplitude, and the velocity increase.
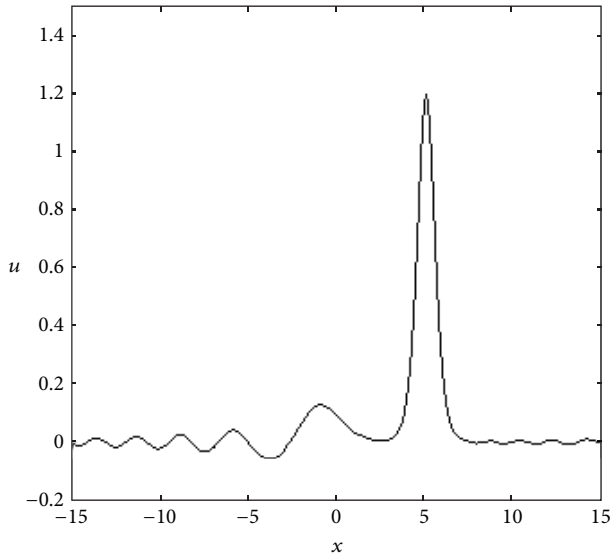
FIGURE 3: Soliton at $t = 12.5$.

FIGURE 5: $t = 12.5$ Maxwellian initial condition; $\varepsilon = 1.0$, $\mu = 0.001$, $h = 0.025$, and $\Delta t = 0.005$.

FIGURE 4: Generated waves at $t = 12.5$, $\varepsilon = 1.0$, $\mu = 0.01$, $h = 0.1$, and $\Delta t = 0.01$.

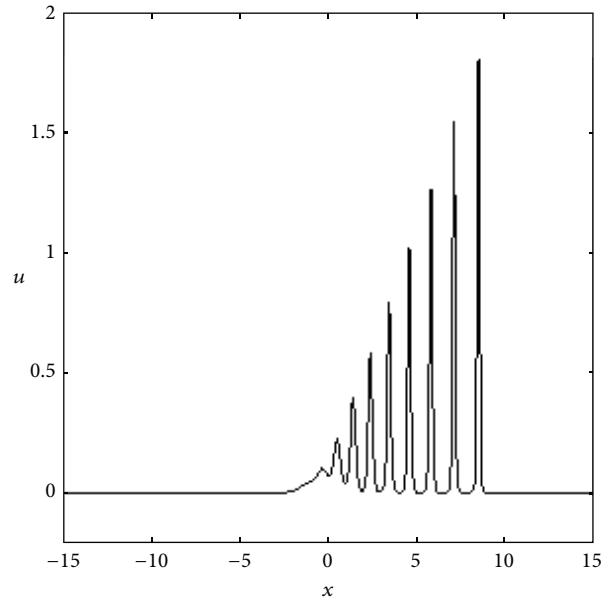TABLE 6: Conservation laws $\varepsilon = 0.2$, $\mu = 0.1$.

t	C_1	C_2	C_3
0	50.0001	45.0004	40.6207
100	49.9999	45.0011	40.4055
200	50.0009	45.0032	40.5267
300	49.9996	45.0044	40.4286
400	49.9989	45.0048	42.6020
500	49.9984	45.0049	40.6564
600	49.9997	45.0051	40.6786
700	50.0009	45.0051	40.4270
800	50.0011	45.0049	40.4629

(c) As a final test example, initial condition

$$U(x, 0) = \frac{1}{2}\left[1 - \tanh\left(\frac{|x| - 25}{5}\right)\right] \qquad (22)$$

together with boundary conditions

$$U(-50, t) = U(150, t) = 0, \quad t > 0 \qquad (23)$$

cause the production of a train of solitons depending on the value of μ for the KdVE. Computation is done on region $[-50, 150]$ up to time $t = 800$ with parameters $\varepsilon = 0.2$, $\mu = 0.1$, $\Delta t = 0.05$, and $h = 0.4$. Visual representation of the solution in Figures 6(a)–6(f) is drawn that 10 solitons have been broken up from the given initial condition.

The first three conservation laws are recorded at some times in Table 6. These are favorably constant. The observed velocity of the leading soliton having the amplitude 1.96342 is $V_n = 0.128$ which was in close agreement with that calculated from its observed amplitude of $V_a = (1.9379 \times 0.2)/3 = 0.129$.

4. Conclusion

The numerical solution of the KdVE is obtained by the collocation method using the exponential basis functions. Performance of the present method is shown by calculating L_∞—the error norm, and conservation laws. The present method gives accurate results and simulations such as the propagation of soliton and generation of waves which are substantiated fairly. Using the exponential cubic B-splines, alternative numerical methods can be set up for finding numerical solutions of the differential equations with high accuracy when an appropriate free parameter is chosen.
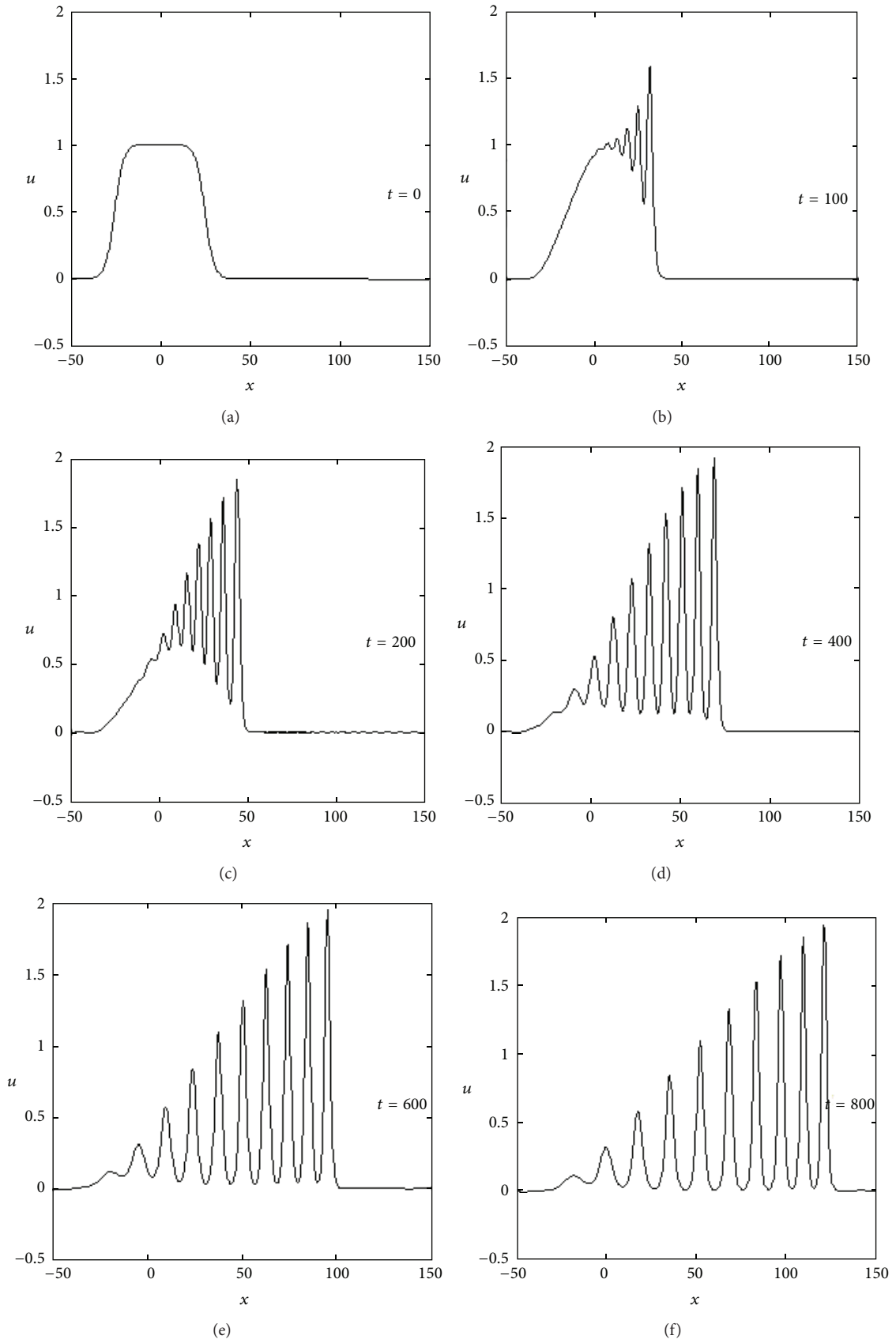
FIGURE 6: (a) The solution graph for $t = 0$. (b) The solution graph for $t = 100$. (c) The solution graph for $t = 200$. (d) The solution graph for $t = 400$. (e) The solution graph for $t = 600$. (f) The solution graph for $t = 800$.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] B. J. McCartin, "Theory computation and application of exponential splines," Tech. Rep. DOE/ER/03077-171, 1981.

[2] B. J. McCartin, "Computation of exponential splines," *SIAM Journal on Scientific and Statistical Computing*, vol. 11, no. 2, pp. 242–262, 1990.

[3] B. J. McCartin, "Theory of exponential splines," *Journal of Approximation Theory*, vol. 66, no. 1, pp. 1–23, 1991.

[4] B. J. McCartin and A. Jameson, "Numerical solution of nonlinear hyperbolic conservation laws using exponential splines," *Computational Mechanics*, vol. 6, no. 2, pp. 77–91, 1990.

[5] M. Sakai and R. A. Usmani, "A class of simple exponential *B*-splines and their application to numerical solution to singular perturbation problems," *Numerische Mathematik*, vol. 55, no. 5, pp. 493–500, 1989.

[6] D. Radunovic, "Multiresolution exponential B-splines and singularly perturbed boundary problem," *Numerical Algorithms*, vol. 47, no. 2, pp. 191–210, 2008.

[7] S. C. Rao and M. Kumar, "Exponential B-spline collocation method for self-adjoint singularly perturbed boundary value problems," *Applied Numerical Mathematics*, vol. 58, no. 10, pp. 1572–1581, 2008.

[8] R. Mohammadi, "Exponential B-spline solution of convection-diffusion equations," *Applied Mathematics*, vol. 4, no. 6, pp. 933–944, 2013.

[9] G. A. Baker, V. A. Dougalis, and O. A. Karakashian, "Convergence of Galerkin approximations for the Korteweg-de Vries equation," *Mathematics of Computation*, vol. 40, no. 162, pp. 419–433, 1983.

[10] G. A. Gardner and L. R. T. Gardner, "A finite element solution for the Korteweg de vries equation using cubic B-spline shape functions," in *Proceedings of the International Conference on Modelling and Simulation*, vol. 1, AMSE Conferance Press, Tassin-la-Demi-Lune, France, 1988.

[11] L. R. Gardner, G. A. Gardner, and A. H. Ali, "Simulations of solitons using quadratic spline finite elements," *Computer Methods in Applied Mechanics and Engineering*, vol. 92, no. 2, pp. 231–243, 1991.

[12] A. A. Soliman, "Collocation solution of the Korteweg-de Vries equation using septic splines," *International Journal of Computer Mathematics*, vol. 81, no. 3, pp. 325–331, 2004.

[13] E. N. Aksan and A. Özdeş, "Numerical solution of Korteweg-de Vries equation by Galerkin B-spline finite element method," *Applied Mathematics and Computation*, vol. 175, no. 2, pp. 1256–1265, 2006.

[14] S. I. Zaki, "A quintic B-spline finite element scheme for the KdVB equation," *Computational and Applied Mathematics*, vol. 190, pp. 532–547, 2006.

[15] P. C. Jain, R. Shankar, and D. Bhardwaj, "Numerical solution of the Korteweg-de Vries (KDV) equation," *Chaos, Solitons and Fractals*, vol. 8, no. 6, pp. 943–951, 1997.

[16] G. Micula and M. Micula, "On the numerical approach of Korteweg-de Vries-Burger equations by spline finite element and collocation methods," *Seminar on Fixed Point Theory Cluj-Napoca*, vol. 3, pp. 261–270, 2002.

[17] A. Canıvar, M. Sari, and I. Dag, "A Taylor-Galerkin finite element method for the KdV equation using cubic B-splines," *Physica B: Condensed Matter*, vol. 405, no. 16, pp. 3376–3383, 2010.

[18] S. Kapoor, S. Rawat, and S. Dhawan, "Numerical investigation of separated solitary waves solution for KdV equation through finite element technique," *International Journal of Computer Applications*, vol. 40, no. 14, pp. 27–33, 2012.

[19] N. K. Amein and M. A. Ramadan, "A small time solutions for the KdV equation using Bubnov-Galerkin finite element method," *Journal of the Egyptian Mathematical Society*, vol. 19, no. 3, pp. 118–125, 2011.

[20] B. Saka and I. Dag, "Quartic B-spline GALerkin approach to the numerical solution of the KdVB equation," *Applied Mathematics and Computation*, vol. 215, no. 2, pp. 746–758, 2009.

[21] R. Yu, R. Wang, and C. Zhu, "A numerical method for solving KdV equation with blended b-spline quasi-interpolation," *Journal of Information and Computational Science*, vol. 10, no. 16, pp. 5093–5101, 2013.

[22] R.-G. Yu, R.-H. Wang, and C.-G. Zhu, "A numerical method for solving KdV equation with multilevel B-spline quasi-interpolation," *Applicable Analysis*, vol. 92, no. 8, pp. 1682–1690, 2013.

[23] S. I. Zaki, "A quintic B-spline finite elements scheme for the KdVB equation," *Computer Methods in Applied Mechanics and Engineering*, vol. 188, no. 1, pp. 121–134, 2000.

[24] B. Saka, "Cosine expansion-based differential quadrature method for numerical solution of the KdV equation," *Chaos, Solitons and Fractals*, vol. 40, no. 5, pp. 2181–2190, 2009.

[25] Dağ and Y. Dereli, "Numerical solutions of KdV equation using radial basis functions," *Applied Mathematical Modelling*, vol. 32, no. 4, pp. 535–546, 2008.