World Academy of Science, Engineering and Technology
International Journal of Mathematical and Computational Sciences
Vol:5, No:4, 2011

Solving one-dimensional hyperbolic telegraph equation using cubic B-spline quasi-interpolation

Marzieh Dosti and Alireza Nazemi

Abstract—In this paper, the telegraph equation is solved numerically by cubic B-spline quasi-interpolation .We obtain the numerical scheme, by using the derivative of the quasi-interpolation to approximate the spatial derivative of the dependent variable and a low order forward difference to approximate the temporal derivative of the dependent variable. The advantage of the resulting scheme is that the algorithm is very simple so it is very easy to implement. The results of numerical experiments are presented, and are compared with analytical solutions by calculating errors L_2 and L_∞ norms to confirm the good accuracy of the presented scheme.

Keywords—Cubic B-spline, quasi-interpolation, collocation method, second-order hyperbolic telegraph equation.

I. Introduction

We consider the second-order linear hyperbolic telegraph equation in one-space dimension, given by

$$\frac{\partial^2 u}{\partial t^2} + 2\alpha \frac{\partial u}{\partial t} + \beta^2 u = \frac{\partial^2 u}{\partial x^2} + f(x,t), \ a \le x \le b, \ t \ge 0, \quad (1)$$

subject to initial conditions

$$u(x,0) = f_0(x), \quad a \le x \le b, \quad (2)$$

$$\frac{\partial u(x,0)}{\partial t} = f_1(x), \quad a \le x \le b, \quad (3)$$

and Dirichlet boundary conditions

$$u(a,t) = g_0(t), \quad u(b,t) = g_1(t), \quad t \ge 0, \quad (4)$$

where α and β are known constant coefficients. We assume that $f_0(x)$, $f_1(x)$ and their derivatives are continuous functions of x, and $g_i(t)$, $i = 0, 1$, and their derivatives are continuous functions of t. Both the electric voltage and the current in a double conductor, satisfy the telegraph equation, where x is distance and t is time. For $\alpha > 0$, $\beta = 0$ Eq. (1) represents a damped wave equation and for $\alpha > \beta > 0$, it is called telegraph equation.

The hyperbolic partial differential equations model the vibrations of structures (e.g. buildings, beams and machines) and are the basis for fundamental equations of atomic physics. Equations of the form Eq. (1) arise in the study of propagation of electrical signals in a cable of transmission line and wave phenomena. Interaction between convection and diffusion or reciprocal action of reaction and diffusion describes a number of nonlinear phenomena in physical, chemical and biological process [1]-[4]. In fact the telegraph equation is more suitable than ordinary diffusion equation in modeling reaction diffusion for such branches of sciences. For example biologists encounter these equations in the study of pulsate blood flow in arteries and in one- dimensional random motion of bugs along a hedge [5]. Also the propagation of acoustic waves in Darcy-type porous media [6], and parallel flows of viscous Maxwell fluids [7] are just some of the phenomena governed [8]-[9] by Eq. (1).

Marzieh Dosti and Alireza Nazemi (Corresponding author) are with the Department of Mathematics, School of Mathematical Sciences, Shahrood University of Technology, P. O. Box 3619995161-316, Tel-Fax No:+98273-3392012, Shahrood, Iran. e-mail: nazemi20042003@yahoo.com, e-mail: dostimarzieh@gmail.com.

B-spline functions have some attractive properties. Due to the being piecewise polynomial, they can be integrated and differentiated easily. Since they have compact support, numerical methods in which B-spline functions are used as a basis function [10]-[14] lead to matrix systems including band matrices. Such systems have solution algorithms with low computational cost. Therefore spline solutions of partial differential equations are suggested in many studies. For instance see [15]-[25].

In this paper, we provide a numerical scheme to solve hyperbolic telegraph equation equation using the derivative of the cubic B-spline quasi-interpolation to approximate the spatial derivative of the differential equations and employ a first order accurate forward difference for the approach of the temporal derivative such as [26], [27] shown. Then we do not require solving any linear system of equation so that we do not meet the question of the ill-condition of the matrix. Therefore, we can save the computational time and decrease the numerical error.

The remainder of paper is organized as follows. In Section 2, the univariate B-spline quasi-interpolants were introduced. In Section 3, we present the numerical techniques using cubic B-spline interpolation to solve telegraph equation. To demonstrate the efficiency of the proposed method, numerical experiments are carried out for several test problems and results are given in section 4. Finally, some conclusions are drawn in Section 5. Note that we have computed the numerical results by Matlab programming.

II. Univariate B-spline quasi-interpolants

For $I = [a, b]$, we denote by $S_d(X_n)$ the space of univariate splines of degree d and C^{d-1} on the uniform partition $X_n = \{x_i = a + ih, i = 0, \ldots, n\}$ with the meshlength $h = \frac{b-a}{n}$, where $b = x_n$. Let the B-spline basis of $S_d(X_n)$ be $\{B_j; j \in J\}$ with $J = \{1, 2, ..., n + d\}$, which can be computed by the de Boor-Cox formula [28].

Using the de Boor-Cox formula [28], for $j \in J$, B_j can be computed as

$$B_j(x)=\begin{cases}\frac{(x-x_{\,})^3}{(x_{\,+1}-x_{\,})(x_{\,+2}-x_{\,})(x_{\,+3}-x_{\,})}, & x\in[x_j,x_{j+1}),\\ \frac{(x-x_{\,})^2(x_{\,+2}-x)}{(x_{\,+2}-x_{\,})(x_{\,+2}-x_{\,+1})(x_{\,+3}-x_{\,})} \\ \frac{(x-x_{\,})(x_{\,+3}-x)(x-x_{\,+1})}{(x_{\,+3}-x_{\,})(x_{\,+3}-x_{\,+1})(x_{\,+2}-x_{\,+1})} \\ +\frac{(x_{\,+4}-x)(x-x_{\,+1})^2}{(x_{\,+2}-x_{\,+1})(x_{\,+3}-x_{\,+1})(x_{\,+4}-x_{\,+1})}, & x\in[x_{j+1},x_{j+2}),\\ \frac{(x-x_{\,})(x_{\,+3}-x)^2}{(x_{\,+3}-x_{\,})(x_{\,+3}-x_{\,+1})(x_{\,+3}-x_{\,+2})} \\ +\frac{(x-x_{\,+1})(x_{\,+4}-x)(x_{\,+3}-x)}{(x_{\,+3}-x_{\,+1})(x_{\,+3}-x_{\,+2})(x_{\,+4}-x_{\,+1})} \\ +\frac{(x_{\,+4}-x)^2(x-x_{\,+2})}{(x_{\,+4}-x_{\,+1})(x_{\,+4}-x_{\,+2})(x_{\,+3}-x_{\,+2})}, & x\in[x_{j+2},x_{j+3}),\\ \frac{(x_{\,+4}-x)^3}{(x_{\,+4}-x_{\,+1})(x_{\,+4}-x_{\,+2})(x_{\,+4}-x_{\,+3})}, & x\in[x_{j+3},x_{j+4}),\\ 0, & \text{otherwise.}\end{cases}$$

With these notations, the support of B_j is $supp(B_j) = [X_{j-d-1}, X_j]$. As usual, we add multiple knots at the endpoints: $a = X_{-d} = X_{-d+1} = ... = X_0$ and $b = X_n = X_{n+1} = ... = X_{n+d}$.

In [29]-[30], univariate B-spline quasi-interpolants (abbr. QIs) can be defined as operators of the form

$$Q_d f = \sum_{j\in J} \mu_j B_j. \tag{5}$$

We denote by Π_d the space of polynomials of total degree at most d. In general, we impose that Q_d is exact on the space Π_d, i.e. $Q_d p = p$ for all $p \in \Pi_d$. As a consequence of this property, the approximation order of Q_d is $O(h^{d+1})$ on smooth functions. In this paper, the coefficient μ_j is a linear combination of discrete values of f at some points in the neighborhood of $supp(B_j)$ as introduced in [29]-[30].

The main advantage of QIs is that they have a direct construction without solving any system of linear equations. Especially, it's very simple and effective for numerical integration and differentiation. Moreover, they are local, in the sense that the value of $Q_d f(x)$ depends only on values of f in a neighborhood of x. Finally, they have a rather small infinity norm, so they are nearly optimal approximates [30]. Since the cubic spline has become the most commonly used spline, we use cubic B-spline quasi-interpolation in this paper.

Let $y_i = f(x_i)$, $i = 0, 1, ..., n$. For the cubic B-spline QI

$$Q_3 f = \sum_{j=1}^{n+3} \mu_j(f) B_j, \tag{6}$$

the coefficients are listed as follows:

$$\begin{cases}\mu_1(f) = f_0,\\ \mu_2(f) = \frac{1}{18}(7f_0 + 18f_1 - 9f_2 + 2f_3),\\ \mu_j(f) = \frac{1}{6}(-f_{j-3} + 8f_{j-2} - f_{j-1}),\ j = 3, ..., n+1,\\ \mu_{n+2}(f) = \frac{1}{18}(2f_{n-3} - 9f_{n-2} + 18f_{n-1} + 7f_n),\\ \mu_{n+3}(f) = f_n.\end{cases}$$

For $f \in C^4(I)$, we have the error estimate [30]

$$\|f - Q_3 f\|_{\infty,I} \ \leq \frac{8}{3} d_{\infty,I}\ (f, \Pi_3) \ \text{ for } \ 1 \leq k \leq n,$$

thus

$$\|f - Q_3 f\|_\infty = O(h^4). \tag{7}$$

Differentiating interpolation polynomials leads to classical finite differences for the approximate computation of derivatives. Therefore, it seems natural to approximate derivatives of f by derivatives of $Q_3 f(x)$ up to the order h^3. We can evaluate the value of f at x_i by $(Q_3 f)^{'} = \sum_{j=1}^{n+3} \mu_j(f) B_j^{'}$, and $(Q_3 f)^{''} = \sum_{j=1}^{n+3} \mu_j(f) B_j^{''}$. For $j \in J$, we can compute $B_j^{'}$ and $B_j^{''}$ by the formula of B-spine's derivatives [28] as follows:

$$B_j^{'}(x)=\begin{cases}\frac{3(x-x_{\,})^2}{(x_{\,+1}-x_{\,})(x_{\,+2}-x_{\,})(x_{\,+3}-x_{\,})}, & x\in[x_j,x_{j+1}),\\ \frac{2(x-x_{\,})(x_{\,+2}-x)-(x-x_{\,})^2}{(x_{\,+2}-x_{\,})(x_{\,+2}-x_{\,+1})(x_{\,+3}-x_{\,})} \\ +\frac{(x-x_{\,})(x_{\,+3}-x)+(x-x_{\,+1})(x_{\,+3}+x_{\,}-2x)}{(x_{\,+3}-x_{\,})(x_{\,+3}-x_{\,+1})(x_{\,+2}-x_{\,+1})} \\ +\frac{-(x-x_{\,+1})^2+2(x_{\,+4}-x)(x-x_{\,+1})}{(x_{\,+2}-x_{\,+1})(x_{\,+3}-x_{\,+1})(x_{\,+4}-x_{\,+1})}, & x\in[x_{j+1},x_{j+2}),\\ \frac{(x_{\,+3}-x)^2-2(x-x_{\,})(x_{\,+3}-x)}{(x_{\,+3}-x_{\,})(x_{\,+3}-x_{\,+1})(x_{\,+3}-x_{\,+2})} \\ +\frac{(x_{\,+3}-2x+x_{\,+1})(x_{\,+4}-x)-(x-x_{\,+1})(x_{\,+3}-x)}{(x_{\,+3}-x_{\,+1})(x_{\,+3}-x_{\,+2})(x_{\,+4}-x_{\,+1})} \\ +\frac{(x_{\,+4}-x)^2-2(x-x_{\,+2})(x_{\,+4}-x)}{(x_{\,+4}-x_{\,+1})(x_{\,+4}-x_{\,+2})(x_{\,+3}-x_{\,+2})}, & x\in[x_{j+2},x_{j+3}),\\ \frac{(-3x_{\,+4}-x)^2}{(x_{\,+4}-x_{\,+1})(x_{\,+4}-x_{\,+2})(x_{\,+4}-x_{\,+3})}, & x\in[x_{j+3},x_{j+4}),\\ 0, & \text{otherwise,}\end{cases}$$

and

$$B_j^{''}(x)=\begin{cases}\frac{6(x-x_{\,})}{(x_{\,+1}-x_{\,})(x_{\,+2}-x_{\,})(x_{\,+3}-x_{\,})}, & x\in[x_j,x_{j+1}),\\ \frac{2(x_{\,+2}+2x_{\,}-3x)}{(x_{\,+2}-x_{\,})(x_{\,+2}-x_{\,+1})(x_{\,+3}-x_{\,})} \\ +\frac{2(x_{\,+3}+x_{\,+1}+x_{\,}-3x)}{(x_{\,+3}-x_{\,})(x_{\,+3}-x_{\,+1})(x_{\,+2}-x_{\,+1})} \\ +\frac{2(x_{\,+4}+2x_{\,+1}-3x)}{(x_{\,+2}-x_{\,+1})(x_{\,+3}-x_{\,+1})(x_{\,+4}-x_{\,+1})}, & x\in[x_{j+1},x_{j+2}),\\ \frac{-4x_{\,+3}-2x_{\,}+6x}{(x_{\,+3}-x_{\,})(x_{\,+3}-x_{\,+1})(x_{\,+3}-x_{\,+2})} \\ +\frac{-2x_{\,+4}-2x_{\,+3}-2x_{\,+1}+6x}{(x_{\,+3}-x_{\,+1})(x_{\,+3}-x_{\,+2})(x_{\,+4}-x_{\,+1})} \\ +\frac{-4x_{\,+4}-2x_{\,+2}+6x}{(x_{\,+4}-x_{\,+1})(x_{\,+4}-x_{\,+2})(x_{\,+3}-x_{\,+2})}, & x\in[x_{j+2},x_{j+3}),\\ \frac{6(x_{\,+4}-x)}{(x_{\,+4}-x_{\,+1})(x_{\,+4}-x_{\,+2})(x_{\,+4}-x_{\,+3})}, & x\in[x_{j+3},x_{j+4}),\\ 0, & \text{otherwise.}\end{cases}$$

Then we obtain the differential formulas for cubic B-spline QI as

$$(Q_3 f)^{'} = \sum_{j=1}^{n+3} \mu_j(f) B_j^{'}, \quad (Q_3 f)^{''} = \sum_{j=1}^{n+3} \mu_j(f) B_j^{''}. \tag{8}$$

III. Numerical scheme using cubic B-spline quasi-interpolant

In this section, we give the numerical scheme for solving telegraph equation (1) based on the cubic B-spline quasi-interpolant.

Discretizing telegraph equation

$$U_{tt} + 2\alpha U_t + \beta^2 U = U_{xx} + f(x, t), \tag{9}$$

in time with meshlength Δt, we get

$$\frac{U_j^{k+1} - 2U_j^k + U_j^{k-1}}{(\Delta t)^2} + 2\alpha \frac{U_j^{k+1} - U_j^k}{\Delta t} + \beta^2 U_j^k =$$

$$(U_{xx})_j^k + f(x_j, t_k). \tag{10}$$

TABLE I
RESULTS WITH $\Delta t = 0.001$ AND $\Delta x = 0.005$ IN EXAMPLE 4.1.

t	$t = 0.2$	$t = 0.4$	$t = 0.6$	$t = 0.8$	$t = 1$
L_∞	1.8918×10^{-4}	3.9943×10^{-4}	7.9715×10^{-4}	1.8799×10^{-3}	8.0113×10^{-3}
L_2	1.2645×10^{-8}	6.2552×10^{-8}	2.3523×10^{-7}	1.0732×10^{-6}	1.771×10^{-5}

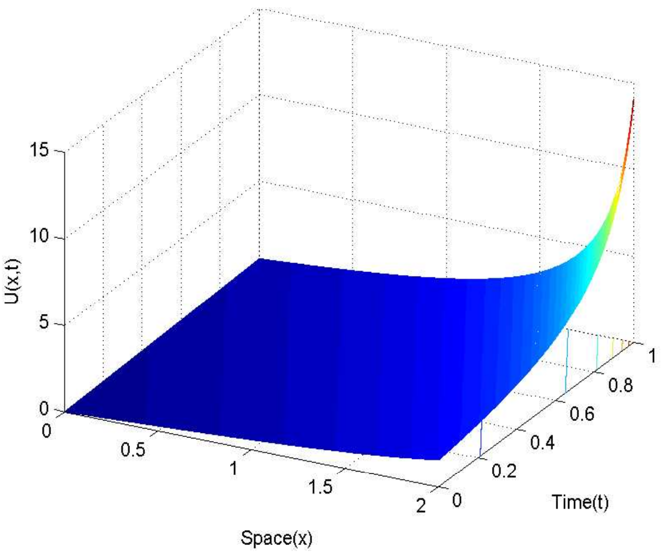

Fig. 1. Three-dimensional plot, with $\Delta t = 0.001$ and $\Delta x = 0.005$ in Example 4.1.

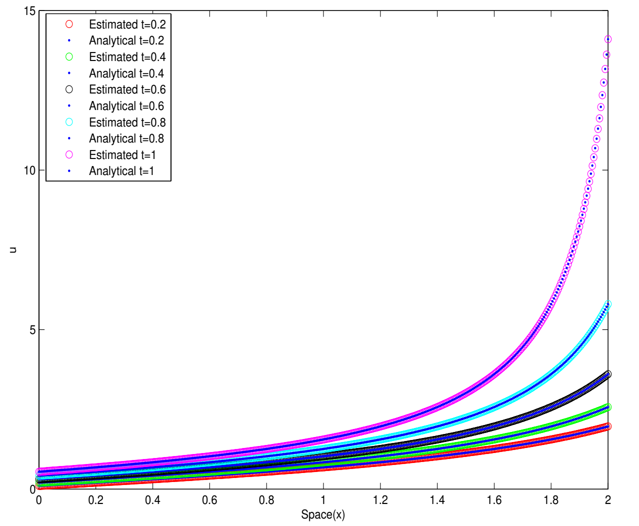

Fig. 2. Comparisons between numerical and analytical solutions of Eq. (1) in $t = 0.2s, t = 0.4s, t = 0.6s, t = 0.8s, t = 1s$, with $\Delta t = 0.001$ and $\Delta x = 0.005$ for Example 4.1.

We can obtain

$$(1 + 2\alpha\Delta t)U_j^{k+1} = (2 + 2\alpha\Delta t - \beta^2(\Delta t)^2)U_j^k - U_j^{k-1} +$$

$$(\Delta t)^2(U_{xx})_j^k + (\Delta t)^2 f(x_j, t_k), \qquad (11)$$

where $U_j^k \approx U(x_j, t_k)$. Then, we use the derivatives of the cubic B-spline quasi-interpolant $Q_3U(x_j, t_k)$ to approximate $(U_{xx})_j^k$. From the initial conditions and boundary conditions (2)-(4), we can compute the numerical solution of telegraph Eq. (1) step by step using the B-spline quasi-interpolation (BSQI for short) scheme (15) and formulas (12).

IV. NUMERICAL EXAMPLES

In this section, some numerical solutions of the telegraph equation in form Eq.(1) with the initial conditions (2) and (3) and boundary conditions (4) with the BSQI scheme (15) are presented. To show the efficiency of the present method for our problem in comparison with the exact solution, we report the root mean square error L_2 and maximum error L_∞ errors:

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

$$L_\infty = |U - U_N|_\infty = \max_j |U_j - (U_N)_j|.$$

Example 4.1: In this example, we consider the hyperbolic telegraph Eq. (1) with $\alpha = 10$, $\beta = 5$, $f(x,t) = \alpha(1 + \tan^2(\frac{x+t}{2})) + \beta^2 \tan(\frac{x+t}{2})$ and $0 \le x \le 2$. The initial conditions are given by

$$\begin{cases} u(x,0) = \tan(\frac{x}{2}), \\ u_t(x,0) = \frac{1}{2}(1 + \tan^2(\frac{x}{2})), \end{cases}$$

and the boundary conditions

$$\begin{cases} u(0,t) = \tan(\frac{t}{2}), \\ u(2,t) = \tan(\frac{2+t}{2}), \end{cases}$$

The exact solution of this example [31] is $u(x,t) = \tan((x+t)/2)$. The root-mean-square error L_2 and maximum error L_∞ are presented in Table 1. The space-time graph of the estimated solution up to $t = 1$ is shown in Figure 1. The graph of analytical and estimated solutions for some different times and $x \in [0,2]$ is presented in Figure 2. Absolute error between the numerical and analytical solution is also depicted at different time in Figure 3.

Example 4.2: Consider the hyperbolic telegraph Eq. (1) with $\alpha = 4$, $\beta = 2$, $f(x,t) = (2 - 2\alpha + \beta^2)\exp(-t)\sin(x)$ and $0 \le x \le \pi$. The initial conditions are given by

$$\begin{cases} u(x,0) = \sin(x), \\ u_t(x,0) = -\sin(x), \end{cases}$$

and the boundary conditions

$$u(0,t) = u(\pi,t) = 0, \qquad (12)$$

TABLE II
RESULTS WITH $\Delta t = 0.001$ AND $\Delta x = 0.02$ IN EXAMPLE 4.2.

t	$t = 0.5$	$t = 1$	$t = 1.5$	$t = 1.75$	$t = 2$
L_∞	1.0676×10^{-3}	7.1563×10^{-4}	4.8126×10^{-4}	3.5192×10^{-4}	2.8398×10^{-4}
L_2	2.8450×10^{-7}	2.8983×10^{-7}	2.5825×10^{-7}	2.0892×10^{-7}	1.5744×10^{-7}

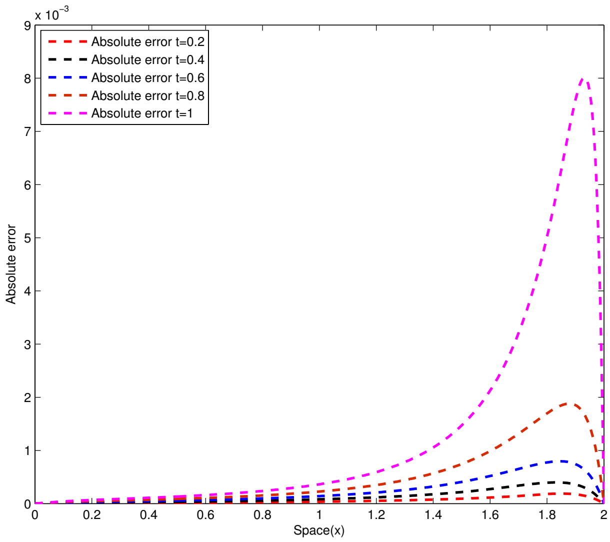

Fig. 3. Absolute error in $t = 0.2s, t = 0.4s, t = 0.6s, t = 0.8s, t = 1s$, with $\Delta t = 0.001$ and $\Delta x = 0.005$ for Example 4.1.

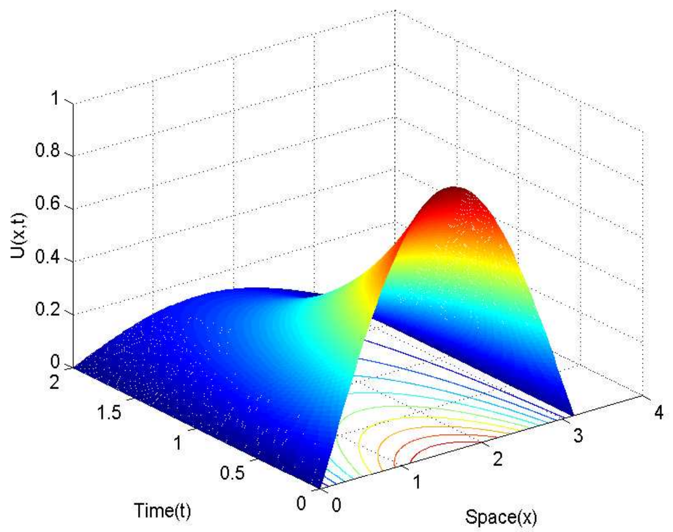

Fig. 4. Three-dimensional plot, with $\Delta t = 0.001$ and $\Delta x = 0.02$ in Example 4.2.

The exact solution of this example [31] is $u(x,t) = \exp(-t)\sin(x)$. The space-time graph of the numerical solution up to $t = 2$ is presented in Figure 4. The graph of analytical and estimated solutions for some different times and $x \in [0, \pi]$ is presented in Figure 5. The accuracy of the B-spline method is measured by using the L_2 and L_∞ errors. The errors are reported in Table 2. Absolute error between the numerical and analytical solution is also depicted at different time in Figure 6.

Example 4.3: Consider Eq. (1) with $\alpha = 6$, $\beta = 2$, $0 \leq x \leq 1$ and the following conditions:

$$\begin{cases} f_0(x) = \sin(x), \\ f_1(x) = 0 \\ g_0(t) = 0, \\ g_1(t) = \cos(t)\sin(1), \\ f(x,t) = -2\alpha\sin(t)\sin(x) + \beta^2\cos(t)\sin(x). \end{cases}$$

The exact solution of this example [31] is $u(x,t) = \cos(t)\sin(x)$. The root-mean-square error and and maximum error are presented in Table 3, also the space-time graph of the estimated solution up to $t = 1$ is presented in Figure 7. The graph of analytical and estimated solutions for some different times and $x \in [0, 1]$ is presented in Figure 8. Absolute error between the numerical and analytical solution is also depicted at different time in Figure 9.

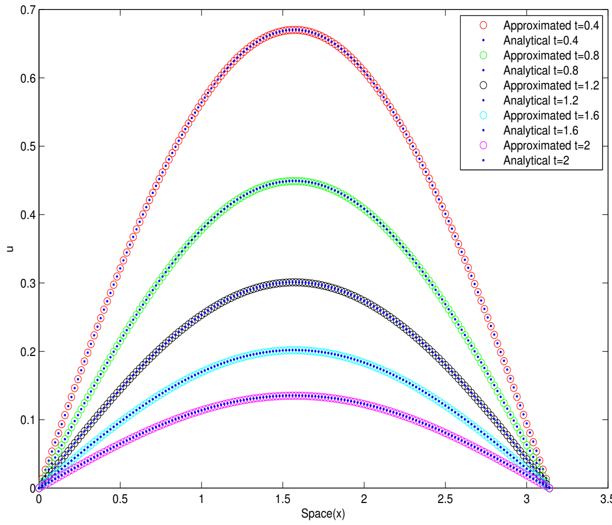

Fig. 5. Comparisons between numerical and analytical solutions of Eq. (1) in $t = 0.4s, t = 0.8s, t = 1.2s, t = 1.6s, t = 2s$, with $\Delta t = 0.001$ and $\Delta x = 0.02$ for Example 4.2.

TABLE III
RESULTS WITH $\Delta t = 0.0005$ AND $\Delta x = 0.002$ IN EXAMPLE 4.3.

t	$t = 0.2$	$t = 0.4$	$t = 0.6$	$t = 0.8$	$t = 1$
L_∞	3.5005×10^{-5}	5.576×10^{-5}	6.9334×10^{-4}	7.686×10^{-5}	7.8908×10^{-5}
L_2	4.8691×10^{-10}	1.4168×10^{-9}	2.3128×10^{-9}	2.9199×10^{-9}	3.1223×10^{-9}

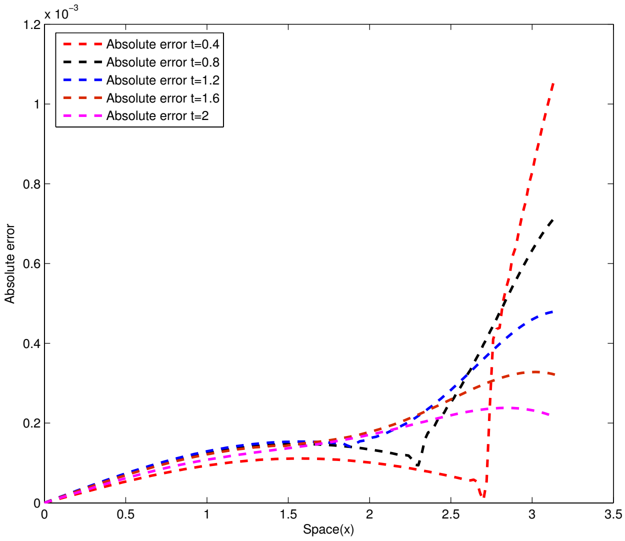

Fig. 6. Absolute error in $t = 0.4s, t = 0.8s, t = 1.2s, t = 1.6s, t = 2s$, with $\Delta t = 0.001$ and $\Delta x = 0.02$ for Example 4.2.

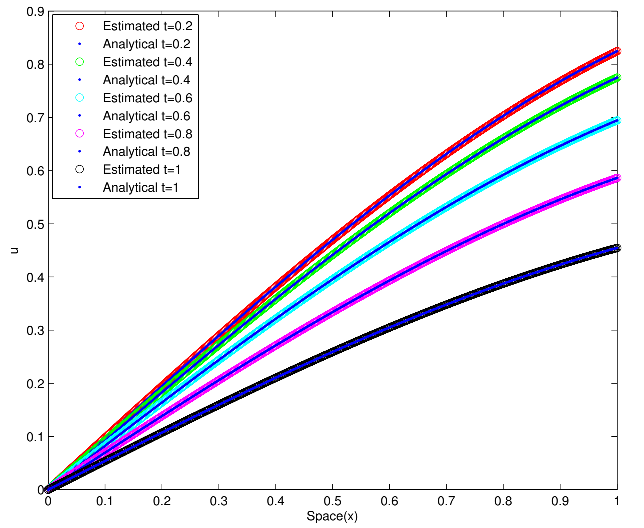

Fig. 8. Comparisons between numerical and analytical solutions of Eq. (1) in $t = 0.2s, t = 0.4s, t = 0.6s, t = 0.8s, t = 1s$, with $\Delta t = 0.0005$ and $\Delta x = 0.002$ for Example 4.3.

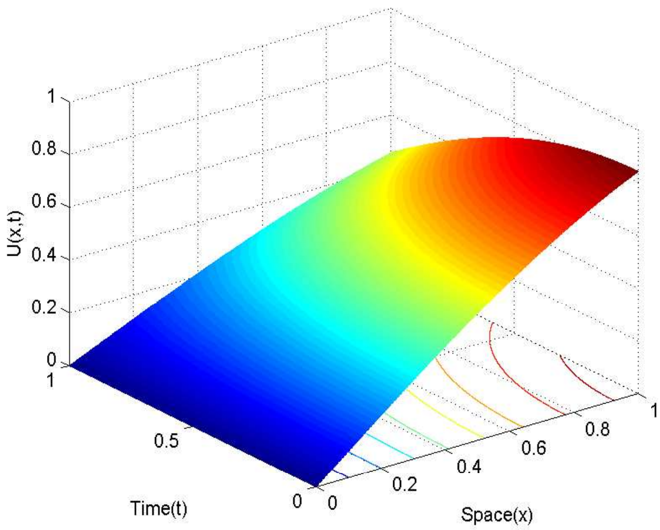

Fig. 7. Three-dimensional plot, with $\Delta t = 0.0005$ and $\Delta x = 0.002$ in Example 4.3.

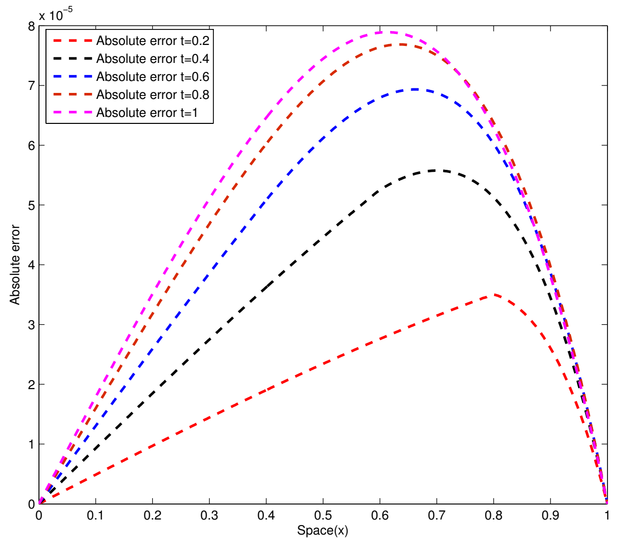

Fig. 9. Absolute error in $t = 0.2s, t = 0.4s, t = 0.6s, t = 0.8s, t = 1s$, with $\Delta t = 0.0005$ and $\Delta x = 0.002$ in Example 4.3.

V. Conclusion

In this paper, a numerical treatment for the second-order hyperbolic telegraph equation is proposed using cubic B-spline quasi-interpolation.From the numerical results, we can say that the BSQI scheme is feasible and the error is acceptable. The numerical solutions are compared with the exact solution by finding L_2 and L_∞ errors.The implementation of the present method is a very easy, acceptable, and valid.

References

[1] D. M. Pozar, *Microwave engineering*, NewYork: Addison-Wesley, 1990.
[2] A. Mohebbi, M. Dehghan, High order compact solution of the one-space-dimensional linear hyperbolic equation, Numerical Methods for Partial Differential Equations, 24 (2008) 1222–1235.
[3] A. Jeffrey, *Advanced engineering mathematics*, Harcourt Academic Press, 2002.
[4] A. Jeffrey, *Applied partial differential equations*, NewYork: Academic Press, 2002.
[5] R. K. Mohanty, New unconditionally stable difference schemes for the solution of multi-dimensional telegraphic equations, Computer Mathematics, 86 (2008) 2061–2071.
[6] H. Pascal, Pressure wave propagation in a fluid flowing through a porous medium and problems related to interpretation of Stoneley's wave attenuation in acoustical well logging, Engineering Science, 24 (1986) 1553–1570.
[7] G. Bohme, *Non-Newtonian fluid mechanics*, NewYork: North-Holland, 1987.
[8] D. J. Evans, H. Bulut, Thenumerical solution of thetelegraph equation by the alternating group explicit method, Computer Mathematics 80 (2003) 1289– 1297.
[9] P. M. Jordan, M. R. Meyer, A. Puri, Causal implications of viscous damping in compressible fluid flows, Physics Review, 62 (2000) 7918–7926.
[10] L. L. Schumaker, *Spline Functions: Basic Theory*, Krieger Publishing Company, Florida, 1981.
[11] J. M. Ahlberg, E. N. Nilson, J. L. Walsh, *The Theory of Splines and Their Applications*, Academic Press, NewYork, 1967.
[12] P. M. Prenter, *Splines and variational methods*, New York: John Wiley, 1975.
[13] C. De Boor, *A Practical Guide to Splines*, Springer-Verlag, New York, 1978.
[14] G. Micula, *Handbook of Splines*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1999.
[15] B. Saka, I. Da$\breve{G}$, Quartic B-spline Galerkin approach to the numerical solution of the KdVB equation, Applied Mathematics and Computation, 215 (2009) 746–758.
[16] I. Da$\breve{G}$, A. Canvar, A. Sahin, TaylorGalerkin and Taylor-collocation methods for the numerical solutions of Burgers equation using B-splines, communication in nonlinear science and numerical simulation, 16 (2010) 2696-2708.
[17] S. Chandra Sekhara Rao, M. Kumar, Exponential B-spline collocation method for self-adjoint singularly perturbed boundary value problems, Applied Numerical Mathematics, 58 (2008) 1572-1581.
[18] M. K. Kadalbajoo, P. Arora, B-spline collocation method for the singular-perturbation problem using artificial viscosity, Computers and Mathematics with Applications, 57 (2009) 650–663.
[19] M. K. Kadalbajoo, V. Gupta, A. Awasthi, A uniformly convergent B-spline collocation method on a nonuniform mesh for singularly perturbed one-dimensional time-dependent linear convectiondiffusion problem, Journal of Computational and Applied Mathematics, 220 (2008) 271 - 289.
[20] M. K. Kadalbajoo, V. Gupta, Numerical solution of singularly perturbed convectiondiffusion problem using parameter uniform B-spline collocation method, Journal of Mathematical Analysis and Applications, 355 (2009) 439-452.
[21] B. Saka, I. Da$\breve{G}$, Quartic B-spline collocation method to the numerical solutions of the Burgers' equation, Chaos, Solitons & Fractals, 32 (2007) 1125–1137.
[22] F. Gao, C. M Chi, Solving third-order obstacle problems with quartic B-splines, Applied Mathematics and Computation, 180 (2006) 270–274.
[23] K. R. Raslan, Collocation method using quartic B-spline for the equal width (EW) equation, Applied Mathematics and Computation, 168 (2005) 795–805.
[24] F. i. Haq, S. u. Islam, I. A. Tirmizi, A numerical technique for solution of the MRLW equation using quartic B-splines, Applied Mathematical Modelling, 34 (2010) 4151–4160.
[25] M. Eck, D. Lasser, B-spline-B$\acute{e}$zier representation of geometric spline curves: Quartics and quintics, Computers & Mathematics with Applications, 23 (1992) 23–39.
[26] C. G. Zhu, R. H. Wang, Numerical solution of Burger's equation by cubic B-spline quasi-interpolation, Applied Mathematics and Computation, 208 (2009) 260-272.
[27] C. G. Zhu, W. S. Kang, Numerical solution of Burgers-Fisher equation by cubic B-spline quasi-interpolation, Applied Mathematics and Computation, 216 (2010) 2679-2686.
[28] G. Farin, *Curves and Surfaces for CAGD*, fifth ed., Morgan Kaufman, San Francisco, 2001.
[29] P. Sablonni$\grave{e}$re, Quasi-interpolants splines sobre particiones uniforms, in: First Meeting in Approximation Theory of the University of Ja$\grave{e}$n, Ubeda, June 29- July 2, 2000, Pr$\grave{e}$publication IRMAR 00-38, Rennes, June 2000.
[30] P. Sablonni$\grave{e}$re, Univariate spline quasi-interpolants and applications to numerical analysis, Rend. Sem. Mat. Univ. Pol. Torino 63 (2005) 211-222.
[31] M. Dehghan, A. Ghesmati, Solution ofthesecond-orderone-dimensional hyperbolic telegraph equation by using the dual reciprocity boundary integral equation (DRBIE) method, Engineering Analysis with Boundary Elements, 34 (2010) 51–59.