Journal of Interdisciplinary Mathematics
ISSN: 0972-0502 (Print), ISSN: 2169-012X (Online)
Vol. 28 (2025), No. 7, pp. 2413–2437
DOI : 10.47974/JIM-1898

Numerical investigation of second order one dimensional telegraph equation using subdivision schemes

Syeda Tehmina Ejaz *
Safia Malik †
Department of Mathematics
The Government Sadiq College Women University
Bahawalpur 63100
Pakistan

Ali Akgül §
Department of Mathematics
Siirt University
56100 Siirt
Turkey

and

Department of Computer Science and Mathematics
Lebanese American University
Beirut
Lebanon

and

Department of Mathematics
Near East University
Mathematics Research Center
Near East Boulevard
PC: 99138, Nicosia /Mersin 10
Turkey

---

* *E-mail:* syedatehmina.ejaz@gscwu.edu.pk
† *E-mail:* safiamalik521@gmail.com
§ *E-mail:* aliakgul00727@gmail.com

Murad Khan Hassani ‡
Department of Mathematics
Ghazni University
Ghazni 2301
Afganistan

Abstract

This article presents a novel approach for solving the one-dimensional second-order telegraph equation along with Dirichlet boundary conditions. The method is based on the subdivision collocation technique, which discretizes the time derivative using a typical finite difference approach. The method employs the fundamental functions of the subdivision scheme as interpolating functions in space. To assess the scheme's stability, the von Neumann (Fourier) method is utilized. The authors provide various test scenarios to demonstrate the accuracy of the new approach and the effectiveness of the subdivision collocation technique. The numerical results align well with previously established exact solutions and research. The proposed method offers numerous advantages, including computational efficiency, applicability, and precision. The article comprehensively discusses the procedure and presents results demonstrating its efficacy. This paper makes a significant contribution to the field of numerical approaches for telegraph equations.

Subject Classification: *65D17, 35Lxx, 65L60, 65N12.*

Keywords: *Subdivision schemes, Telegraph equation, Subdivision collocation method, Stability.*

1. Introduction

In this paper, we are dealing with second order linear Telegraph equation (TE)

$$\frac{\partial^2 u(x,t)}{\partial t^2}+a\frac{\partial u(x,t)}{\partial t}+bu=c^2\frac{\partial^2 u(x,t)}{\partial x^2}+\varphi(x,t), \tag{1.1}$$

with initial conditions

$$u(x,0)=\varsigma_1(x), \qquad u_t(x,0)=\varsigma_2(x), \qquad \text{for} \quad 0\le x\le L, \tag{1.2}$$

and Drichlet boundary conditions

$$u(0,t)=\rho_1(t), \qquad u(L,t)=\rho_2(t), \qquad \text{for} \quad t>0, \tag{1.3}$$

where a, b, c are the known constants and $\varsigma_1,\varsigma_2,\rho_1$ and ρ_2 the continuous known functions also $\varphi(x,t)$ is a continuous function. Equation (1.1) describes damped wave equation if $a>0, b=0,$ and represents TE if

‡ *E-mail:* **mhassani@gu.edu.af**

$a > b > 0$. The TE (1.1), finds extensive application in the study of electric signal wave propagation in cable transmission lines and wave phenomena. Numerous scholars have tackled the TE using various numerical and analytical methods. For instance, the Spectral Galerkin method has been employed for solving equation (1.1) via generalized Lucas polynomials. Additionally, a cubic B-spline-based differential quadrature method was introduced in [1] and [2], respectively. Adaptive Monte Carlo methods were introduced in [3], while Laplace transform collocation, cubic trigonometric B-spline, and cubic B-spline collocation methods for solving equation (1.1) were presented by the authors in [4], [5], [6], [7] and [8], respectively. Dehghan and Shokria [9] solved TE numerically. They used collocation points and approximated the solution through thin-plate splines with a radial basis function. To address the solution of equation (1.1), Dehghan and Lakestani [10] devised a numerical method based on Chebyshev cardinal functions. Saadatmandi and Dehghan [11] introduced the Chebyshev Tau method, while Dehghan and Ghesmati [12] developed truly meshless local weak-strong (MLWS) methods. Lakestani and Saray [13] presented a numerical method for solving a second-order, 1D linear TE using interpolating scaling functions. Javidi and Nyamoradi [14] employed the well-known homotopy perturbation method (HPM) and Laplace transform techniques to tackle the one-dimensional linear hyperbolic telegraph equation. Inc et al. [15] proposed a method for solving partial differential equations known as the reproducing kernel method (RKM). Hashemi et al. [16] addressed the non-homogeneous 1D TE using a group preserving scheme. The differential transform method was introduced by the authors of [17], and the reduced differential transform method for solving equation (1.1) was presented by the authors of [18].

The primary objective of this paper is to solve the TE (1.1) with initial and boundary conditions (1.2) and (1.3) using the Subdivision Collocation Agorithm (SCA). Additionally, we aim to compare SCA with other existing methods. Subdivision-based techniques have previously been utilized by Qu and Agarwal [20, 21, 22] and Ejaz et al. [23, 24] for solving ordinary differential equations with constraints specified at the endpoints. However, subdivision-based algorithms have not been widely employed for solving partial differential equations. The main purpose of this paper is to explore the application of the subdivision scheme (SS) in direction of constructing numerical algorithm for solving the TE.

The paper is organized as follows: In section 2, we provide an overview of the fundamental characteristics of the 6-point binary approximation of the subdivision system. Additionally, we introduce the Subdivision Collocation Algorithm (SCA) for solving the given problem. Section 3 covers the derivation of the initial solution and includes a discussion of the suggested algorithm's convergence and error evaluation. In section 4, we present numerical results obtained using the suggested methodology, conduct a comparison with alternative methods, and section 5 provide a conclusion based on the results.

2. Subdivision Collocation Algorithm (SCA)

In this section, we present some fundamental characteristics of the 6-point subdivision scheme. Following that, we explain in detail the procedure known as the subdivision collocation algorithm (SCA) for solving (1.1).

2.1 *Fundamental characteristic of subdivision scheme (SS)*

The goal of this research is to solve TE using appropriate subdivision schemes, and the following binary subdivision scheme has been chosen for this purpose [19].

$$\begin{cases} R_{2i}^{k+1} = R_i^k, \\ R_{2i+1}^{k+1} = \frac{3}{256}(R_{i-2}^k + R_{i+3}^k) - \frac{25}{256}(R_{i-1}^k + R_{i+2}^k) + \frac{22}{256}(R_i^k + R_{i+1}^k). \end{cases} \tag{2.1}$$

The smoothness order, degree of generation, degree of reproduction, approximation order and support of scheme (2.1) are two, five, six and $(-5,5)$ respectively. The fundamental solution and the basis function are defined in equations (2.2) and (2.3) respectively.

$$\kappa(i) = \begin{cases} 1, & \text{if} \quad i = 0, \\ 0, & \text{if} \quad i \neq 0, \end{cases} \tag{2.2}$$

and if r_k is the mask of (2.1) then (2.2) satisfies the relation given below

$$\kappa(x) = \Sigma r_k \kappa(2x - k), \tag{2.3}$$

The first and second-order derivatives at integral support of (2.1) are computed using a similar process as described by [20]. All other derivative values vanish at the integral points other than the support.

$$\begin{cases} \kappa'(0)=0, \\ \kappa'(\pm 1)=\mp\frac{272}{365}, \\ \kappa'(\pm 2)=\mp\frac{53}{365}, \\ \kappa'(\pm 3)=\mp\frac{16}{1095}, \\ \kappa'(\pm 4)=\mp\frac{1}{2920}, \end{cases} \text{ and } \begin{cases} \kappa''(0)=-\frac{295}{28}, \\ \kappa'' \pm 1)=\frac{712}{105}, \\ \kappa''(\pm 2)=-\frac{184}{105}, \\ \kappa''(\pm 3)=\frac{8}{35}, \\ \kappa''(\pm 4)=\frac{3}{280}, \end{cases} \tag{2.4}$$

2.2 *Construction of SCA for Telegraph Equation*

Let the approximate solution of (1.1), with unknown collocation points $\mathcal{V}_s$ is given below

$$U(x,t)=\sum_{s=-4}^{N+4}\mathcal{V}_s\frac{S_{2,s}(x)}{\eta_{r-s}} \quad 0\le x\le L, \quad \text{where} \quad \eta_{r-s}=\begin{cases}1, & \text{if} \quad r\ne s, \\ \eta_0>1, & \text{if} \quad r=s\end{cases} \tag{2.5}$$

$$S_{2,s}=\kappa\left(\frac{x_r-x_s}{h}\right), \tag{2.6}$$

is the basis function of scheme (2.1). The constants $\mathcal{V}_s$'s are the time dependant unknowns. Let the approximate solution U_r^k at the point (x_r,t_k) over the interval $[x_r,x_{r+1}]$ can be simplified into

$$U_r^k=\sum_{s=r-4}^{r+4}\mathcal{V}_s\frac{S_{2,s}(x)}{\eta_{r-s}}, \quad r=0,1,2,\cdots,N. \tag{2.7}$$

The approximate values of $S_{2,s}$ and its derivative at the integer support are given in (2.2) and (2.4) respectively. The approximations of (1.1), at $t_{k+1}th$ time level can be considered by

$$(U_{tt})_r^k+a(U_t)_r^k+bU_r^k+(1-\theta)f_r^k+\theta f_r^{k+1}=\varphi(x_r,t), \tag{2.8}$$

where

$$f_r^k=-(U_{xx})_r^k. \tag{2.9}$$

Using forward difference schemes the time derivatives are discretized at consecutive time levels k and $k+1$ is expressed as

$$(U_t)_r^k=\frac{1}{\Delta t}[U_r^{k+1}-U_r^k]. \tag{2.10}$$

$$(U_{tt})_r^k=\frac{1}{\Delta t^2}[U_r^{k-1}-2U_r^k+U_r^{k+1}]. \tag{2.11}$$

Now using (2.10) and (2.11) in (2.8), we obtain

$$\frac{U_r^{k-1}-2U_r^k+U_r^{k+1}}{\Delta t^2}+a\left[\frac{U_r^{k+1}-U_r^k}{\Delta t}\right]+b\sum_{s=r-4}^{r+4}\mathcal{V}_s^k\frac{S_{2,s}(x_s)}{\eta_{r-s}}-\frac{(1-\theta)c^2}{h^2}\sum_{s=r-4}^{r+4}\mathcal{V}_s^k\frac{S''_{2,s}(x_s)}{\eta_{r-s}}-\frac{\theta c^2}{h^2}\sum_{s=r-4}^{r+4}\mathcal{V}_s^{k+1}\frac{S''_{2,s}(x_s)}{\eta_{r-s}} \tag{2.12}$$

$$=\varphi(x_r,t),\quad \text{where } r=0,1,2,3,\cdots,N,$$

where Δt is the step size of time system (2.12).

For $\theta=0.5$ equation (2.12) becomes

$$\frac{U_r^{k-1}-2U_r^k+U_r^{k+1}}{\Delta t^2}+a\left[\frac{U_r^{k+1}-U_r^k}{\Delta t}\right]+b\sum_{s=r-4}^{r+4}\mathcal{V}_s^k\frac{S_{2,s}(x_s)}{\eta_{r-s}}-\left(\frac{1}{2h^2}\right)c^2\sum_{s=r-4}^{r+4}\mathcal{V}_s^k\frac{S''_{2,s}(x_s)}{\eta_{r-s}}-\frac{1}{2h^2}c^2\sum_{s=r-4}^{r+4}\mathcal{V}_s^{k+1}\frac{S''_{2,s}(x_s)}{\eta_{r-s}} \tag{2.13}$$

$$=\varphi(x_r,t),\qquad \text{where } r=0,1,2,3,\cdots,N,$$

equation (2.13) can be simplified as

$$2h^2U_r^{k-1}-4h^2U_r^k+2h^2U_r^{k+1}+2a\Delta t h^2U_r^{k+1}-2a\Delta t h^2U_r^k+2b(\Delta t)^2h^2\sum_{s=r-4}^{r+4}\mathcal{V}_s^k\frac{S(r-s)}{\eta(r-s)}-$$

$$c^2(\Delta t)^2\sum_{s=r-4}^{r+4}\mathcal{V}_s^k\frac{S''(r-s)}{\eta(r-s)}+c^2(\Delta t)^2\sum_{s=r-4}^{r+4}\mathcal{V}_s^{k+1}\frac{S''(r-s)}{\eta(r-s)}=2h^2(\Delta t)^2\varphi(x_r,t),$$

where $r=0,1,2,3,\cdots,N,$

this implies

$$2h^2U_r^{k-1}+2h^2U_r^{k+1}+2a\Delta t h^2U_r^{k+1}-c^2(\Delta t)^2\sum_{s=r-4}^{r+4}\mathcal{V}_s^{k+1}\frac{S''(r-s)}{\eta_{r-s}}=4h^2U_r^k+2h^2a\Delta tU_r^k$$

$$+(\Delta t)^2\sum_{s=r-4}^{r+4}(-2bh^2\frac{S(r-s)}{\eta_{r-s}}+c^2\frac{S''(r-s)}{\eta_{r-s}})\mathcal{V}_s^k+2h^2(\Delta t)^2\varphi(x_r,t),$$

for $r=0,1,2,3,\cdots,N,$

hence at each level of time, we get linear system of $(N+1)$ equations,

$$2h^2(1+a\Delta t)U_r^{k+1}-c^2(\Delta t)^2\sum_{s=r-4}^{r+4}\mathcal{V}_s^{k+1}\frac{S''(r-s)}{\eta_{r-s}}=2h^2(2+a\Delta t)U_r^k+(\Delta t)^2$$

$$\sum_{s=r-4}^{r+4}(-2bh^2S(r-s)+c^2\frac{S''(r-s)}{\eta_{r-s}})\mathcal{V}_s^k+2h^2(\Delta t)^2\varphi(x_r,t),\ \text{for } r=0,1,2,3,\cdots,N, \tag{2.14}$$

with $(N+9)$ unknowns

$$V^{k+1}=(\mathcal{V}_{-4}^{k+1},\mathcal{V}_{-3}^{k+1},\mathcal{V}_{-2}^{k+1},\cdots,\mathcal{V}_{N+4}^{k+1})^T. \tag{2.15}$$

Theorem 2.1: *The matrix representation of the linear system of equations (2.14) is*

$$-AV^{k+1} = BV^k + D \tag{2.16}$$

or

$$V^{k+1} = EV^k + G \qquad \text{where} \quad E = -A^{-1}B, \quad \text{and} \quad G = -A^{-1}D, \tag{2.17}$$

in which the matrices A, B & G and the vectors V^{k+1} & V^k are define as

$$A = [\gamma^{(\mu\nu)}_{\nu-\mu-4}]_{N+1\times N+9}, \qquad B = [\Upsilon^{(\mu\nu)}_{\nu-\mu-4}]_{N+1\times N+9}, \tag{2.18}$$

with integers $1 \le \mu \le N+1$, $1 \le \nu \le N+9$ and $\gamma(\tau)$ & $\Upsilon(\tau)$ are defined as

$$\gamma(\tau) = \begin{cases} \kappa''_\tau, & \text{if} \quad \tau \in [-4,0) \quad \text{and} \quad (0,4] \\ 0, & \text{if} \quad \tau \notin [-4,4] \\ \frac{\kappa''_\tau}{\eta_0} - \frac{2h^2(1+a\Delta t)}{c^2\eta_0(\Delta t)^2} & \text{if} \quad \tau = 0 \end{cases} \tag{2.19}$$

$$\Upsilon(\tau) = \begin{cases} \kappa''_\tau, & \text{if} \quad \tau \in [-4,0) \quad \text{and} \quad (0,4] \\ 0, & \text{if} \quad \tau \notin [-4,4] \\ \frac{\kappa''_\tau}{\eta_0} + 2h^2\left(\frac{2+a\Delta t - b(\Delta t)^2}{c^2\eta_0(\Delta t)^2}\right), & \text{if} \quad \tau = 0 \end{cases} \tag{2.20}$$

$$V^{k+1} = (\mathcal{V}^{k+1}_{-4}, \cdots, \mathcal{V}^{k+1}_{N+4})^T, V^k = (\mathcal{V}^k_{-4}, \cdots, \mathcal{V}^k_{N+4})^T \,\&\, G = 2h^2(\Delta t)^2(\varphi(x_0,t), \cdots, \varphi(x_N,t))^T \tag{2.21}$$

Proof: Substituting $r = 0$ in equation (2.14), we have

$$2h^2(1+a\Delta t)U^{k+1}_0 - c^2(\Delta t)^2 \sum_{s=-4}^{4} \mathcal{V}^{k+1}_s \frac{S''(-s)}{\eta_{-s}} = 2h^2(2+a\Delta t)U^k_0 + (\Delta t)^2 \times$$
$$\sum_{s=-4}^{4}\left(-2bh^2 S(-s) + c^2 \frac{S''(-s)}{\eta_{-s}}\right)\mathcal{V}^k_s + 2h^2(\Delta t)^2\varphi(x_0,t).$$

From (2.7) we have

$$U^{k+1}_0 = \frac{\mathcal{V}^{k+1}_0}{\eta_0} \quad \text{and} \quad U^k_0 = \frac{\mathcal{V}^k_0}{\eta_0}, \tag{2.22}$$

using (2.22) in above expression and after simplification, we get

$$-c^2(\Delta t)^2\sum_{s=-4}^{-1}\mathcal{V}_s^{k+1}S''(-s)+(2h^2(1+a\Delta t)-c^2(\Delta t)^2S''(0))\frac{\mathcal{V}_0^{k+1}}{\eta_0}-c^2(\Delta t)^2\times$$
$$\sum_{s=1}^{4}c_s^{k+1}S''(-s)=(\Delta t)^2\sum_{s=-4}^{-1}(-2bh^2S(-s)+c^2S''(-s))\mathcal{V}_s^k+(2h^2(2+a\Delta t)+$$
$$(\Delta t)^2(-2bh^2S(0)+c^2S''(0)))\frac{\mathcal{V}_0^k}{\eta_0}+(\Delta t)^2\sum_{s=1}^{4}(-2bh^2S(-s)+c^2S''(-s))c_s^k+2h^2(\Delta t)^2\varphi(x_0,t),$$

using (2.2) and (2.6) in above expression, we get

$$-c^2(\Delta t)^2\sum_{s=-4}^{-1}\mathcal{V}_s^{k+1}\kappa''(-s)+(2h^2(1+a\Delta t)-c^2(\Delta t)^2\kappa''(0))\frac{\mathcal{V}_0^{k+1}}{\eta_0}-c^2(\Delta t)^2\sum_{s=1}^{4}\mathcal{V}_s^{k+1}\kappa''(-s)$$
$$=(\Delta t)^2c^2\sum_{s=-4}^{-1}\kappa''(-s)c_s^k+(2h^2(2+a\Delta t-b(\Delta t)^2)+(\Delta t)^2c^2\kappa''(0))\frac{\mathcal{V}_0^k}{\eta_0}+$$
$$(\Delta t)^2c^2\sum_{s=1}^{4}\kappa''(-s)\mathcal{V}_s^k+2h^2(\Delta t)^2\varphi(x_0,t),$$

further implies

$$-\sum_{s=-4}^{-1}\mathcal{V}_s^{k+1}\kappa''(-s)-\left(-\frac{2h^2(1+a\Delta t)}{c^2(\Delta t)^2}+\kappa''(0)\right)\frac{\mathcal{V}_0^{k+1}}{\eta_0}-\sum_{s=1}^{4}\mathcal{V}_s^{k+1}\kappa''(-s)=\sum_{s=-4}^{-1}\kappa''(-s)\mathcal{V}_s^k+$$
$$\left(2h^2\left(\frac{2+a\Delta t-b(\Delta t)^2}{c^2(\Delta t)^2}\right)+\kappa''(0)\right)\frac{\mathcal{V}_0^k}{\eta_0}+\sum_{s=1}^{4}\kappa''(-s)\mathcal{V}_s^k+2h^2(\Delta t)^2\varphi(x_0,t). \tag{2.23}$$

Now Substituting $r=1$ in equation (2.14), we have

$$2h^2(1+a\Delta t)U_1^{k+1}-c^2(\Delta t)^2\sum_{s=-3}^{5}\mathcal{V}_s^{k+1}\frac{S''(1-s)}{\eta_{1-s}}=2h^2(2+a\Delta t)u_1^k+$$
$$(\Delta t)^2\sum_{s=-3}^{5}(-2bh^2S(1-s)+c^2S''(1-s))\frac{\mathcal{V}_s^k}{\eta_{i-s}}+2h^2(\Delta t)^2\varphi(x_1,t).$$

From (2.7) we have

$$U_1^{k+1}=\frac{\mathcal{V}_1^{k+1}}{\eta_0}\quad\text{and}\quad U_1^k=\frac{\mathcal{V}_1^k}{\eta_0}, \tag{2.24}$$

using (2.24) in above expression and after simplification, we get

$$-c^2(\Delta t)^2\sum_{s=-3}^{0}\mathcal{V}_s^{k+1}S''(1-s)+(2h^2(1+a\Delta t)-c^2(\Delta t)^2S''(0))\frac{\mathcal{V}_1^{k+1}}{\eta_0}-c^2(\Delta t)^2\times\sum_{s=2}^{5}\mathcal{V}_s^{k+1}S''(1-s)$$
$$=(\Delta t)^2\sum_{s=-4}^{-1}(-2bh^2S(1-s)+c^2S''(1-s))\mathcal{V}_s^k+(2h^2(2+a\Delta t)+(\Delta t)^2(-2bh^2S(0)+$$

$$c^2S''(0))\frac{\mathcal{V}_1^k}{\eta_0}+(\Delta t)^2\sum_{s=2}^{5}(-2bh^2S(1-s)+c^2S''(1-s))\mathcal{V}_s^k+2h^2(\Delta t)^2\varphi(x_1,t),$$

using (2.2) and (2.6) in above expression, we get

$$-c^2(\Delta t)^2\sum_{s=-3}^{0}\mathcal{V}_s^{k+1}\kappa''(1-s)+(2h^2(1+a\Delta t)-c^2(\Delta t)^2\kappa''(0))\frac{\mathcal{V}_1^{k+1}}{\eta_0}-c^2(\Delta t)^2\sum_{s=2}^{5}\mathcal{V}_s^{k+1}\kappa''(1-s)$$
$$=(\Delta t)^2c^2\sum_{s=-3}^{0}\kappa''(1-s)c_s^k+(2h^2(2+a\Delta t-b(\Delta t)^2)+(\Delta t)^2c^2\kappa''(0))\frac{\mathcal{V}_1^k}{\eta_0}$$
$$+(\Delta t)^2c^2\sum_{s=2}^{5}\kappa''(1-s)\mathcal{V}_s^k+2h^2(\Delta t)^2\varphi(x_1,t),$$

further implies

$$-\sum_{s=-3}^{0}\mathcal{V}_s^{k+1}\kappa''(1-s)-\left(-\frac{2h^2(1+a\Delta t)}{c^2(\Delta t)^2}+\kappa''(0)\right)\frac{\mathcal{V}_1^{k+1}}{\eta_0}-\sum_{s=2}^{5}c_s^{k+1}\kappa''(1-s)$$
$$=\sum_{s=-3}^{0}\kappa''(1-s)\mathcal{V}_s^k+\left(2h^2\left(\frac{2+a\Delta t-b(\Delta t)^2}{c^2(\Delta t)^2}\right)+\kappa''(0)\right)\frac{\mathcal{V}_1^k}{\eta_0}+\sum_{s=2}^{5}\kappa''(1-s)\mathcal{V}_s^k+ \tag{2.25}$$
$$2h^2(\Delta t)^2\varphi(x_1,t).$$

Similarly, for $r=2$, we have

$$-\sum_{s=-2}^{1}\mathcal{V}_s^{k+1}\kappa''(2-s)-\left(-\frac{2h^2(1+a\Delta t)}{c^2(\Delta t)^2}+\kappa''(0)\right)\frac{\mathcal{V}_2^{k+1}}{\eta_0}-\sum_{s=3}^{6}\mathcal{V}_s^{k+1}\kappa''(2-s)=$$
$$\sum_{s=-2}^{1}\kappa''(2-s)\mathcal{V}_s^k+\left(2h^2\left(\frac{2+a\Delta t-b(\Delta t)^2}{c^2(\Delta t)^2}\right)+\kappa''(0)\right)\frac{\mathcal{V}_2^k}{\eta_0}+\sum_{s=3}^{6}\kappa''(2-s)\mathcal{V}_s^k \tag{2.26}$$
$$+2h^2(\Delta t)^2\varphi(x_2,t).$$

$\vdots$

for $r=N$, we have

$$-\sum_{s=-(N-4)}^{N-1}\mathcal{V}_s^{k+1}\kappa''(N-s)-\left(-\frac{2h^2(1+a\Delta t)}{c^2(\Delta t)^2}+\kappa''(0)\right)\frac{\mathcal{V}_N^{k+1}}{\eta_0}-\sum_{s=N+1}^{N+4}\mathcal{V}_s^{k+1}\kappa''(N-s)$$
$$=\sum_{s=-(N-4)}^{N-1}\kappa''(N-s)\mathcal{V}_s^k+\left(2h^2\left(\frac{2+a\Delta t-b(\Delta t)^2}{c^2(\Delta t)^2}\right)+\kappa''(0)\right)\frac{\mathcal{V}_N^k}{\eta_0}+ \tag{2.27}$$
$$\sum_{s=3}^{6}\kappa''(N-s)\mathcal{V}_s^k+2h^2(\Delta t)^2\varphi(x_N,t).$$

Combining the equations (2.23)- (2.27), we get the result (2.16). This completes the proof.

It is necessary to add more equations to obtain the unique solution of system (2.16) or (2.17).

2.3 *Boundary treatment*

Since the system (2.17) has dimensions of $(N+1)\times(N+9)$, eight additional conditions are required to ensure consistency and enable a unique solution for either (2.16) or (2.17). After incorporating the boundary conditions provided in equation (1.3), the system's dimension increases, and we establish the remaining six conditions evenly at the left and right ends of the domain. Given that the 6PBSS generates fifth-degree polynomials with an approximation order of six, the newly introduced conditions, known as forced conditions, have an order of five. These conditions are constructed as follows

The left and right end points are represented by $\mathcal{V}_{-3}, \mathcal{V}_{-2}, \mathcal{V}_{-1}$ and $\mathcal{V}_{N+1}, \mathcal{V}_{N+2}, \mathcal{V}_{N+3}$, respectively. We use following relations to find the required conditions

$$\mathcal{V}_{-\tilde{m}} = B(-x_{\tilde{m}}),\ \tilde{m} = 1,2,3,$$

where

$$B(x_{\tilde{m}}) = \sum_{r_1=1}^{6} \binom{6}{r_1} (-1)^{r_1+1} U(x_{\tilde{m}-r_1}). \tag{2.28}$$

Since by (2.5), $U(x_{\tilde{m}}) = \mathcal{V}_{\tilde{m}}$ for $\tilde{m} = 1,2,3$ and after replacing $x_{\tilde{m}}$ by $-x_{\tilde{m}}$ in (2.28), we have

$$B(-x_{\tilde{m}}) = \sum_{r_1=1}^{6} \binom{6}{r_1} (-1)^{r_1+1} \mathcal{V}_{\tilde{m}-r_1},$$

implies

$$\sum_{r_1=0}^{6} \binom{6}{r_1} (-1)^{r_1} \mathcal{V}_{r_1-\tilde{m}} = 0, \qquad \tilde{m} = 3,2,1. \tag{2.29}$$

From equation (2.29) left end conditions are obtained, similarly we have following right end conditions

$$\sum_{r_1=0}^{6} \binom{6}{r_1} (-1)^{r_1} \mathcal{V}_{\tilde{m}-r_1} = 0,\ \tilde{m} = N+3, N+2, N+1. \tag{2.30}$$

After completing the boundary treatment, we get the consistent system of order $(N+9)\times(N+9)$ as follows

$$V^{k+1} = \mathcal{J}_v V^k + \mathcal{G}, \tag{2.31}$$

here

$$\mathcal{J}_v = (\mathcal{L}_{v_0}^T, E^T, \mathcal{R}_{v_N}^T), \tag{2.32}$$

where E is defined in (2.17), $\mathcal{L}_{v_0}$ represents the matrix of left end boundary conditions in which first three rows are derived from equation (2.29) and fourth row is obtained from (1.3) at $\mathcal{V}(0) = v_0 = \rho_1(t)$. Similarly $\mathcal{R}_{v_N}$ is the matrix representation of right end boundary conditions in which first row is obtained from (1.3) at $\mathcal{V}(1) = v_N = \rho_2(t)$ and last three rows are derived using equation (). The column vector $\mathcal{G}$ is defined as

$$\mathcal{G} = (0,0,0,\rho_1, G^T, \rho_2, 0,0,0)^T, \tag{2.33}$$

where G is given in (2.21).

3. Initial Solution

Iterative procedure is required to solve the problem to start iteration we need initial vector

$$V^{(0)} = (\mathcal{V}_{-4}, \mathcal{V}_{-3}, \ldots, \mathcal{V}_{N+3}, \mathcal{V}_{N+4})^T, \tag{3.1}$$

this initial vector is obtained by initial and boundary conditions discussed as follows

$$U(x,0) = \sum_{p=-4}^{N+4} \mathcal{V}_p S_{2,p} \qquad 0 \le x \le L, \tag{3.2}$$

where $\mathcal{V}_p$'s are unknown. The initial approximation (3.2) must satisfy the following conditions

$$U(x_p, 0) = \varsigma_1(x_p), \qquad U_t(x_p, 0) = \varsigma_2(x_p), \qquad p = 0,1,\cdots,N \tag{3.3}$$

and

$$U(x_0, 0) = \rho_1(t), \qquad U(x_N, 0) = \rho_2(t). \tag{3.4}$$

Thus the following $(N+9)\times(N+9)$ matrix system is constructed to obtain the initial solution as

$$\mathcal{J}_c V^0 + \mathcal{G} = F, \tag{3.5}$$

where $\mathcal{J}_c$ is defined in (2.32), V^0 is defined in (3.1) and the column matrix F which has $(N+9)$ rows is defined as $F = (0,0,0,\ \rho_1(t),\ \varsigma_1(x_0),\ \cdots,\ \varsigma_1(x_N),\ \rho_2(t),\ 0,0,0)^T$.

3.1 *Iterative Algorithm*

In this section, a three-step iterative algorithm is presented in the following form:

- **Zeroth order Solution**: The zeroth order solution V^0 is obtained by solving (3.5).
- ***kth* order Solution** The kth order solution V^k, is obtained by solving (3.6).

$$V^k = \mathcal{J}_c V^{k-1} + \mathcal{G}, \qquad k = 1,2,3,\cdots, \tag{3.6}$$

- **Termination Conditions**:

  The process of obtaining the kth solution (3.6) will terminate if the following condition holds

$$\| V^k - V^{k-1} \| \leq \varepsilon, \qquad k = 1,2,3,\cdots, \tag{3.7}$$

  where ε is the given error tolerance.

3.2 *Stability Analysis and Errors Evaluation of SCA*

We provide the convergence and estimation of errors, results of the proposed iterative technique in this part.

Theorem 3.1: *Subdivision collocation algorithm for one dimensional linear Telegraph equation is convergent if it satisfies the relation* $\frac{b(\Delta t)^2}{3+2a\Delta t} \leq 1$ *for* $a, b, \Delta t > 0$.

Proof: To prove the convergence of algorithm, we use Von-Neumann stability approach. Detail of the procedure is given below

$$\sum_{m=-4}^{4} \kappa''(m) V_m^{k+1} = \sum_{m=-4}^{4} \kappa''(m))' V_m^k, \tag{3.8}$$

substitution of $V_m^n = \exp(i\eta mh)$ in (3.8)

$$\sum_{m=-4}^{4} \kappa''(m) \exp(i\eta mh) = \sum_{m=-4}^{4} \kappa''(m))' \exp(i\eta mh), \tag{3.9}$$

after simplification we get the result

$$\xi = \frac{A + iB}{A' + iB'}, \tag{3.10}$$

$$|\xi| = \left| \frac{A + iB}{A' + iB'} \right| \leq 1, \tag{3.11}$$

where

$$A=\frac{3}{280}cos4\eta h+\frac{8}{35}cos3\eta h-\frac{184}{105}cos2\eta h+\frac{712}{105}cos\eta h-\frac{295}{28}-\frac{2h^2(1+a\Delta t)}{c^2(\Delta t)^2},$$

$B=0,$

and

$$A'=\frac{3}{280}cos4\eta h+\frac{8}{35}cos3\eta h-\frac{184}{105}cos2\eta h+\frac{712}{105}cos\eta h-\frac{295}{28}+2h^2\left(\frac{2+a\Delta t-b(\Delta t)^2}{c^2(\Delta t)^2}\right),$$

$B'=0$, using values of A, A', B, B' in equation (3.10), after simplification we get the required result that is $\frac{b(\Delta t)^2}{3+2a\Delta t}\leq 1$ for $a, b, \Delta t>0$. shows the stability of the algorithm.

3.3 *Error estimation of SCA*

Let $u(x_r,t)$ and $U(x_r,t)$ are the exact solution and approximate solution of the TE at x_r respectively. Let N be the total number of node points between the domain [0, L] the following errors between the exact and approximate are defined as

- **Absolute error** The absolute errors L^∞ and L^2 are obtained using the following relations
$$L^\infty=\max_{1\leq i\leq N}|u(x_r,t)-U(x_r,t)|, L^2=\sqrt{\sum_{i=1}^{N}|u(x_r,t)-U(x_r,t)|^2}.$$

- **Root mean square error (RMSE)** The RMSE is defined as
$$RMSE=\sqrt{\frac{\sum_{i=1}^{N}(u(x_r,t)-U(x_r,t))^2}{N}}.$$

4. Computational Illustration of SCA

This section presents numerical results for the second-order one-dimensional TE (1.1) with its initial conditions (1.2) and the boundary conditions (1.3). We compare the approximate solutions which are obtained using the subdivision-based collocation algorithm for the TE (1.1), with known exact solutions and other existing numerical methods from the literature. The numerical computations are performed using

MATLAB, and the results are presented in tables and graphs, which are displayed for various values of t.

Example 4.1: In the following example, we address TE (1.1), with $\varsigma_1 = tan(x/2)$, & $\varsigma_2 = 1/2(1+tan^2 x/2)$, $\rho_1(t) = tan(t/2)$ & $\rho_2(t) = tan(2+t/2)$, and source term $\varphi(x,t) = \mu(1+tan^2\ (x+t/2) + v^2 tan\ (x+t/2))$ where $0 \le x \le 2$, having exact solution $tan(x+t/2)$. The parameters we use for comparison with [2, 4, 7] are $a = 2\mu$, $b = v^2$, $t = 0.2$, 0.4, 0.6, 0.8, 1 and $\Delta t = 10^{-2}$, 10^{-3}, 10^{-4} $\mu = 10$, $v = 5$, $N = 100$.

Example 4.2: In the following example, we address TE (1.1), with $\varsigma_1 = sinhx$, & $\varsigma_2 = -2sinhx$, $\rho_1(t) = 0$ & $\rho_2(t) = exp(-2t)sinhL$, and source term $\varphi(x,t) = (3-2\mu+v)\ exp\ (-2t)\ sinhx$ where $0 \le x \le 1$, having exact solution $exp(-2t)sinhx$. The parameters we use for comparison with [2, 3, 4, 7] are $a = 2\mu$, $b = v^2$, $t = 1,2$ and $\Delta t = 10^{-2}$, $\mu = 10$, $v = 5$, $N = 51$.

Example 4.3: In the following example, we address TE (1.1), with $\varsigma_1 = sinx$, & $\varsigma_2 = 0$, $\rho_1(t) = 0$ & $\rho_2(t) = costsinL$, and source term $\varphi(x,t) = -2\mu sintsinx + v^2 costsinx$ where $0 \le x \le 1$, having exact solution $costsinx$. The parameters we use for comparison with [2, 7] are $a = 2\mu$, $b = v^2$, $t = 0.2$, 0.4, 0.6, 0.8, 1 and $\Delta t = 10^{-3}$, 10^{-4}, $\mu = 6$, $v = 2$, $N = 100$, 200.

Example 4.4: In the following example, we address TE (1.1), with $\varsigma_1 = \varsigma_2 = 0$, $\rho_1(t) = 0 = \rho_2(t)$, and source term $\varphi(x,t) = exp(-t)\ ((2-2t+t^2)\ (x-x^2) + 2t^2)$ where $0 \le x \le 1$, having exact solution $(x-x^2)\ t^2 exp(-t)$. The parameters we use for comparison with [1, 2, 3, 4, 6, 9, 7] are $a = 2\mu$, $b = v^2$, $t = 0.01$,& $t = 1,2,3,4,5$ and $\Delta t = 10^{-3}$, 0.005, $\mu = 0.5$, $v = 1$, $N = 100$.

Example 4.5: In the following example, we address TE (1.1), with $\varsigma_1 = \varsigma_2 = 0$, $\rho_1(t) = 0 = \rho_2(t)$, and source term $\varphi(x,t) = exp(-t)\ (v^2x^2 + 2\mu x^2 + x^2 - 2)$ where $0 \le x \le 1$, having exact solution $x^2 exp(t)$. The parameters we use for comparison with [2] are $a = 2\mu$, $b = v^2$, for different values of $t = 2,4,6,8,10$ and $\Delta t = 10^{-2}$, 10^{-3}, $\mu = 1$, $v = 1$, $N = 32$.

Example 4.6: In the following example, we address TE (1.1), with $\varsigma_1 = sinx$, $\varsigma_2 = -sinx$, $\rho_1(t) = 0 = \rho_2(t)$, without source term and exact solution $exp(-t)sinx$ where where $0 \le x \le \pi$. The parameters we use for comparison $a = 4$, $b = 2$, for $t = 0.01$ with [5].

Example 4.7: In the following example, we address TE (1.1), with $\varsigma_1 = x^2$, $\varsigma_2 = 1$, $\rho_1(t) = t$, $\rho_2(t) = 1+t$, and source term $\varphi(x,t) = x^2 + t - 1$ where $0 \le x \le 1$, having exact solution $x^2 + t$. The parameters we use for comparison $a = 1$, $b = 1$, and $t = 0.01$ with [5].

Table 1

Error comparison of Example 4.1 for N = 100

	for $\Delta t = 10^{-2}$				for $\Delta t = 10^{-3}$							
	by [2]		by SCA		by [2]		by [7]		by [4]		by SCA	
t	L^∞	L^2	L^∞	L^2	L^∞	L^2	L^∞	L^2	L^∞	L^2	L^∞	L^2
0.2	5.0027e-7	1.0975e-7	0.0000	0.0000	5.1120e-7	1.1462e-7	2.6332e-4	1.8782e-4	2.79e-4	1.85e-4	0.0000	0.0000
0.4	1.5838e-6	3.4507e-7	0.0000	0.0000	1.6045e-6	3.5544e-7	6.9997e-4	4.8888e-4	6.35e-4	4.43e-4	0.0000	0.0000
0.6	6.9892e-6	1.4459e-6	0.0000	0.0000	7.0288e-6	1.4699e-6	1.4860e-3	9.4864e-4	1.17e-3	7.96e-4	0.0000	0.0000
0.8	5.7991e-5	1.1002e-5	0.0000	0.0000	5.7949e-5	1.1060e-5	3.4057e-3	1.8743e-3	2.40e-3	1.47e-3	0.0000	0.0000
1.0	2.4742e-3	4.0339e-4	0.0000	0.0000	2.4582e-3	4.0174e-4	1.1148e-2	5.0981e-3	9.51e-3	4.19e-3	0.0000	0.0000

Table 2

Error comparison of Example 4.1 for N = 100

	for $\Delta t = 10^{-4}$					
	by [4]		by [7]		by SCA	
t	L^∞	L^2	L^∞	L^2	L^∞	L^2
0.2	3.35e-5	2.13e-5	3.4797e-5	5.0305e-5	0.0000	0.0000
0.4	7.92e-5	5.25e-5	5.3456e-5	9.5163e-5	0.0000	0.0000
0.6	1.63e-4	1.02e-4	9.4725e-5	2.2041e-4	0.0000	0.0000
0.8	4.65e-4	2.39e-4	1.8792e-4	7.8267e-4	0.0000	0.0000
1.0	5.59e-4	1.73e-3	5.8720e-4	7.9277e-3	0.0000	0.0000

Table 3

Error comparison of Example 4.2 for N = 51

	for $\Delta t = 10^{-2}$								
	by [2]			by [3]			by SCA		
t	L^∞	L^2	RMSE	L^∞	L^2	RMSE	L^∞	L^2	RMSE
1	2.49e-9	1.20e-9	1.19e-9	1.70e-5	8.35e-5	1.19e-5	0.0000	0.0000	0.0000
2	4.03e-10	2.47e-10	2.45e-10	4.63e-6	2.37e-5	3.38e-6	0.0000	0.0000	0.0000

Table 4

Error comparison of Example 4.3 for $N = 100$

	for $\Delta t = 10^{-4}$								
	by [2]			by [7]			by SCA		
t	L^∞	L^2	RMSE	L^∞	L^2	RMSE	L^∞	L^2	RMSE
0.2	5.5379e-10	1.7141e-10	1.7056e-10	5.2412e-6	2.6903e-6	2.6775e-6	0.0000	0.0000	0.0000
0.4	5.3219e-10	1.9603e-10	1.9505e-10	8.6095e-6	5.6183e-6	5.5904e-6	0.0000	0.0000	0.0000
0.6	4.9235e-10	2.0429e-10	2.0328e-10	1.2529e-5	9.7545e-6	9.7061e-6	0.0000	0.0000	0.0000
0.8	4.3383e-10	2.0231e-10	2.0131e-10	2.0274e-5	1.3774e-5	1.3706e-5	0.0000	0.0000	0.0000
1.0	3.5991e-10	1.9219e-10	1.9123e-10	2.7555e-5	1.7347e-5	1.7261e-5	0.0000	0.0000	0.0000

Table 5

Error comparison of Example 4.3 for $N = 200$

	for $\Delta t = 10^{-3}$					
	by [2]		by [7]		by SCA	
t	L^∞	L^2	L^∞	L^2	L^∞	L^2
0.2	3.7266e-11	1.1580e-11	6.8272e-5	3.4308e-5	0.0000	0.0000
0.4	3.9851e-11	1.5794e-11	1.4935e-4	8.5756e-5	0.0000	0.0000
0.6	4.3584e-11	2.0695e-11	2.2410e-4	1.3367e-4	0.0000	0.0000
0.8	4.8410e-11	2.6223e-11	2.8978e-4	1.7532e-4	0.0000	0.0000
1.0	5.4166e-11	3.2111e-11	3.4386e-4	2.0929e-4	0.0000	0.0000

Table 6

Error comparison of Example 4.4 for $N = 100$

	for $\Delta t = 10^{-3}$									
	by [1]		by [2]		by [9]		by [7]		by SCA	
t	L^∞	L^2	L^∞	L^2	L^∞	L^2	L^∞	L^2	L^∞	L^2
1	1.13978e-7	6.52754e-8	5.3167e-12	3.8017e-12	1.8479e-5	1.4386e-4	5.9153e-5	4.5526e-5	0.0000	0.0000
2	-	-	4.4222e-12	3.1440e-12	1.0713e-5	8.0879e-5	1.7864e-5	1.4307e-5	0.0000	0.0000
3	-	-	6.8479e-13	4.8213e-13	1.8161e-5	1.2944e-4	1.4309e-5	6.4273e-6	0.0000	0.0000
4	-	-	1.5544e-12	1.1137e-12	1.6489e-5	1.1845e-4	1.3529e-5	8.9203e-6	0.0000	0.0000
5	-	-	2.1010e-13	1.4593e-13	1.0455e-5	7.5545e-5	5.2032e-6	3.0161e-6	0.0000	0.0000

Table 7

Error comparison of Example 4.4 for N = 100

	for $\Delta t = 0.005$														
	by [2]			by [3]			by [4]			by [6]			by SCA		
t	L^∞	L^2	RMSE	L^∞	L^2	RMSE	L^∞	L^2	RMSE	L^∞	L^2	RMSE	L^∞	L^2	RMSE
1	1.03e-11	8.63e-12	8.59e-12	9.93e-7	7.54e-6	7.58e-7	8.76e-5	6.31e-5	-	1.67e-7	2.61e-8	1.68e-8	0.0000	0.0000	0.0000
2	8.84e-12	7.92e-12	7.88e-12	2.12e-6	1.57e-5	2.58e-6	3.29e-5	2.34e-5	-	4.72e-7	6.10e-8	4.74e-8	0.0000	0.0000	0.0000

Table 8

Error comparison of Example 4.5 for N = 100

	for $\Delta t = 10^{-2}$				for $\Delta t = 10^{-3}$			
	by [2]			by SCA	by [2]			by SCA
t	L^∞	L^2	RMSE	L^∞ L^2, RMSE	L^∞	L^2	RMSE	L^∞ L^2, RMSE
2	7.9206e-10	5.2924e-10	5.2116e-10	0.0000	5.6077e-10	3.0167e-10	2.9706e-10	0.0000
4	5.9098e-9	3.9723e-9	3.9117e-9	0.0000	4.7043e-9	2.5397e-9	2.5009e-9	0.0000
6	4.3660e-8	2.9356e-8	2.8908e-8	0.0000	3.5321e-8	1.9078e-8	1.8786e-8	0.0000
8	3.2260e-7	2.1691e-7	2.1360e-7	0.0000	2.6155e-7	1.4128e-7	1.3912e-7	0.0000
10	2.3836e-6	1.6027e-6	1.5783e-6	0.0000	1.9332e-6	1.0442e-6	1.0283e-6	0.0000

Table 9

Error comparison of Example 4.6 for N = 20

	for $\Delta t = 10^{-2}$				
Grid point	Exact sol:	Solution by	Absolute	Solution by	Absolute errors
		[5]	errors [5]	SCA	SCA
0.0	0.000000000000	3.27459e-10	3.2745e-10	0.000000000000	0
$\frac{11}{14}$	0.700292224223	0.700292209635	2.1840e-8	0.700292224223	0
$\frac{11}{7}$	0.990049635871	0.990049654607	2.8070e-8	0.990049635871	0

Contd...

$\frac{33}{14}$	0.699406712281	0.699406697618	2.1952e-8	0.699406712281	0
$\frac{22}{7}$	-0.001251907055	-0.001251907182	2.7466e-11	-0.001251907055	0

Table 10

Error comparison of Example 4.7 for $N = 20$

	for $\Delta t = 10^{-2}$				
Grid point	Exact sol:	Solution by	Absolute errors	Solution by SCA	Absolute errors SCA
0.0	0.0100	0.0100	0	0.0100	0
$\frac{1}{4}$	0.0725	0.0725	0	0.0725	0
$\frac{1}{2}$	0.2600	0.2600	0	0.2600	0
$\frac{3}{4}$	0.5725	0.5725	0	0.5725	0
1.0	1.0100	1.0100	0	1.0100	0

4.1 *Results and discussion*

- Computational results of Example 4.1, for different values of t are presented in the Tables 1-2 with $a = 20$, $b = 25$ and $\Delta t = 10^{-2}$, 10^{-3}, 10^{-4}, $N = 100$. These tables compare the errors L^{∞}, L^{2} of current method with the existing methods of [2, 4, 7]. Figure 1 depicts the exact (black color) and approximate (red color) solutions of Example 4.1, for different values of t and $\Delta t = 10^{-3}$, $N = 100$. Computational and graphical representation shows the accuracy and efficiency of the current method. Mathematically results describe that the travelling of electrical signals along a transmission line is smooth with in a short time.
- Numerical solutions of Example 4.2, using different values of t are shown in Tables 3 with $a = 20$, $b = 25$ and $\Delta t = 10^{-2}$, $N = 51$. These tables compare the errors L^{∞}, L^{2}, $RMSE$ of current method with the existing methods of [2, 3, 4, 7]. Figure 2 depicts the exact (black color) and approximate (red color) solutions of Example 4.2, both solutions show same behavior for distinct values of t and $\Delta t = 10^{-3}$, $N = 33$. Numerical and graphical results show the accuracy and efficiency

of the current method. Mathematically results describe that the electrical signals travel rapidly along a transmission line with in a short time. Transmission of signals with position over the time t using current technique is symmetric with the given phenomena of Example 4.2.

- Numerical solutions of Example 4.3, using distinct values of t are given in Tables 4-5 with $a=12$, $b=4$ and $\Delta t=10^{-3}$, 10^{-4}, $N=100, 200$. These tables compare the errors L^{∞}, L^{2}, $RMSE$ of current method with the existing methods of [2, 7]. Figure 3 depicts the exact (black color) and approximate (red color) solutions of Example 4.3, both solutions show the same behavior for distinct values of t and $\Delta t=10^{-3}$, $N=100$. Numerical and graphical results show that current method converges smoothly with using less computational time.
- Numerical solutions of Example 4.4, using distinct values of t are shown in Tables 6-7 with $a=1$, $b=1$ and $\Delta t=10^{-3}$, $N=100$. These tables compare the errors L^{∞}, L^{2} of current method with the existing methods of [1, 2, 9, 7]. $RMSE$ for $\Delta t=0.005$, $N=100$ is calculated and listed in Table 7 compared with [2, 3, 4, 6]. Figure 4.1 depicts the exact (black color) and approximate (red color) solutions of Example 4.4, both solutions show the same behavior for distinct values of t and $\Delta t=10^{-3}$, $N=100$. It is clear from the tabulated results and graphical representation of Example 4.4 that the current method's numerical results closely match the exact solutions.
- Numerical solutions of Example 4.5, for distinct values of t are given in Tables 8 with $a=2$, $b=1$ and $\Delta t=10^{-2}, 10^{-3}$, $N=100$. These tables show the comparison of the errors L^{∞}, L^{2}, $RMSE$ of current method with the existing methods of [2]. Figure 5 depicts the exact (black color) and approximate (red color) solutions of Example 4.5, both solutions show the same behavior for distinct values of t and $\Delta t=10^{-3}$, $N=100$. It is clear from the tabulated results and graphical representation of Example 4.5 that the current method's numerical results closely match the exact solutions for distinct values of t.
- Example 4.6 and 4.7, for $a=4, b=2$ and $a=1, b=1$ are taken from [5] and the numerical data presented at different grid points in Tables 9 & 10 respectively. These tables show the comparison of approximate solution of current method and methods in [5] with absolute errors at $t=0.01$ for $\Delta t=10^{-2}$, $N=20$. Figures 6 & 7 show exact (black color) and the approximate (red color) solutions of Example 4.6 & 4.7

for $N=100, t=0.01$ respectively. It is clear from the tabulated results and graphical representations of Example 4.6 and 4.7 that the current method's numerical results closely match the exact solutions. for different values of t

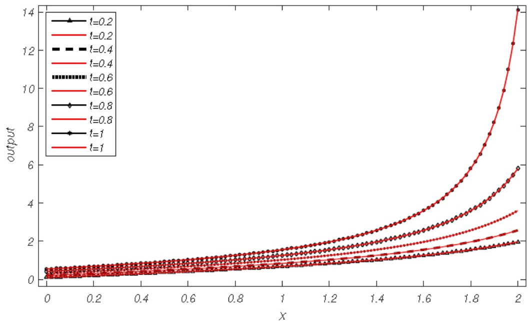

Figure 1

Graphical behavior of Example 4.1 for $\Delta t=10^{-3}, N=100$

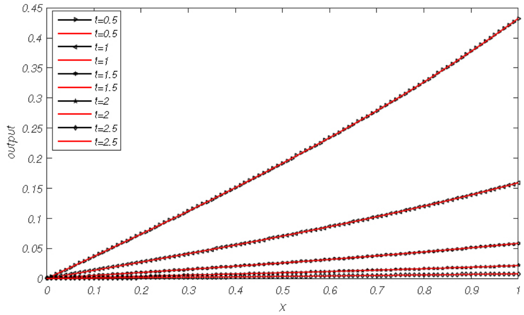

Figure 2

Graphical behavior of Example 4.2 for $\Delta t=10^{-3}, N=33$

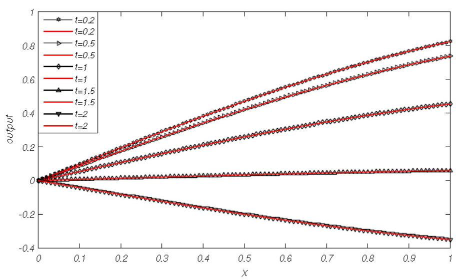

Figure 3

Graphical behavior of Example 4.3 for $\Delta t=10^{-3}, N=100$

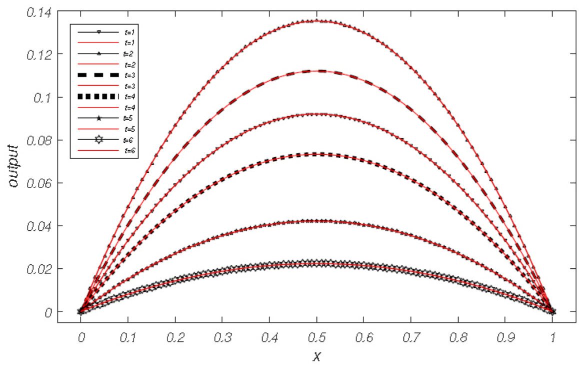

Figure 4

Graphical behavior of Example 4.4 for $\Delta t = 10^{-3}, N = 100$

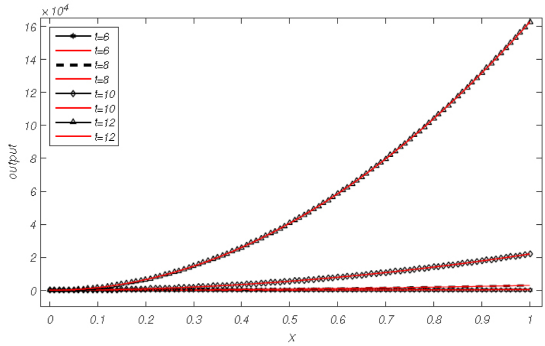

Figure 5

Graphical behavior of Example 4.5 for $\Delta t = 10^{-3}, N = 100$

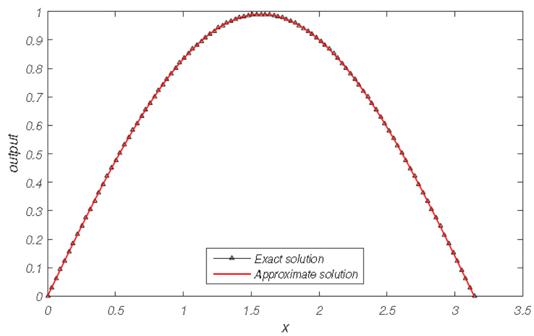

Figure 6

Graphical behavior of Example 4.6 for $t = 0.01$

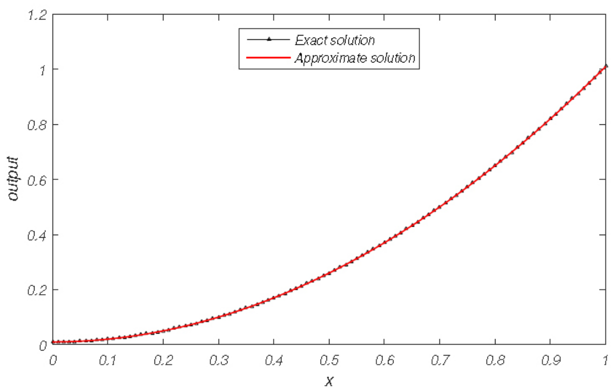

Figure 7

Comparison between exact and approximate solutions of Example 4.7 by SCA at $t = 0.01$.

5. Concluding Remarks

This paper investigates the application of the SS for finding the approximate solution of the Telegraph equation with initial conditions and Dirichlet-type boundary conditions. We use a conventional finite difference approach to discretize the time derivatives. The stability of the present algorithm is confirmed using the von Neumann (Fourier) method. The numerical results presented in Tables 1-10 and Figures 1-7 demonstrate the reliability of the obtained results. The approximate solutions obtained from the current method are compared with the following methods: the Spectral Galerkin method [1], the Differential Quadrature method based on cubic B-spline [2], Adaptive Monte Carlo methods [3], LTCM [5], the Cubic Trigonometric B-spline Collocation method [5], the Cubic B-spline Collocation method [6-7], and the collocation method in [9]. The results demonstrate that the introduced technique is highly accurate, converges rapidly, and is very easy to apply in various engineering problems. In the future, the authors are interested in extending the technique to higher-dimensional Telegraph equations.

Data Accessibility

"The data used to support the findings of the study are available within this paper".

Conflicting Interest

“The authors declare that there are no conflicts of interest regarding the publication of this paper”.

Acknowledgments

“This research is not funded by any agency”.

References

[1] Y. H. Youssri, W. M. Abd-Elhameed, and A. G. Atta, “Spectral Galerkin treatment of linear one-dimensional telegraph type problem via the generalized Lucas polynomials,” *Arabian Journal of Mathematics*, vol. 11, no. 3, pp. 601–615 (2022).

[2] A. Babu, B. Han, and N. Asharaf, “Numerical solution of the hyperbolic telegraph equation using cubic B-spline-based differential quadrature of high accuracy,” *Computational Methods for Differential Equations,* vol. 10, no. 4, pp. 837–859 (2022).

[3] Z. Hong, Y. Wang, and H. Hao, “Adaptive Monte Carlo methods for solving hyperbolic telegraph equation,” *Journal of Computational and Applied Mathematics*, vol. 345, pp. 405–415 (2019).

[4] T. Nazir, M. Abbas, and M. Yaseen, “Numerical solution of second-order hyperbolic telegraph equation via new cubic trigonometric B-splines approach,” *Cogent Mathematics*, vol. 4, no. 1, p. 1382061 (2017).

[5] A. O. Adewumi, S. O. Akindeinde, A. A. Aderogba, and B. S. Ogundare, “Laplace transform collocation method for solving hyperbolic telegraph equation,” *International Journal of Engineering Mathematics*, pp. 1–9 (2017).

[6] S. Sharifi and J. Rashidinia, “Numerical solution of hyperbolic telegraph equation by cubic B-spline collocation method,” *Applied Mathematics and Computation*, vol. 281, pp. 28–38 (2016).

[7] R. C. Mittal and R. Bhatia, “Numerical solution of second-order one-dimensional hyperbolic telegraph equation by cubic B-spline collocation method,” *Applied Mathematics and Computation*, vol. 220, pp. 496–506 (2013).

[8] M. El-Gamel and A. El-Shenawy, “The solution of a time-dependent problem by the B-spline method,” *Journal of Computational and Applied Mathematics*, vol. 267, pp. 254–265 (2014).

[9] M. Dehghan and A. Shokri, "A numerical method for solving the hyperbolic telegraph equation," *Numerical Methods for Partial Differential Equations: An International Journal*, vol. 24, no. 4, pp. 1080–1093 (2008).

[10] M. Dehghan and M. Lakestani, "The use of Chebyshev cardinal functions for solution of the second order one-dimensional telegraph equation," *Numerical Methods for Partial Differential Equations*, vol. 25, no. 4, pp. 931–938 (2009).

[11] A. Saadatmandi and M. Dehghan, "Numerical solution of hyperbolic telegraph equation using the Chebyshev tau method," *Numerical Methods for Partial Differential Equations: An International Journal,* vol. 26, no. 1, pp. 239–252 (2010).

[12] M. Dehghan and A. Ghesmati, "Solution of the second-order one-dimensional hyperbolic telegraph equation by using the dual reciprocity boundary integral equation (DRBIE) method," *Engineering Analysis with Boundary Elements*, vol. 34, no. 1, pp. 51–59 (2010).

[13] M. Lakestani and B. N. Saray, "Numerical solution of telegraph equation using interpolating scaling functions," *Computers and Mathematics with Applications*, vol. 60, no. 7, pp. 1964–1972 (2010).

[14] M. Javidi and N. Nyamoradi, "Numerical solution of telegraph equation by using LT inversion technique," *International Journal of Advanced Mathematical Sciences*, vol. 1, no. 2, pp. 64–77 (2013).

[15] M. Inc, A. Akgül, and A. Kiliçman, "Numerical solutions of the second-order one-dimensional telegraph equation based on reproducing kernel Hilbert space method," *Abstract and Applied Analysis,* Hindawi (2013).

[16] M. S. Hashemi, E. Karatas, and E. Darvishi, "Numerical treatment on one-dimensional hyperbolic telegraph equation by the method of line-group preserving scheme," *The European Physical Journal Plus,* vol. 134, no. 4, p. 153 (2019).

[17] D. Arslan, "The numerical study of a hybrid method for solving telegraph equation," *Applied Mathematics and Nonlinear Sciences*, vol. 5, no. 1, pp. 293–302 (2020).

[18] V. K. Srivastava, M. K. Awasthi, R. K. Chaurasia, and M. Tamsir, "The telegraph equation and its solution by reduced differential transform method," *Modelling and Simulation in Engineering* (2013).

[19] B. G. Lee, Y. J. Lee, and J. Yoon, "Stationary binary subdivision schemes using radial basis function interpolation," *Advances in Computational Mathematics,* vol. 25, no. 1, pp. 57–72 (2006).

[20] R. Qu, "A new approach to numerical differentiation and integration," *Mathematical and Computer Modelling*, vol. 24, no. 10, pp. 55–68 (1996).

[21] R. Qu and R. P. Agarwal, "Solving two-point boundary value problems by interpolatory subdivision algorithms," *International Journal of Computer Mathematics,* vol. 60, pp. 279–294 (1996).

[22] R. Qu and R. P. Agarwal, "A subdivision approach to the construction of approximate solutions of boundary-value problems with deviating arguments," *Journal of Computers and Mathematics with Applications,* vol. 35, no. 11, pp. 121–135 (1998).

[23] S. T. Ejaz and G. Mustafa, "A subdivision-based iterative collocation algorithm for nonlinear third-order boundary value problems," *Advances in Mathematical Physics*, vol. 2016, Article ID 5026504, 15 pages.

[24] G. Mustafa, M. Abbas, S. T. Ejaz, A. I. M. Ismail, and F. Khan, "A numerical approach based on subdivision schemes for solving nonlinear fourth-order boundary value problems," *Journal of Computational Analysis and Applications*, vol. 23, no. 4, pp. 607–623 (2017).

Received November, 2023