



## On the numerical solution of some differential equations with nonlocal integral boundary conditions via Haar wavelet

Imran Aziz<sup>a\*</sup>, Muhammad Nisar<sup>b,c</sup> and Siraj-ul-Islam<sup>d</sup>

<sup>a</sup> Department of Mathematics, University of Peshawar, Pakistan

<sup>b</sup> Department of Mathematics, FATA University, F.R Kohat, Pakistan

<sup>c</sup> Department of Mathematics and Statistics, Macquarie University, Sydney, Australia

<sup>d</sup> Department of Basic Sciences, University of Engineering and Technology, Peshawar, Pakistan

Received 8 July 2021, accepted 11 October 2021, available online 2 February 2022

© 2022 Authors. This is an Open Access article distributed under the terms and conditions of the Creative Commons Attribution 4.0 International License CC BY 4.0 (<http://creativecommons.org/licenses/by/4.0>).

**Abstract.** Differential equations with nonlocal boundary conditions are used to model a number of physical phenomena encountered in situations where data on the boundary cannot be measured directly. This study explores numerical solutions to elliptic, parabolic and hyperbolic equations with two different types of nonlocal integral boundary conditions. The numerical solutions are obtained using the Haar wavelet collocation method with the aid of Finite Differences for time derivatives. The method is applicable to both linear and nonlinear problems. To obtain the numerical solutions, Gauss elimination method is used for linear and Newton's method for nonlinear differential equations. The validity of the proposed method is demonstrated by solving several benchmark test problems from the literature: two elliptic linear and two nonlinear samples covering both types of nonlocal integral boundary conditions; one nonlinear and two linear test problems for parabolic partial differential equations; two linear samples for hyperbolic partial differential equations. The accuracy of the method is verified by comparing the numerical results with the analytical solutions. The numerical results confirm that the method is simple and effective.

**Key words:** Haar wavelet, elliptic equation, parabolic equation, hyperbolic equation, nonlocal integral condition.

### 1. INTRODUCTION

Partial Differential Equations (PDEs) play a central role in mathematical modelling of different real-world problems. Linear and nonlinear phenomena that arise in numerous areas of science such as chemical kinetics, fluid dynamics, mathematical biology, plasma physics and solid state physics, can be modelled with PDEs.

Elliptic PDEs are applied to modelling engineering problems and real physical situations such as velocity potential, stream function, etc. Parabolic PDEs are used to model flow equations describing unsteady transport problems such as heat transfer, population dynamics, etc. Hyperbolic PDEs model vibrations of structures (buildings, beams and machines) [1].

The focus of our study is on linear and nonlinear Ordinary Differential Equations (ODEs) of second order along with parabolic and hyperbolic PDEs for special types of boundary conditions called nonlocal conditions. A boundary condition for a boundary-value problem is called a nonlocal boundary condition

\* Corresponding author, [imran\\_aziz@uop.edu.pk](mailto:imran_aziz@uop.edu.pk)

if the new value on the boundary depends on data from a remote region. In other words, if the data on the boundary can not be monitored immediately, then nonlocal conditions arise. The nonlocal boundary conditions may be of different types. The type called nonlocal integral boundary conditions arises if the total amount or integral average on space domain is known, and if it is impossible to directly determine the values of the sought quantity on the boundary [2]. A variety of nonlocal mathematical models formulating different physical phenomena have been encountered in the recent past. These models are discussed for nonlocal elliptic problems in [3], parabolic problems in [4–11], and hyperbolic problems in [12–14]. The nonlocal integral boundary conditions are related to plasma physics [15], thermo-elasticity [16], chemical engineering [17] and heat conduction [18–20]. These problems are investigated for parabolic PDEs in [17–22] and for hyperbolic PDEs in [12,23]. Generally, the applications of such problems occur in many physical phenomena such as population dynamics, the process of heat conduction, control theory, theory of elastic stability, evolution equation for species' population densities, image processing, porous media flow and turbulence [24]. Some of the particular physical situations that can be modelled with the considered PDEs are discussed in [8].

In the last two decades the use of wavelets in numerical computing has come to prominence. Wavelets have a wide range of applications in scientific computing, for which reason they have been extensively used in numerical approximation in recent relevant literature. Researchers have employed various methods in applying wavelets to numerical approximations. These include wavelet collocation method [25–28], wavelet Galerkin method [29], wavelet-based finite element method [30], wavelet meshless methods [31], etc. A survey of some of the earlier work can be found in [32–36] and a review on harmonic wavelets is presented in [37]. Applications of Haar wavelet for numerical approximations are indicated in references [38–50]. Recently, Haar wavelet method has been extended to solve fractional partial differential equations [51].

In the present work, the general form of the linear elliptic problems will be as follows:

$$s(x) \frac{d^2 u}{dx^2} + w(x) \frac{du}{dx} + q(x)u(x) = r(x), \quad 0 \leq x \leq 1, \quad (1)$$

whereas the form considered for nonlinear elliptic problems will be:

$$s(x) \frac{d^2 u}{dx^2} + w(x) \frac{du}{dx} + q(x)u(x) = r(x) + G(u(x)), \quad 0 \leq x \leq 1. \quad (2)$$

Both linear and nonlinear elliptic problems indicated above are subject to two types of nonlocal integral boundary conditions, which in the present work will be called Type I and Type II integral boundary conditions. Type I integral boundary conditions are given by:

$$\begin{aligned} u(0) &= \int_0^1 \phi(x)u(x)dx + c_1, \\ u(1) &= \int_0^1 \psi(x)u(x)dx + c_2, \end{aligned} \quad (3)$$

and Type II boundary conditions by:

$$\begin{aligned} \int_0^1 \phi(x)u(x)dx &= c_1, \\ \int_0^1 \psi(x)u(x)dx &= c_2, \end{aligned} \quad (4)$$

where  $c_1$  and  $c_2$  are constants,  $\phi$  and  $\psi$  denote known functions and  $u(x)$  represents the unknown function.

With regard to parabolic PDEs with integral boundary conditions, we will consider linear parabolic PDEs in the following form:

$$\frac{\partial u}{\partial t} = s(t,x) \frac{\partial^2 u}{\partial x^2} + w(t,x) \frac{\partial u}{\partial x} + q(t,x)u(t,x) + r(t,x), \quad 0 \leq x \leq 1, t \geq 0, \quad (5)$$

and nonlinear parabolic PDEs in the following form:

$$\frac{\partial u}{\partial t} = s(t,x) \frac{\partial^2 u}{\partial x^2} + w(t,x) \frac{\partial u}{\partial x} + q(t,x)u(t,x) + r(t,x) + G(u), \quad 0 \leq x \leq 1, \quad t \geq 0, \quad (6)$$

where  $w, q, r$  and  $s$  are known functions and  $u$  is the unknown function. The nonlinear term in Eq. (6) is  $G(u)$ . Both linear and nonlinear parabolic PDEs are subject to the initial condition

$$u(0,x) = f(x), \quad (7)$$

and are either subject to Type I integral boundary conditions:

$$\begin{aligned} u(t,0) &= \int_0^1 \phi(t,x)u(t,x)dx + f_1(t), \quad t \geq 0, \\ u(t,1) &= \int_0^1 \psi(t,x)u(t,x)dx + f_2(t), \quad t \geq 0, \end{aligned} \quad (8)$$

or Type II integral boundary conditions:

$$\begin{aligned} \int_0^1 \phi(t,x)u(t,x)dx &= f_1(t), \quad t \geq 0, \\ \int_0^1 \psi(t,x)u(t,x)dx &= f_2(t), \quad t \geq 0. \end{aligned} \quad (9)$$

As for hyperbolic PDEs with integral boundary conditions, the linear hyperbolic PDEs considered in this work will have the following general form:

$$\frac{\partial^2 u}{\partial t^2} = s(t,x) \frac{\partial^2 u}{\partial x^2} + w(t,x) \frac{\partial u}{\partial x} + q(t,x)u + r(t,x), \quad 0 \leq x \leq 1, \quad t \geq 0, \quad (10)$$

and nonlinear hyperbolic PDEs the following general form:

$$\frac{\partial^2 u}{\partial t^2} = s(t,x) \frac{\partial^2 u}{\partial x^2} + w(t,x) \frac{\partial u}{\partial x} + q(t,x)u + r(t,x) + G(u(t,x)), \quad 0 \leq x \leq 1, \quad t \geq 0, \quad (11)$$

where  $u$  is unknown function and  $q, r, s$  and  $w$  are known functions. The nonlinearity in Eq. (11) is in the form of  $G(u)$ . Both linear and nonlinear hyperbolic PDEs are subject to the initial conditions:

$$\begin{aligned} u(0,x) &= f(x), \\ \frac{\partial u}{\partial t}(0,x) &= f_0(x), \end{aligned} \quad (12)$$

and are either subject to Type I integral boundary conditions of Eq. (8) or Type II integral boundary conditions of Eq. (9).

The rest of the paper is organized as follows. In Section 2, Haar wavelet and its integrals are defined. Section 3 explains the formulation of the method based on Haar wavelet for elliptic problem (Subsection 3.1), parabolic PDEs (Subsection 3.2) and hyperbolic PDEs (Subsection 3.3). Numerical results are reported in Section 4 and some conclusions are presented in Section 5.

## 2. HAAR WAVELET

The scaling function for the family of Haar wavelet is defined on the interval  $[0, 1)$  and is given as:

$$h_1(x) = \begin{cases} 1 & \text{for } x \in [0, 1) \\ 0 & \text{elsewhere.} \end{cases}$$

All other functions in the Haar wavelet family are defined on subintervals of  $[0, 1)$  and are given as follows:

$$h_i(x) = \begin{cases} 1 & \text{for } x \in [\alpha, \beta) \\ -1 & \text{for } x \in [\beta, \gamma) \\ 0 & \text{elsewhere,} \end{cases} \quad (13)$$

where

$$\alpha = \frac{k}{m}, \quad \beta = \frac{(k+0.5)}{m}, \quad \gamma = \frac{(k+1)}{m};$$

$$m = 2^j, \quad j = 0, 1, \dots, \quad k = 0, 1, \dots, m-1.$$

The integer  $j$  indicates the level of the wavelet and  $k$  is the translation parameter. The index  $i$  in Eq. (13) is calculated using the formula  $i = m + k + 1$ . The function  $h_2(x)$  is called mother wavelet and all other functions in the Haar wavelet family except the scaling function are generated from the mother wavelet by dilation and translation.

Any function  $f(x)$  which is square integrable in the interval  $(0, 1)$  can be expressed as a linear combination of the members of the Haar wavelet family as:

$$f(x) = \sum_{i=1}^{\infty} a_i h_i(x).$$

The above series terminates at finite terms if  $f(x)$  is piecewise constant or can be approximated as piecewise constant during each subinterval.

The following notations are introduced:

$$p_{i,1}(x) = \int_0^x h_i(z) dz,$$

and

$$p_{i,v+1}(x) = \int_0^x p_{i,v}(z) dz, \quad v = 1, 2, \dots$$

These integrals can be evaluated using the definition of Haar wavelet and are given as follows [52]:

$$p_{i,n}(x) = \begin{cases} 0 & \text{for } x \in [0, \alpha) \\ \frac{1}{n!} (x - \alpha)^n & \text{for } x \in [\alpha, \beta) \\ \frac{1}{n!} [(x - \alpha)^n - 2(x - \beta)^n] & \text{for } x \in [\beta, \gamma) \\ \frac{1}{n!} [(x - \alpha)^n - 2(x - \beta)^n + (x - \gamma)^n] & \text{for } x \in [\gamma, 1), \quad n = 1, 2, \dots \end{cases}$$

## 3. NUMERICAL METHOD

In this section a numerical technique for the presented problems will be developed by examining both linear and nonlinear cases. A method of solution for elliptic problems will be considered in the first subsection followed by the development of a numerical method for parabolic problems. The third subsection will introduce a method of a numerical solution for hyperbolic problems.

### 3.1. Elliptic Problem

In this section a numerical solution based on Haar wavelet for elliptic problems with nonlocal integral boundary conditions given in Eq. (3) and Eq. (4) will be discussed. For simplicity, the following notations are introduced:

$$u_0 = u(0) \quad \text{and} \quad u_1 = u(1). \quad (14)$$

In the Haar wavelet collocation method, the highest derivative in the differential equation is approximated using the Haar wavelet basis. Thus, in our case it is:

$$\frac{d^2 u}{dx^2} = \sum_{i=1}^{2M} a_i h_i(x). \quad (15)$$

Integration yields the following expressions:

$$\frac{du}{dx} = u_1 - u_0 + \sum_{i=1}^{2M} a_i (p_{i,1}(x) - p_{i,2}(1)), \quad (16)$$

and

$$u(x) = (1-x)u_0 + xu_1 + \sum_{i=1}^{2M} a_i (p_{i,2}(x) - xp_{i,2}(1)). \quad (17)$$

These expressions are then substituted in the elliptic problem and the resulting equation is discretized using the following collocation points:

$$x_j = \frac{j-0.5}{2M}, \quad j = 1, 2, \dots, 2M. \quad (18)$$

In the linear case, from Eq. (1) we obtain the following system of linear equations:

$$\begin{aligned} & \left( q(x_j)(1-x_j) - w(x_j) \right) u_0 + \left( q(x_j)x_j + w(x_j) \right) u_1 \\ & + \sum_{i=1}^{2M} a_i \left( s(x_j)h_i(x_j) + w(x_j)(p_{i,1}(x_j) - p_{i,2}(1)) + q(x_j)(p_{i,2}(x_j) - x_j p_{i,2}(1)) \right) \\ & = r(x_j), \quad j = 1, 2, 3, \dots, 2M, \end{aligned} \quad (19)$$

whereas for nonlinear case, from Eq. (2) the following system of nonlinear equations is obtained:

$$\begin{aligned} & \left( q(x_j)(1-x_j) - w(x_j) \right) u_0 + \left( q(x_j)x_j + w(x_j) \right) u_1 \\ & + \sum_{i=1}^{2M} a_i \left( s(x_j)h_i(x_j) + w(x_j)(P_{i,1}(x_j) - P_{i,2}(1)) + q(x_j)(p_{i,2}(x_j) - x_j p_{i,2}(1)) \right) \\ & - G \left( (1-x_j)u_0 + x_j u_1 + \sum_{i=1}^{2M} a_i (p_{i,2}(x_j) - x_j p_{i,2}(1)) \right) = r(x_j), \quad j = 1, 2, 3, \dots, 2M. \end{aligned} \quad (20)$$

The next step is to incorporate the nonlocal integral boundary conditions into the above systems. The procedure is discussed for Type I and Type II integral boundary conditions separately.

### Type I integral boundary conditions

We examine Type I integral boundary conditions first, which are given in Eq. (3). Substituting Haar approximation of the unknown function  $u(x)$  in Eq. (3), we obtain

$$\begin{aligned} u(0) &= \int_0^1 \phi(x) \left( (1-x)u_0 + xu_1 + \sum_{i=1}^{2M} a_i(p_{i,2}(x) - xp_{i,2}(1)) \right) dx + c_1, \\ u(1) &= \int_0^1 \psi(x) \left( (1-x)u_0 + xu_1 + \sum_{i=1}^{2M} a_i(p_{i,2}(x) - xp_{i,2}(1)) \right) dx + c_2. \end{aligned} \quad (21)$$

The integrals in the above conditions are approximated using the Haar wavelet method for numerical integration [53] and thus we obtain

$$\begin{aligned} \left( 1 - \frac{1}{2M} \sum_{k=1}^{2M} \phi(x_k)(1-x_k) \right) u_0 - \frac{1}{2M} \sum_{k=1}^{2M} \phi(x_k)x_k u_1 \\ - \frac{1}{2M} \sum_{i=1}^{2M} a_i \left( \sum_{k=1}^{2M} \phi(x_k) (p_{i,2}(x_k) - x_k p_{i,2}(1)) \right) = c_1, \end{aligned} \quad (22)$$

and

$$\begin{aligned} -\frac{1}{2M} \sum_{k=1}^{2M} \psi(x_k)(1-x_k)u_0 + \left( 1 - \frac{1}{2M} \sum_{k=1}^{2M} \psi(x_k)x_k \right) u_1 \\ - \frac{1}{2M} \sum_{i=1}^{2M} a_i \left( \sum_{k=1}^{2M} \psi(x_k) (p_{i,2}(x_k) - x_k p_{i,2}(1)) \right) = c_2. \end{aligned} \quad (23)$$

### Type II integral boundary conditions

Type II integral boundary conditions are given in Eq. (4). Substituting Haar approximation of the unknown function  $u(x)$  in Eq. (4), we get

$$\begin{aligned} \int_0^1 \phi(x) \left( (1-x)u_0 + xu_1 + \sum_{i=1}^{2M} a_i(p_{i,2}(x) - xp_{i,2}(1)) \right) dx = c_1, \\ \int_0^1 \psi(x) \left( (1-x)u_0 + xu_1 + \sum_{i=1}^{2M} a_i(p_{i,2}(x) - xp_{i,2}(1)) \right) dx = c_2. \end{aligned} \quad (24)$$

Applying numerical integration by using Haar wavelet, we obtain

$$\frac{1}{2M} \sum_{k=1}^{2M} \phi(x_k)(1-x_k)u_0 + \frac{1}{2M} \sum_{k=1}^{2M} \phi(x_k)x_k u_1 + \frac{1}{2M} \sum_{i=1}^{2M} a_i \left( \sum_{k=1}^{2M} \phi(x_k) (p_{i,2}(x_k) - x_k p_{i,2}(1)) \right) = c_1, \quad (25)$$

and

$$\frac{1}{2M} \sum_{k=1}^{2M} \psi(x_k)(1-x_k)u_0 + \frac{1}{2M} \sum_{k=1}^{2M} \psi(x_k)x_k u_1 + \frac{1}{2M} \sum_{i=1}^{2M} a_i \left( \sum_{k=1}^{2M} \psi(x_k) (p_{i,2}(x_k) - x_k p_{i,2}(1)) \right) = c_2. \quad (26)$$

### Linear Case

For the linear elliptic problems, the linear system given in Eq. (19) is combined either with Eqs (22) and (23) in the case of Type I integral boundary conditions or with Eqs (25) and (26) in the case of Type II integral boundary conditions. In either case we obtain a system of  $2M + 2$  linear equations with  $2M + 2$  unknowns:  $a_i, i = 1, 2, \dots, 2M, u(0)$  and  $u(1)$ . Solving this system of equations by any linear solver, we obtain the values of the  $2M + 2$  unknowns. The numerical solution at any point of the domain can then be obtained easily by putting the values of these unknowns in Eq. (17). We can also obtain the first and second order derivatives of the unknown function  $u(x)$  at any point of the domain by substituting the values in Eqs (16) and (15), respectively.

### Nonlinear Case

For the nonlinear elliptic problems, in the case of Type I integral boundary conditions of Eqs (22) and (23) are combined with the nonlinear system given in Eq. (20), whereas in the case of Type II integral boundary conditions of Eqs (25) and (26) are combined with the nonlinear system given in Eq. (20). In each case a system of  $2M + 2$  nonlinear equations with  $2M + 2$  unknowns  $a_i, i = 1, 2, \dots, 2M, u(0)$  and  $u(1)$  is obtained. This system may be solved using any iterative method such as Newton's method, Broyden's method, etc. The numerical solution at any point of the domain is obtained using a similar procedure as described above for the linear case.

Both Newton's and Broyden's methods require the calculation of the Jacobian. We will describe the procedure for calculating the Jacobian in the case of Type I integral boundary conditions only. For Type II integral boundary conditions a similar procedure can be applied.

We rewrite the nonlinear system given in Eq. (20) in combination with Eq. (22) and Eq. (23) for Type I integral boundary conditions in the following form:

$$F_j(u_0, u_1, a_1, a_2, \dots, a_N) = \left( q(x_j)(1-x_j) - w(x_j) \right) u_0 + \left( q(x_j)x_j + w(x_j) \right) u_1 + \sum_{i=1}^{2M} a_i \left( s(x_j)h_i(x_j) + w(x_j) \left( P_{i,1}(x_j) - P_{i,2}(1) \right) + q(x_j) \left( p_{i,2}(x_j) - x_j p_{i,2}(1) \right) \right) - G \left( (1-x_j)u_0 + x_j u_1 + \sum_{i=1}^{2M} a_i (p_{i,2}(x_j) - x_j p_{i,2}(1)) \right) - r(x_j), \quad j = 1, 2, 3, \dots, 2M, \quad (27)$$

$$F_{N+1}(u_0, u_1, a_1, a_2, \dots, a_N) = \left( 1 - \frac{1}{2M} \sum_{k=1}^{2M} \phi(x_k)(1-x_k) \right) u_0 - \frac{1}{2M} \sum_{k=1}^{2M} \phi(x_k)x_k u_1 - \frac{1}{2M} \sum_{i=1}^{2M} a_i \left( \sum_{k=1}^{2M} \phi(x_k) (p_{i,2}(x_k) - x_k p_{i,2}(1)) \right) - c_1, \quad j = 1, 2, \dots, 2M, \quad (28)$$

and

$$F_{N+2}(u_0, u_1, a_1, a_2, \dots, a_N) = -\frac{1}{2M} \sum_{k=1}^{2M} \psi(x_k)(1-x_k)u_0 + \left( 1 - \frac{1}{2M} \sum_{k=1}^{2M} \psi(x_k)x_k \right) u_1 - \frac{1}{2M} \sum_{i=1}^{2M} a_i \left( \sum_{k=1}^{2M} \psi(x_k) (p_{i,2}(x_k) - x_k p_{i,2}(1)) \right) - c_2, \quad j = 1, 2, \dots, 2M. \quad (29)$$

For calculating the Jacobian, partial derivatives are obtained as follows:

$$\frac{\partial F_j}{\partial a_r} = s(x_j)h_r(x_j) + w(x_j) \left( p_{r,1}(x_j) - p_{r,2}(1) \right) + q(x_j) \left( p_{r,2}(x_j) - x_j p_{r,2}(1) \right) - G'(u) \left( p_{r,2}(x_j) - x_j p_{r,2}(1) \right), \quad j, r = 1, 2, \dots, 2M,$$

$$\begin{aligned}\frac{\partial F_j}{\partial u_0} &= q(x_j)(1-x_j) - w(x_j) - G'(u)(1-x_j), \quad j = 1, 2, \dots, 2M, \\ \frac{\partial F_j}{\partial u_1} &= q(x_j)x_j + w(x_j) - G'(u)x_j, \quad j = 1, 2, \dots, 2M, \\ \frac{\partial F_{N+1}}{\partial a_r} &= -\frac{1}{2M} \sum_{k=1}^{2M} \phi(x_k) \left( p_{r,2}(x_k) - x_k p_{r,2}(1) \right), \quad r = 1, 2, \dots, 2M, \\ \frac{\partial F_{N+2}}{\partial a_r} &= -\frac{1}{2M} \sum_{k=1}^{2M} \psi(x_k) \left( p_{r,2}(x_k) - x_k p_{r,2}(1) \right), \quad r = 1, 2, \dots, 2M, \\ \frac{\partial F_{N+1}}{\partial u_0} &= 1 - \frac{1}{2M} \sum_{k=1}^{2M} \phi(x_k)(1-x_k), \quad \frac{\partial F_{N+1}}{\partial u_1} = -\frac{1}{2M} \sum_{k=1}^{2M} \phi(x_k)x_k,\end{aligned}$$

and

$$\frac{\partial F_{N+2}}{\partial u_0} = -\frac{1}{2M} \sum_{k=1}^{2M} \psi(x_k)(1-x_k), \quad \frac{\partial F_{N+2}}{\partial u_1} = 1 - \frac{1}{2M} \sum_{k=1}^{2M} \psi(x_k)x_k.$$

### 3.2. Parabolic PDEs

In this section, a numerical solution based on Haar wavelet for parabolic PDEs with Type I and Type II nonlocal integral boundary conditions is discussed in view of both linear and nonlinear parabolic problems. The time derivative is discretized using finite difference scheme whereas Haar wavelet is used for space derivatives.

#### Time discretization

Let  $t_n = t_o + n\Delta t$ ,  $n = 1, 2, \dots, T_f/\Delta t$  be the time discretization where  $\Delta t$  is the time-step size,  $t_o$  denotes the initial time and  $T_f$  refers to the final time. For a time period  $[t_n, t_{n+1}]$ , the time derivative for parabolic problems is estimated by using Euler's formula given below:

$$\frac{\partial u}{\partial t} \approx \frac{u^{n+1}(x) - u^n(x)}{\Delta t}, \quad n = 0, 1, 2, \dots, \frac{T_f}{\Delta t} - 1, \quad (30)$$

where  $u^n(x) = u(t_n, x)$ . Applying this time discretization to the linear parabolic Eq. (5), we obtain:

$$(1 - \Delta t q^n(x))u^{n+1}(x) = u^n(x) + \Delta t s^n(x) \frac{\partial^2 u^{n+1}(x)}{\partial x^2} + \Delta t w^n(x) \frac{\partial u^{n+1}(x)}{\partial x} + \Delta t r^n(x), \quad (31)$$

and similarly, applying the time discretization to the nonlinear parabolic Eq. (6), we have:

$$(1 - \Delta t q^n(x))u^{n+1}(x) = u^n(x) + \Delta t s^n(x) \frac{\partial^2 u^{n+1}(x)}{\partial x^2} + \Delta t w^n(x) \frac{\partial u^{n+1}(x)}{\partial x} + \Delta t r^n(x) + \Delta t G(u^{n+1}(x)), \quad (32)$$

where we use the notations:

$$s^n(x) = s(t_n, x), \quad q^n(x) = q(t_n, x), \quad w^n(x) = w(t_n, x), \quad \text{and} \quad r^n(x) = r(t_n, x). \quad (33)$$

The magnitude of time-step size for the explicit method is restricted by the following condition:

$$\frac{\Delta t}{h^2} \leq \frac{1}{2},$$

where  $h$  is the spatial step size.

### Space discretization

We will use Haar wavelet for discretization of space derivatives. Let us assume that the higher order space derivatives are approximated using Haar wavelet as follows:

$$\frac{\partial^2 u}{\partial x^2} = \sum_{i=1}^{2M} a_i(t) h_i(x). \quad (34)$$

Integration yields the following:

$$\frac{\partial u}{\partial x} = u(t, 1) - u(t, 0) + \sum_{i=1}^{2M} a_i(t) (p_{i,1}(x) - P_{i,2}(1)), \quad (35)$$

and

$$u(t, x) = (1 - x)u(t, 0) + xu(t, 1) + \sum_{i=1}^{2M} a_i(t) (p_{i,2}(x) - xp_{i,2}(1)). \quad (36)$$

These expressions are then substituted in the parabolic problem and collocation points defined in Eq. (18) are used for spatial discretization. The following system of linear equations is obtained in the linear case of the parabolic equations:

$$\begin{aligned} & \left( (1 - \Delta t q_j^n)(1 - x_j) + \Delta t w_j^n \right) u_0^{n+1} + \left( (1 - \Delta t q_j^n)x_j - \Delta t w_j^n \right) u_1^{n+1} \\ & + \sum_{i=1}^{2M} a_i(t^{n+1}) \left( (1 - \Delta t q_j^n)(p_{i,2}(x_j) - x_j p_{i,2}(1)) - \Delta t s_j^n h_i(x_j) - \Delta t w_j^n (P_{i,1}(x_j) - P_{i,2}(1)) \right) \\ & = u_j^n + \Delta t r_j^n, \quad j = 1, 2, 3, \dots, 2M, \end{aligned} \quad (37)$$

whereas for the nonlinear case of the parabolic equations, the following system of nonlinear equations is obtained:

$$\begin{aligned} & \left( (1 - \Delta t q_j^n)(1 - x_j) + \Delta t w_j^n \right) u_0^{n+1} + \left( (1 - \Delta t q_j^n)x_j - \Delta t w_j^n \right) u_1^{n+1} \\ & + \sum_{i=1}^{2M} a_i(t^{n+1}) \left( (1 - \Delta t q_j^n)(p_{i,2}(x_j) - x_j p_{i,2}(1)) - \Delta t s_j^n h_i(x_j) - \Delta t w_j^n (P_{i,1}(x_j) - P_{i,2}(1)) \right) \\ & - \Delta t G \left( (1 - x_j)u_0^{n+1} + x_j u_1^{n+1} + \sum_{i=1}^{2M} a_i(t^{n+1})(p_{i,2}(x_j) - x_j p_{i,2}(1)) \right) \\ & = u_j^n + \Delta t r_j^n, \quad j = 1, 2, 3, \dots, 2M, \end{aligned} \quad (38)$$

where

$$u_0^{n+1} = u(t_{n+1}, 0), \quad u_1^{n+1} = u(t_{n+1}, 1) \quad \text{and} \quad u_j^n = u(t_n, x_j). \quad (39)$$

The next step is to incorporate the nonlocal integral boundary conditions into the above systems. This is performed in a similar way as explained above in the case of elliptic problems.

### Type I integral boundary conditions

For Type I integral boundary conditions we obtain the following equations:

$$\begin{aligned} & \left( 1 - \frac{1}{2M} \sum_{k=1}^{2M} \phi_k^n (1 - x_k) \right) u_0^{n+1} - \frac{1}{2M} \sum_{k=1}^{2M} \phi_k^n x_k u_1^{n+1} \\ & - \frac{1}{2M} \sum_{i=1}^{2M} a_i(t^{n+1}) \left( \sum_{k=1}^{2M} \phi_k^n (p_{i,2}(x_k) - x_k p_{i,2}(1)) \right) = f_1(t^n), \end{aligned} \quad (40)$$

$$\begin{aligned} & \left( 1 - \frac{1}{2M} \sum_{k=1}^{2M} \psi_k^n x_k \right) u_1^{n+1} - \frac{1}{2M} \sum_{k=1}^{2M} \psi_k^n (1 - x_k) u_0^{n+1} \\ & - \frac{1}{2M} \sum_{i=1}^{2M} a_i(t^{n+1}) \left( \sum_{k=1}^{2M} \psi_k^n (p_{i,2}(x_k) - x_k p_{i,2}(1)) \right) = f_2(t^n). \end{aligned} \quad (41)$$

### Type II integral boundary conditions

For Type II integral boundary conditions the following equations are obtained:

$$\begin{aligned} & \frac{1}{2M} \sum_{k=1}^{2M} \phi_k^n (1 - x_k) u_0^{n+1} + \frac{1}{2M} \sum_{k=1}^{2M} \phi_k^n x_k u_1^{n+1} \\ & + \frac{1}{2M} \sum_{i=1}^{2M} a_i(t^{n+1}) \left( \sum_{k=1}^{2M} \phi_k^n (p_{i,2}(x_k) - x_k p_{i,2}(1)) \right) = f_1(t^n), \end{aligned} \quad (42)$$

$$\begin{aligned} & \frac{1}{2M} \sum_{k=1}^{2M} \psi_k^n (1 - x_k) u_0^{n+1} + \frac{1}{2M} \sum_{k=1}^{2M} \psi_k^n x_k u_1^{n+1} \\ & + \frac{1}{2M} \sum_{i=1}^{2M} a_i(t^{n+1}) \left( \sum_{k=1}^{2M} \psi_k^n (p_{i,2}(x_k) - x_k p_{i,2}(1)) \right) = f_2(t^n). \end{aligned} \quad (43)$$

### 3.3. Hyperbolic PDEs

A numerical solution based on Haar wavelet for hyperbolic problems with Type I (8) and Type II (9) integral boundary conditions will be discussed in this section. A second-order finite difference scheme is used to discretize time derivatives whereas Haar wavelet is used for space derivatives.

#### Time discretization

A similar procedure as discussed in the case of parabolic problems is applied to time discretization of hyperbolic problems. The time derivative for hyperbolic problems is estimated using a second-order central difference formula

$$\frac{\partial^2 u}{\partial t^2} \approx \frac{u^{n+1}(x) - 2u^n(x) + u^{n-1}(x)}{\Delta t^2}, \quad n = 1, 2, \dots, \frac{T_f}{\Delta t}. \quad (44)$$

Applying this time discretization to linear hyperbolic Eq. (10), we obtain:

$$\begin{aligned} (1 - \Delta t^2 q^n(x)) u^{n+1}(x) &= 2u^n(x) - u^{n-1}(x) \\ &+ \Delta t^2 s^n(x) \frac{\partial^2 u^{n+1}(x)}{\partial x^2} + \Delta t^2 w^n(x) \frac{\partial u^{n+1}(x)}{\partial x} + \Delta t^2 r^n(x), \end{aligned} \quad (45)$$

and similarly, applying the time discretization to nonlinear hyperbolic Eq. (11), we have:

$$\begin{aligned} (1 - \Delta t^2 q^n(x)) u^{n+1}(x) &= 2u^n(x) - u^{n-1}(x) + \Delta t^2 s^n(x) \frac{\partial^2 u^{n+1}(x)}{\partial x^2} \\ &+ \Delta t^2 w^n(x) \frac{\partial u^{n+1}(x)}{\partial x} + \Delta t^2 r^n(x) + \Delta t^2 G(u^{n+1}(x)), \end{aligned} \quad (46)$$

where  $u^{n-1}(x)$  is the solution at the previous time-step,  $u^n(x)$  denotes the solution at the current time-step and  $u^{n+1}(x)$  refers to the solution at the next time-step. Applying Euler's formula to the second initial condition (12), we obtain:

$$u^n(x) = u^{n-1}(x) + \Delta t f_\circ(x). \quad (47)$$

Using the solution at the previous time-step and Eq. (47), we obtain the solution at the current time-step. The solution at the next time-step is then obtained by using the solution at the current time-step, at the previous time-step and using Eq. (45) for the linear case or Eq. (46) for the nonlinear case.

### Space discretization

Haar wavelet is used for discretization of space derivatives in a similar way as explained for parabolic problems. In the case of linear hyperbolic problems, the following system of linear equations is obtained:

$$\begin{aligned} & \left( (1 - \Delta t^2 q_j^n)(1 - x_j) + \Delta t^2 w_j^n \right) u_0^{n+1} + \left( (1 - \Delta t^2 q_j^n)x_j - \Delta t^2 w_j^n \right) u_1^{n+1} \\ & + \sum_{i=1}^{2M} a_i \left( (1 - \Delta t^2 q_j^n) \left( p_{i,2}(x_j) - x_j p_{i,2}(1) \right) - \Delta t^2 s_j^n h_i(x_j) - \Delta t^2 w_j^n \left( p_{i,1}(x_j) - p_{i,2}(1) \right) \right) \\ & = 2u_j^n - u_j^{n-1} + \Delta t^2 r_j^n, \quad j = 1, 2, 3, \dots, 2M, \end{aligned} \quad (48)$$

whereas for the nonlinear case of hyperbolic equations, the following system of nonlinear equations is obtained:

$$\begin{aligned} & \left( (1 - \Delta t^2 q_j^n)(1 - x_j) + \Delta t^2 w_j^n \right) u_0^{n+1} + \left( (1 - \Delta t^2 q_j^n)x_j - \Delta t^2 w_j^n \right) u_1^{n+1} \\ & + \sum_{i=1}^{2M} a_i \left( (1 - \Delta t^2 q_j^n) \left( p_{i,2}(x_j) - x_j p_{i,2}(1) \right) - \Delta t^2 s_j^n h_i(x_j) - \Delta t^2 w_j^n \left( p_{i,1}(x_j) - p_{i,2}(1) \right) \right) \\ & - \Delta t^2 G \left( (1-x)u_0^{n+1} + xu_1^{n+1} + \sum_{i=1}^{2M} a_i (p_{i,2}(x) - xp_{i,2}(1)) \right) \\ & = 2u_j^n - u_j^{n-1} + \Delta t^2 r_j^n, \quad j = 1, 2, 3, \dots, 2M. \end{aligned} \quad (49)$$

Next, the nonlocal integral boundary conditions are incorporated into the above systems in a similar way as discussed in the case of parabolic problems.

## 4. NUMERICAL EXPERIMENTS

In this section, numerical experiments are presented in order to demonstrate the accuracy, wide applicability and convergence of the proposed method. The method is applied to both linear and nonlinear samples. For linear problems we have used Gauss elimination method and for nonlinear problems Newton's method. The initial guess for Newton's method was taken to be one and the iterations were terminated when the convergent criterion  $10^{-5}$  was satisfied. The notation  $L_\infty$  is used for maximum absolute errors at  $N = 2M$  collocation points. The experimental rate of convergence  $R_c(N)$  is also calculated, which is defined as

$$R_c(N) = \frac{\log \left( \frac{L_\infty(\frac{N}{2})}{L_\infty(N)} \right)}{\log 2}.$$

**Test Problem 1.** Consider the elliptic equation:

$$2\frac{d^2u}{dx^2} - x\frac{du}{dx} - 3u(x) = -e^x(1+x), \quad 0 \leq x \leq 1,$$

subject to the first type of integral boundary conditions:

$$u(0) = \int_0^1 xu(x)dx,$$

$$u(1) = \int_0^1 u(x)dx - 1.$$

The exact solution to the problem is given by

$$u(x) = e^x.$$

This is a linear equation with Type I integral boundary conditions. The proposed method is applied to this problem and the numerical results are reported in Table 1. This table shows that the proposed method has a very good performance in terms of accuracy as well as efficiency for this test problem. The beauty of the present method is that it can easily handle the integral boundary conditions and the performance of the method in the presence of integral boundary conditions is as good as in the presence of any other local boundary conditions such as Dirichlet, Neumann, periodic, etc. [25].

Table 1 shows maximum absolute errors, experimental rates of convergence, condition numbers of the matrices involved and CPU times for different numbers of collocation points ( $N$ ) starting from  $N = 2$  to  $N = 2048$ . One important feature of the Haar wavelet collocation method is that it gives better accuracy by increasing the number of collocation points, as can be observed in the table. The maximum absolute errors are decreased up to order of  $10^{-7}$  for  $N = 1024$  and  $N = 2048$ . One can obtain even better accuracy by increasing further the number of collocation points; however, this will increase computational cost as well. The experimental rates of convergence for this test problem approach 2 and this confirms the theoretical results obtained in [54,55]. Another important feature of the Haar wavelet collocation method is that the matrix involved has a small condition number. The table shows that the condition number for the matrix of the size  $2048 \times 2048$  is only 674.5. Due to this important feature we do not face the problem of ill conditioning in the Haar wavelet collocation method. The last column of the table validates that the proposed method is very efficient, as for  $N = 2048$  the numerical solutions are obtained in only 1.916 seconds.

**Table 1.**  $L_\infty$  of Test Problem 1 at different numbers of collocation points

$J$	$N$	$L_\infty$	$R_c$	Condition number	CPU times
0	2	$8.66 \times 10^{-2}$		19.4193	0.004
1	4	$2.46 \times 10^{-2}$	1.8157	29.0057	0.004
2	8	$6.50 \times 10^{-3}$	1.9201	41.7553	0.004
3	16	$1.70 \times 10^{-3}$	1.9349	59.3989	0.004
4	32	$4.21 \times 10^{-4}$	2.0111	84.1830	0.005
5	64	$1.06 \times 10^{-4}$	1.9914	119.1562	0.006
6	128	$2.65 \times 10^{-5}$	1.9958	168.5767	0.019
7	256	$6.65 \times 10^{-6}$	1.9978	238.4458	0.050
8	512	$1.66 \times 10^{-6}$	1.9989	337.2420	0.162
9	1024	$4.16 \times 10^{-7}$	1.9994	476.9520	0.469
10	2048	$1.04 \times 10^{-7}$	1.9996	674.5258	1.916

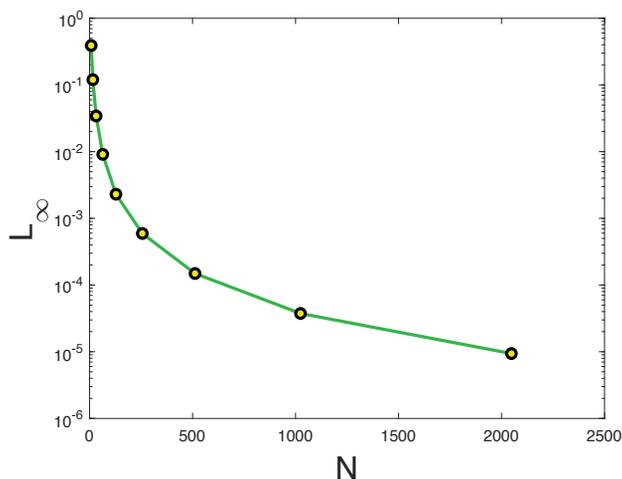


Fig. 1.  $L_\infty$  errors for Test Problem 2.

**Test Problem 2.** Consider the elliptic equation:

$$2x \frac{d^2 u}{dx^2} - x^2 \frac{du}{dx} - \cos x u(x) = -e^x ((-4+x)x \cos x + (x^2 + \cos x) \sin x), \quad 0 \leq x \leq \pi,$$

subject to the integral boundary conditions:

$$\int_0^\pi (1+x)u(x)dx = \frac{\pi e^\pi}{2},$$

$$\int_0^\pi x^2 u(x)dx = \frac{1 + e^\pi (-1 + \pi)^2}{2}.$$

The exact solution to the problem is given by

$$u(x) = e^x \sin x.$$

This is also a linear problem, but with Type II integral boundary conditions. The performance of the proposed method for this type of boundary conditions is equally good. The numerical results in terms of  $L_\infty$  errors are shown in Fig. 1. It is obvious from the figure that the accuracy of the method improves by increasing the number of collocation points.

We have also compared the approximate solution at the collocation points obtained by the proposed method with the exact solution in Fig. 2 for  $N = 64$ . The figure shows that in this case both of the solutions match closely.

**Test Problem 3.** Consider the nonlinear elliptic equation:

$$x^2 \frac{d^2 u}{dx^2} - x \frac{du}{dx} - 2u = \frac{-1 + (-2+x)x}{(1+x)^3} - u^2, \quad 0 \leq x \leq 1,$$

subject to the following integral boundary conditions:

$$u(0) = \int_0^1 x u(x) dx - \log 2,$$

$$u(1) = \int_0^1 x^2 u(x) dx - 1 + \log 2.$$

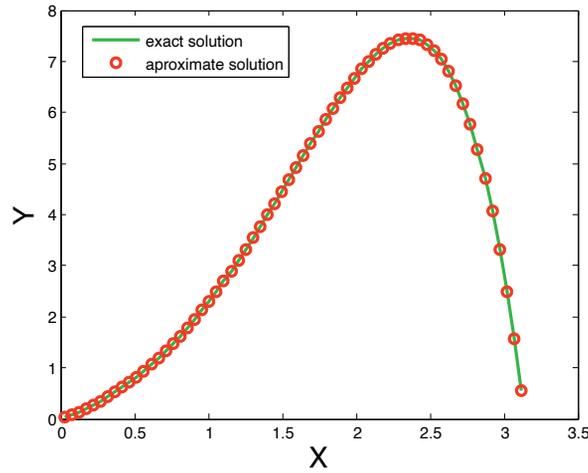


Fig. 2. Comparison of the exact and the approximate solution for Test Problem 2.

The exact solution to the problem is given by

$$u(x) = \frac{1}{1+x}.$$

This is a nonlinear equation and after discretizing the equation using Haar wavelet approximations the resulting nonlinear system of equations is solved by Newton’s method. The initial guess was taken to be  $a_i = 1, \forall i = 1, 2, \dots, N$ , and the iterations were terminated when the convergence criterion of  $10^{-5}$  was satisfied. In Table 2, we have shown the number of iterations of Newton’s method for this test problem using different numbers of collocation points which are equal to either 4 or 5. This demonstrates that the proposed method is converging very fast. In addition to numbers of iterations, also maximum absolute errors, numerical rates of convergence and CPU times for different numbers of collocation points are presented in the table. Moreover, the excellent performance of the proposed method in terms of  $L_\infty$  errors is easily observable. The numerical rates of convergence in this case approach 2 as well. The CPU times confirm that the method is efficient also for nonlinear problems.

Table 2.  $L_\infty$  of Test Problem 3 at different numbers of collocation points

$J$	$N$	$L_\infty$	$R_c$	Iterations	CPU times
0	2	$1.03 \times 10^{-2}$		4	0.0069
1	4	$2.40 \times 10^{-3}$	2.0887	4	0.0092
2	8	$5.64 \times 10^{-4}$	2.0952	4	0.0080
3	16	$1.37 \times 10^{-4}$	2.0365	4	0.0073
4	32	$3.42 \times 10^{-5}$	2.0070	4	0.0148
5	64	$8.53 \times 10^{-6}$	2.0030	5	0.0194
6	128	$2.13 \times 10^{-6}$	2.0002	5	0.0317
7	256	$5.33 \times 10^{-7}$	2.0000	5	0.0928
8	512	$1.33 \times 10^{-7}$	2.0000	5	0.4869
9	1024	$3.33 \times 10^{-8}$	2.0000	5	2.5549
10	2048	$8.33 \times 10^{-9}$	1.9999	5	11.9585

**Test Problem 4.** Consider the nonlinear elliptic problem:

$$\frac{d^2u}{dx^2} + 2\frac{du}{dx} + xu = 2\cos x + \sin x(-1 + x - \sin x) + u^2, \quad 0 \leq x \leq \pi,$$

subject to the following integral boundary conditions:

$$\int_0^\pi u(x)dx = 2,$$

$$\int_0^\pi x^2 u(x)dx = -4 + \pi^2.$$

The exact solution to the problem is given by

$$u(x) = \sin x.$$

This is another nonlinear elliptic equation with Type II integral boundary conditions. The proposed method was applied to this test problem as well, and the resulting nonlinear system was solved using Newton's method. The initial guess for this test problem was also taken as  $a_i = 1, \quad \forall i = 1, 2, \dots, N$ , and the iterations were terminated using the same criterion of  $10^{-5}$  as in the previous sample. The numerical results are shown in Table 3. As it can be seen in the table, the method is accurate as the maximum absolute errors are decreasing up to order of  $10^{-7}$ , the numerical rate of convergence is approximately equal to 2, the method is converging very fast as the number of iterations for different numbers of collocation points is not more than 6. The efficiency of this method is further validated, as for  $N = 2048$  the CPU time is only 10.969 seconds.

**Test Problem 5.** Consider the linear parabolic equation [9]:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + \frac{-2(x^2 + t + 1)}{(t + 1)^3}, \quad 0 < x < 1, \quad 0 < t < 1,$$

subject to the initial condition:

$$u(0, x) = x^2, \quad 0 < x < 1,$$

**Table 3.**  $L_\infty$  of Test Problem 4 for different numbers of collocation points

$J$	$N$	$L_\infty$	$R_c$	Iteration	CPU time
0	2	$1.09 \times 10^{-1}$		3	0.0076
1	4	$3.08 \times 10^{-2}$	1.8252	5	0.0067
2	8	$8.60 \times 10^{-3}$	1.8374	5	0.0081
3	16	$3.10 \times 10^{-3}$	1.4737	6	0.0143
4	32	$9.26 \times 10^{-4}$	1.7419	6	0.0162
5	64	$2.53 \times 10^{-4}$	1.8719	6	0.0314
6	128	$6.61 \times 10^{-5}$	1.9357	6	0.0483
7	256	$1.69 \times 10^{-5}$	1.9678	6	0.1222
8	512	$4.27 \times 10^{-6}$	1.9838	6	0.5230
9	1024	$1.07 \times 10^{-6}$	1.9919	6	2.4394
10	2048	$2.69 \times 10^{-7}$	1.9959	6	10.969

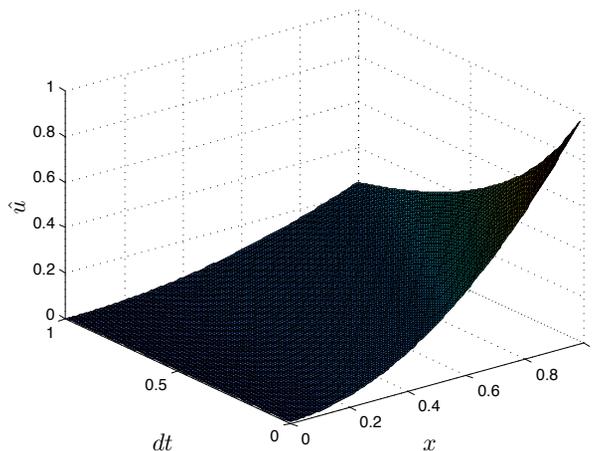


Fig. 3. 2D plot of the approximate solution for Test Problem 5.

and integral boundary conditions:

$$u(0, x) = \int_0^1 xu(t, x)dx + \frac{-1}{4(t+1)^2}, \quad 0 < t < 1,$$

$$u(1, x) = \int_0^1 xu(t, x)dx + \frac{3}{4(t+1)^2}, \quad 0 < t < 1.$$

The exact solution to the problem is given by

$$u(t, x) = \left( \frac{x}{t+1} \right)^2.$$

In order to show the performance of the proposed method in the case of parabolic equations with integral boundary conditions, we have presented this test problem, which is a linear parabolic equation with Type I integral boundary conditions. The approximate solution using the proposed method for  $N = 128$  and  $\Delta t = 0.01$  is shown in Fig. 3. As illustrated in the figure, the value of the unknown function  $u(x, t)$  is approaching 1 when  $x = 1, t = 0$  and  $u(x, t)$  is approaching 0.25 when  $x = 1, t = 1$ . The correctness of these values can be confirmed by the exact solution to this test sample.

The numerical results for this test problem at  $t = 1$  are reported in Table 4. We have taken 3 different values of  $\Delta t$ :  $10^{-2}$ ,  $10^{-3}$  and  $10^{-4}$ . For  $\Delta t = 10^{-2}$  no considerable improvement in the accuracy of the method can be seen when we increase the number of collocation points from  $N = 16$  onwards. However, up to  $N = 16$  the method has an excellent performance with rates of convergence nearly equal to 2. Similarly, for  $\Delta t = 10^{-3}$  the performance of the method is good up to  $N = 64$ . But from  $N = 64$  onwards a poor performance of the method can be observed. Finally, for  $\Delta t = 10^{-4}$  the proposed method has been applied up to  $N = 256$  and the method performs very well with rates of convergence approximately equal to 2. Based on these observations we have come to the conclusion that in order to obtain a more accurate solution, small values of  $\Delta t$  and a large number of collocation points should be used.

**Test Problem 6.** Consider the linear parabolic equation [56]:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + \sin x, \quad 0 < x < \pi, \quad 0 < t < 1,$$

**Table 4.** Numerical results of Test Problem 5 at  $t = 1$  and at different numbers of collocation points

$J$	$N$	$L_\infty$	$R_c$	CPU times
$(\Delta t = 10^{-2})$				
0	2	$1.93 \times 10^{-2}$		0.0242
1	4	$4.80 \times 10^{-3}$	2.0075	0.0221
2	8	$1.00 \times 10^{-3}$	2.2630	0.0257
3	16	$1.79 \times 10^{-4}$	2.4775	0.0155
4	32	$1.31 \times 10^{-4}$	0.4506	0.0255
5	64	$1.89 \times 10^{-4}$	-0.5293	0.0702
6	128	$2.04 \times 10^{-4}$	-0.1066	0.0610
7	256	$2.07 \times 10^{-4}$	-0.0255	0.1810
$(\Delta t = 10^{-3})$				
0	2	$1.95 \times 10^{-2}$		0.1065
1	4	$4.90 \times 10^{-3}$	1.9926	0.0865
2	8	$1.20 \times 10^{-3}$	2.0297	0.0993
3	16	$2.91 \times 10^{-4}$	2.0402	0.1008
4	32	$5.97 \times 10^{-5}$	2.2881	0.1264
5	64	$8.26 \times 10^{-6}$	2.8530	0.1613
6	128	$1.58 \times 10^{-5}$	-0.936	0.2303
7	256	$1.94 \times 10^{-5}$	-0.296	0.6727
$(\Delta t = 10^{-4})$				
0	2	$1.95 \times 10^{-2}$		0.8576
1	4	$5.00 \times 10^{-3}$	1.9635	0.8837
2	8	$1.20 \times 10^{-3}$	2.0589	0.8801
3	16	$3.09 \times 10^{-4}$	1.9535	0.8817
4	32	$7.59 \times 10^{-5}$	2.0282	0.9937
5	64	$1.74 \times 10^{-5}$	2.1185	1.4303
6	128	$3.30 \times 10^{-6}$	2.3696	2.5609
7	256	$8.53 \times 10^{-7}$	1.9872	8.4505

subject to the initial condition:

$$u(0, x) = \cos x, \quad 0 < x < \pi,$$

and integral boundary conditions:

$$\int_0^\pi xu(t, x) dx = -(2 + \pi)e^{-t} + \pi, \quad 0 < t < 1,$$

$$\int_0^\pi (L - x)u(t, x) dx = (2 + \pi - 2L)e^{-t} + 2L - \pi, \quad 0 < t < 1,$$

where  $L$  is a constant, the value of which is taken  $L = 1$ . The exact solution to the problem is given by

$$u(t, x) = e^{-t} \cos x + (1 - e^{-t}) \sin x.$$

This is also a linear parabolic equation with Type II integral boundary conditions. The approximate solution for  $\Delta t = 10^{-2}$  and  $N = 128$  is shown in Fig. 4. The accuracy of the method in terms of  $L_\infty$  error norms

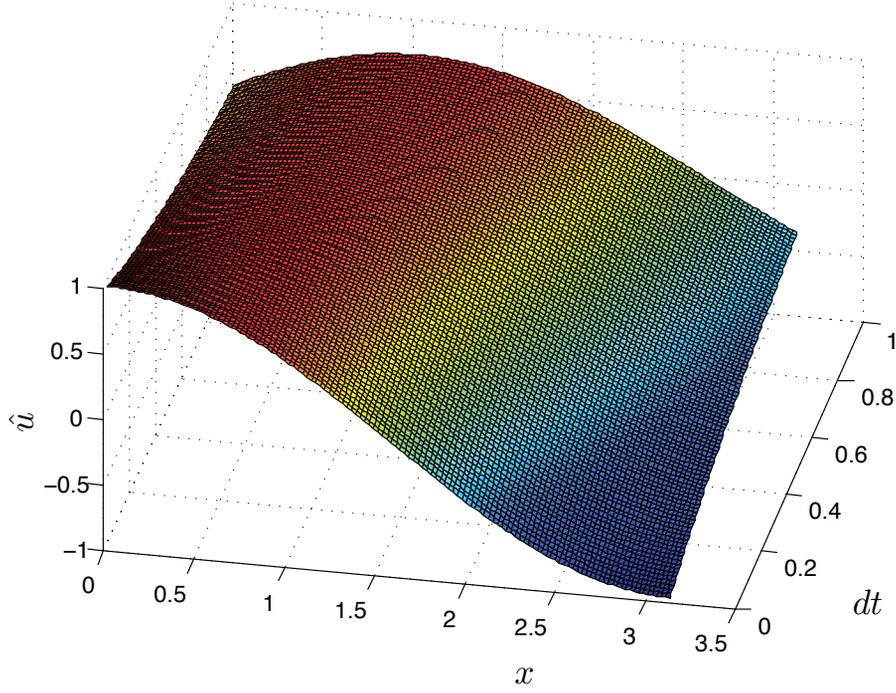


Fig. 4. Plot of the approximate solution for Test Problem 6.

for different values of time-step  $\Delta t$  and different numbers of collocation points  $N$  is depicted in Fig. 5. Similarly to the previous problem, the proposed method does not behave well for large values of time-steps  $\Delta t$ . However, better accuracy can be obtained by decreasing the value of time-step  $\Delta t$  and increasing the number of collocation points.

**Test Problem 7.** Consider the nonlinear parabolic equation [56]:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} - u \frac{\partial u}{\partial x}, \quad 0 < x < 1, \quad 0 < t < 1,$$

subject to the initial condition:

$$u(0, x) = x, \quad 0 < x < 1,$$

and integral boundary conditions:

$$\int_0^1 u(t, x) dx = \frac{1}{2(1+t)}, \quad 0 < t < 1,$$

$$\int_0^1 (e^x - 1)u(t, x) dx = \frac{1}{2(1+t)}, \quad 0 < t < 1.$$

The exact solution to the problem is given by

$$u(t, x) = \frac{x}{1+t}.$$

This is a nonlinear parabolic equation with the nonlinear term  $u \frac{\partial u}{\partial x}$ . Before applying the proposed method to this test problem, we have linearized the nonlinear term according to the following formula:

$$u(t, x) \frac{\partial u}{\partial x}(t, x) = u(t, x) \frac{\partial u}{\partial x}(t_0, x) + u(t_0, x) \frac{\partial u}{\partial x}(t, x) - 2u(t_0, x) \frac{\partial u}{\partial x}(t_0, x).$$

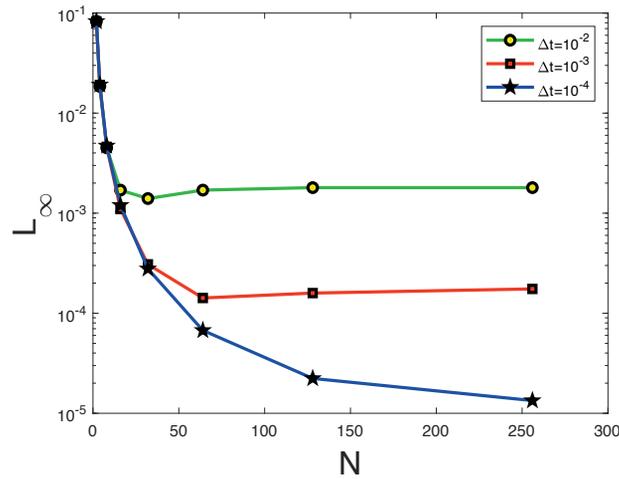


Fig. 5.  $L_\infty$  errors for different time-step sizes for Test Problem 6.

The numerical results at  $t = 1$  and for  $\Delta t = 10^{-1}$  are reported in Table 5. It can be seen that the performance of the proposed method for nonlinear parabolic problems is as good as for linear parabolic problems.

**Test Problem 8.** Consider the linear hyperbolic equation [57]:

$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2} + 2e^{-t-x}(t-x), \quad 0 \leq x \leq 1, \quad 0 \leq t \leq 1,$$

subject to the initial conditions:

$$\begin{aligned} u(0, x) &= 0, \\ \frac{\partial u(0, x)}{\partial t} &= xe^{-x}, \quad 0 \leq x \leq 1, \end{aligned}$$

**Table 5.** Numerical results of Test Problem 7 at  $t = 1$  and at different numbers of collocation points

$J$	$N$	$L_\infty$	$R_c$	CPU times
$(\Delta t = 10^{-1})$				
0	2	$5.48 \times 10^{-2}$		0.010
1	4	$1.70 \times 10^{-2}$	1.6886	0.006
2	8	$4.80 \times 10^{-3}$	1.8244	0.009
3	16	$1.30 \times 10^{-3}$	1.8845	0.015
4	32	$3.35 \times 10^{-4}$	1.9562	0.042
5	64	$8.52 \times 10^{-5}$	1.9739	0.036
6	128	$2.15 \times 10^{-5}$	1.9868	0.064
7	256	$5.40 \times 10^{-6}$	1.9933	0.132
8	512	$1.35 \times 10^{-6}$	1.9967	0.774
9	1024	$3.38 \times 10^{-7}$	1.9984	4.744
10	2048	$8.47 \times 10^{-8}$	1.9991	25.804

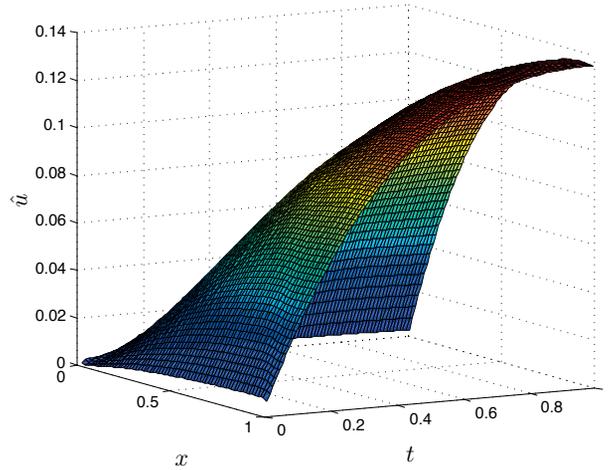


Fig. 6. Plot of the numerical solution for Test Problem 8.

and integral boundary conditions:

$$u(0,t) = \int_0^1 xte^{-t-x}dx - 2(-2+e)e^{-t-1}t, \quad 0 < t < 1,$$

$$u(1,t) = \int_0^1 (1-x)xe^{-t-x}dx + 2(-2+e)e^{-t-1}t, \quad 0 < t < 1.$$

The exact solution to the problem is given by

$$u(t,x) = xte^{-t-x}.$$

This sample is taken to apply the performance of the proposed method to hyperbolic problems with integral boundary conditions. This is a linear hyperbolic problem with Type I integral boundary conditions. Figure 6 shows the approximate solution using the proposed method for  $N = 128$  and at  $\Delta t = 10^{-2}$ . It is clear from the figure that the value of the unknown function  $u(x,t)$  is approaching 0 when  $x = 1, t = 0$  and  $u(x,t)$  is approaching  $\frac{1}{e^2}$  when  $x = 1$ , and  $t = 1$ . The correctness of these values can be confirmed by the exact solution to this test problem.

Table 6 reports the numerical results for this test problem at  $t = 1$  and for 3 different values of  $\Delta t$ :  $10^{-2}$ ,  $10^{-3}$  and  $10^{-4}$ . For  $\Delta t = 10^{-2}$  no considerable improvement in the accuracy of the method can be seen when we increase the number of collocation points from  $N = 8$  onwards. However, up to  $N = 8$  the method has good performance. Similarly, for  $\Delta t = 10^{-3}$  the method performs well up to  $N = 32$ . But from  $N = 32$  onwards a poor performance of the method is observed. Finally, for  $\Delta t = 10^{-4}$  we have applied the proposed method up to  $N = 512$  and the performance of the method is good for smaller values of  $\Delta t$ . Based on these observations we have come to the conclusion that in order to obtain a more accurate solution, small values of  $\Delta t$  and a large number of collocation points should be considered.

**Test Problem 9.** Consider the linear hyperbolic equation [56]:

$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2}, \quad 0 \leq x \leq 1, \quad 0 \leq t \leq 1,$$

**Table 6.** Numerical results at  $t = 1$  for Test Problem 8

$J$	$N$	$L_\infty$	$R_c$	CPU times
$(\Delta t = 10^{-2})$				
0	2	$1.300 \times 10^{-2}$		0.023
1	4	$4.300 \times 10^{-3}$	1.5961	0.020
2	8	$2.200 \times 10^{-3}$	0.9668	0.021
3	16	$1.600 \times 10^{-3}$	0.4594	0.022
4	32	$1.400 \times 10^{-3}$	0.1926	0.031
5	64	$1.400 \times 10^{-3}$	0.1926	0.060
$(\Delta t = 10^{-3})$				
0	2	$1.320 \times 10^{-2}$		0.103
1	4	$4.000 \times 10^{-3}$	1.7225	0.096
2	8	$1.000 \times 10^{-3}$	2.0000	0.110
3	16	$3.935 \times 10^{-4}$	1.3455	0.101
4	32	$2.616 \times 10^{-4}$	0.5888	0.122
5	64	$2.128 \times 10^{-4}$	0.2979	0.194
6	128	$1.994 \times 10^{-4}$	0.0941	0.225
7	256	$1.969 \times 10^{-4}$	0.0181	0.567
$(\Delta t = 10^{-4})$				
0	2	$1.320 \times 10^{-2}$		0.665
1	4	$4.100 \times 10^{-3}$	1.6868	0.684
2	8	$1.100 \times 10^{-3}$	1.8981	0.659
3	16	$2.745 \times 10^{-4}$	2.0026	0.696
4	32	$7.874 \times 10^{-5}$	1.8016	0.761
5	64	$4.104 \times 10^{-5}$	0.9399	1.084
6	128	$2.883 \times 10^{-5}$	0.5095	1.705
7	256	$2.438 \times 10^{-5}$	0.2421	5.181
8	512	$2.272 \times 10^{-5}$	0.1017	56.25

subject to the initial conditions:

$$u(0, x) = x^2, \quad 0 \leq x \leq 1,$$

$$\frac{\partial u(0, x)}{\partial t} = 0,$$

and integral boundary conditions:

$$\int_0^1 x^2 + t^2 dx = \frac{1}{3} + t^2, \quad 0 < t < 1,$$

$$\int_0^1 x^3 + xt^2 dx = \frac{1}{4} + \frac{t^2}{2}, \quad 0 < t < 1.$$

The exact solution to the problem is given by

$$u(t, x) = x^2 + t^2.$$

This is also a linear hyperbolic equation with Type II integral boundary conditions. The approximate solution for  $\Delta t = 10^{-2}$  and  $N = 128$  is shown in Fig. 7. The accuracy of the method in terms of maximum absolute

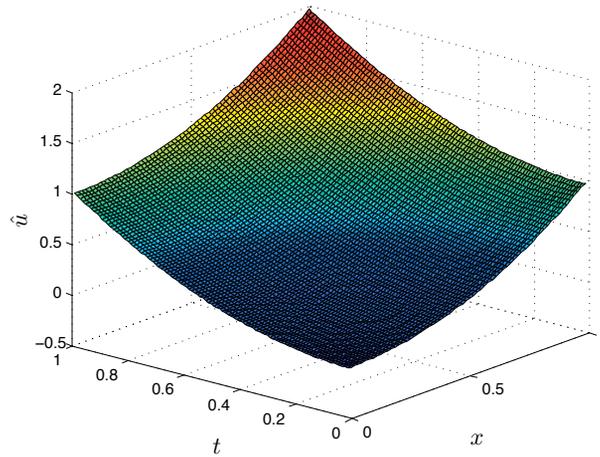


Fig. 7. Plot of the numerical solution for Test Problem 9.

errors for different values of time-step  $\Delta t$  and different numbers of collocation points  $N$  is illustrated in Table 7. It can be seen from the table that the performance of the method is not very good for large values of time-steps  $\Delta t$ . However, better accuracy can be obtained by decreasing the value of time-step  $\Delta t$  and increasing the number of collocation points.

Table 7. Numerical results at  $t = 1$  for Test Problem 9

$J$	$2M$	$L_\infty$	$R_c$	CPU times
$(\Delta t = 10^{-2})$				
1	4	$3.130 \times 10^{-2}$		0.005
2	8	$8.900 \times 10^{-3}$	1.814	0.004
3	16	$2.300 \times 10^{-3}$	1.952	0.010
4	32	$5.819 \times 10^{-4}$	1.982	0.005
5	64	$3.210 \times 10^{-4}$	0.858	0.010
6	128	$2.948 \times 10^{-4}$	0.122	0.024
7	256	$2.887 \times 10^{-4}$	0.0302	0.047
$(\Delta t = 10^{-3})$				
1	4	$3.210 \times 10^{-2}$		0.029
2	8	$1.080 \times 10^{-2}$	1.571	0.024
3	16	$3.100 \times 10^{-3}$	1.800	0.014
4	32	$1.000 \times 10^{-3}$	1.632	0.033
5	64	$3.340 \times 10^{-4}$	1.582	0.057
6	128	$8.017 \times 10^{-5}$	2.058	0.054
7	256	$5.550 \times 10^{-5}$	0.530	0.101
8	512	$2.492 \times 10^{-5}$	1.155	0.479
9	1024	$9.162 \times 10^{-6}$	1.443	2.264

## 5. CONCLUSIONS

The Haar wavelet collocation method is applied in order to find a numerical solution of elliptic, parabolic and hyperbolic equations with two different types of nonlocal integral boundary conditions. The method has several advantages, e.g. it can be applied to both linear and nonlinear problems and performs equally well in both cases. The method has simpler applicability and good accuracy. It is efficient as for a very large number of collocation points the results are obtained in fractions of a second. The validity of the proposed method is confirmed by several test problems. The method may be further refined by considering a  $\theta$ -weighted scheme instead of a finite difference scheme for time derivatives in the parabolic and hyperbolic cases.

## ACKNOWLEDGEMENT

The publication costs of this article were covered by the Estonian Academy of Sciences.

## REFERENCES

1. Siraj-ul-Islam, Aziz, I., Al-Fhaid, A. S. and Shah, A. A numerical assessment of parabolic partial differential equations using Haar and Legendre wavelets. *Appl. Math. Model.*, 2013, **37**(23), 9455–9481.
2. Siraj-ul-Islam, Aziz, I. and Ahmad, M. Numerical solution of two-dimensional elliptic PDEs with nonlocal boundary conditions. *Comput. Math. Appl.*, 2015, **69**(3), 180–205.
3. Diaz, J. I. On a nonlocal elliptic problem arising in the magnetic confinement of a plasma in a stellarator. *Nonlinear Anal.*, 1997, **30**(7), 3963–3974.
4. Bouziani, A. On a class of parabolic equations with a nonlocal boundary condition. *Acad. Roy. Belg. Bull. Cl. Sci.*, 1999, **10**, 61–77.
5. Bouziani, A. Strong solution for a mixed problem with nonlocal condition for certain pluriparabolic equations. *Hiroshima Math. J.*, 1997, **27**(3), 373–390.
6. Day, W. A. Existence of a property of solutions of the heat equation subject to linear thermoelasticity and other theories. *Q. Appl. Math.*, 1982, **40**, 319–330.
7. Day, W. A. A decreasing property of solutions of parabolic equations with applications to thermoelasticity. *Q. Appl. Math.*, 1983, **41**, 468–475.
8. Day, W. A. Parabolic equations and thermodynamics. *Q. Appl. Math.*, 1992, **50**(3), 523–533.
9. Martin-Vaquero, J. and Vigo-Aguiar, J. A note on efficient techniques for the second-order parabolic equation subject to nonlocal conditions. *Appl. Numer. Math.*, 2009, **59**(6), 1258–1264.
10. Wang, S. and Lin, Y. A numerical method for the diffusion equation with nonlocal boundary specifications. *Int. J. Eng. Sci.*, 1990, **28**(6), 543–546.
11. Mohammadi, M., Mokhtari, R. and Panahipour, H. Solving two parabolic inverse problems with a nonlocal boundary condition in the reproducing kernel space. *Appl. Comput. Math.*, 2014, **13**, 91–106.
12. Beilin, S. A. Existence of solutions for one-dimensional wave equations with nonlocal conditions. *Electron. J. Diff. Equ.*, 2001, **76**, 1–8. <http://eudml.org/doc/122051>
13. Beilin, S. On a mixed nonlocal problem for a wave equation. *Electron. J. Diff. Equ.*, 2006, **2006**(103), 1–10. <http://eudml.org/doc/127384>
14. Pulkina, L. S. A non-local problem with integral conditions for hyperbolic equations. *Electron. J. Diff. Equ.*, 1999, **1999**(45), 1–6. <https://ejde.math.txstate.edu>
15. Samarskii, A. A. Some problems in differential equations theory. *Diff. Equ.*, 1980, **16**(11), 1221–1228.
16. Shi, P. Weak solution to evolution problem with a nonlocal constraint. *SIAM J. Math. Anal.*, 1993, **24**, 46–58.
17. Choi, Y.S. and Chan, K. Y. A parabolic equation with nonlocal boundary conditions arising from electrochemistry. *Nonlinear Anal. Theory Methods Appl.*, 1992, **18**(4), 317–331.
18. Cahlon, B., Kulkarni, D. M. and Shi, P. Stepwise stability for the heat equation with a nonlocal constraint. *SIAM J. Numer. Anal.*, 1995, **32**(2), 571–593.
19. Cannon, J. R. The solution of the heat equation subject to the specification of energy. *Q. Appl. Math.*, 1963, **21**, 155–160.
20. Kamynin, L. I. A boundary value problem in the theory of the heat conduction with nonclassical boundary condition. *USSR Comput. Math. Math. Phys.*, 1964, **4**(6), 33–59.
21. Batten, G. W. Second-order correct boundary conditions for the numerical solution of the mixed boundary problem for parabolic equations. *Math. Comput.*, 1963, **17**, 405–413.

22. Yurchuk, N. I. Mixed problem with an integral condition for certain parabolic equations. *Differ. Equ.*, 1986, **22**, 1457–1463.
23. Pulkina, L. A nonlocal problem with integral conditions for hyperbolic equations. *Electron. J. Diff. Eq.*, 1999, **45**, 1–6.
24. Guezane-Lakoud, A. and Frioui, A. Nonlinear three point boundary-value problem. *Sarajevo J. Math.*, 2012, **8**(20), 101–106.
25. Siraj-ul-Islam, Aziz, I. and Šarler, B. The numerical solution of second-order boundary-value problems by collocation method with the Haar wavelets. *Math. Comput. Model.*, 2010, **52**(9–10), 1577–1590.
26. Moradipour, M. and Yousefi, S. A. Using two collocation methods to solve the Black-Scholes partial differential equation of American options. *Appl. Comput. Math.*, 2016, **15**(2), 200–211.
27. Rashid, A. and Ismail, A. I. B. M. A Chebyshev spectral collocation method for the coupled nonlinear Schrödinger equation. *Appl. Comput. Math.*, 2010, **9**(1), 104–115.
28. Heydari, M. H., Hooshmandasl, M. R. and Cattani, C. Wavelet method for solving nonlinear stochastic Itô-Volterra integral equations. *Georgian Math. J.*, 2020, **27**(1), 81–95.
29. Jang, G.-W., Kim, Y. Y. and Choi, K. K. Remesh-free shape optimization using the wavelet-Galerkin method. *Int. J. Solids Struct.*, 2004, **41**(22–23), 6465–6483.
30. Diaz, L. A., Martin, M. T. and Vampa, V. Daubechies wavelet beam and plate finite elements. *Finite Elem. Anal. Des.*, 2009, **45**, 200–209.
31. Liu, Y. N., Liu, Y. H. and Cen, Z. Z. Daubechies wavelet meshless method for 2-D elastic problems. *Tsinghua Sci. Technol.*, 2008, **13**, 605–608.
32. Dahmen, W., Kurdila, A. and Oswald, P. (eds). *Multiscale Wavelet Methods for Partial Differential Equations*. Academic Press, San Diego, 1997.
33. Kaya, D., Gulbahar, S. and Yokuş, A. Numerical solutions of the fractional KdV-Burgers-Kuramoto equation. *Therm. Sci.*, 2018, **22**(1), 153–158.
34. Kaya, D., Gulbahar, S., Yokuş, A. and Gulbahar, M. Solutions of the fractional combined KdV-mKdV equation with collocation method using radial basis function and their geometrical obstructions. *Adv. Differ. Equ.*, 2018, 77. <https://doi.org/10.1186/s13662-018-1531-0>
35. Singh, V., Siraj-ul-Islam and Mohanty, R. K. Local meshless method for convection dominated steady and unsteady partial differential equations. *Eng. Comput.*, 2019, **35**(3), 803–812.
36. Mohanty, R. K. and Khurana, G. A new spline in compression method of order four in space and two in time based on half-step grid points for the solution of the system of 1D quasi-linear hyperbolic partial differential equations. *Adv. Differ. Equ.*, 2017, **2017**(97). <https://doi.org/10.1186/s13662-017-1147-9>
37. Cattani, C. A review on harmonic wavelets and their fractional extension. *J. Adv. Eng. Comput.*, 2018, **2**(4), 224–238.
38. Lepik, Ü. and Hein, H. *Haar Wavelets with Applications*. Springer, 2014.
39. Chen, C. F. and Hsiao, C. H. Haar wavelet method for solving lumped and distributed-parameter systems. *IEE Proc., Control Theory Appl.*, 1997, **144**(1), 87–94.
40. Hsiao, C. H. Haar wavelet approach to linear stiff systems. *Math. Comput. Simul.*, 2004, **64**(5), 561–567.
41. Hsiao, C. H. and Wang, W. J. Haar wavelet approach to nonlinear stiff systems. *Math. Comput. Simul.*, 2001, **57**(6), 347–353.
42. Lepik, Ü. Numerical solution of evolution equations by the Haar wavelet method. *Appl. Math. Comput.*, 2007, **185**, 695–704.
43. Lepik, Ü. Haar wavelet method for nonlinear integro-differential equations. *Appl. Math. Comput.*, 2006, **176**(1), 324–333.
44. Maleknejad, K. and Mirzaee, F. Using rationalized Haar wavelet for solving linear integral equations. *Appl. Math. Comput.*, 2005, **160**(2), 579–587.
45. Babolian, E. and Shamsavaran, A. Numerical solution of nonlinear Fredholm integral equations of the second kind using Haar wavelets. *J. Comput. Appl. Math.*, 2009, **225**(1), 87–95.
46. Majak, J., Pohlak, M., Eerme, M. and Shvartsman, B. Solving ordinary differential equations with higher order Haar wavelet method. *AIP Conf. Proc.*, 2019, **2116**(330002), 1–4.
47. Majak, J., Pohlak, M., Karjust, K., Eerme, M., Kurnitski, J. and Shvartsman, B. New higher order Haar wavelet method: Application to FGM structures. *Compos. Struct.*, 2018, **201**, 72–78.
48. Majak, J., Shvartsman, B., Ratas, M., Bassir, D., Pohlak, M., Karjust, K. and Eerme, M. Higher-order Haar wavelet method for vibration analysis of nanobeams. *Mater. Today Commun.*, 2020, **25**, 101290.
49. Ratas, M., Salupere, A. and Majak, J. Solving nonlinear PDEs using the higher order Haar wavelet method on nonuniform and adaptive grids. *Math. Model. Anal.*, 2021, **26**(1), 147–169.
50. Ratas, M. and Salupere, A. Application of higher order Haar wavelet method for solving nonlinear evolution equations. *Math. Model. Anal.*, 2020, **25**(2), 271–288.
51. Zada, L. and Aziz, I. The numerical solution of fractional Korteweg-de Vries and Burgers' equations via Haar wavelet, mathematical methods in the applied sciences. *Math. Method Appl. Sci.*, 2021, **44**(13), 10564–10577.
52. Siraj-ul-Islam, Aziz, I. and Al-Fhaid, A. An improved method based on Haar wavelets for numerical solution of nonlinear integral and integro-differential equations of first and higher orders. *J. Comput. Appl. Math.*, 2014, **260**, 449–469.
53. Siraj-ul-Islam, Aziz, I. and Haq, F. A comparative study of numerical integration based on Haar wavelets and hybrid functions. *Comput. Math. Appl.*, 2010, **59**(6), 2026–2036.
54. Majak, J., Shvartsman, B., Karjust, K., Mikola, M., Haavajõe, A. and Pohlak, M. On the accuracy of the Haar wavelet discretization. *Compos. Part B*, 2015, **80**, 321–327.

55. Majak, J., Shvartsman, B., Kirs, M., Pohlak, M. and Herranen, H. Convergence theorem for the Haar wavelet based discretization method. *Compos. Struct.*, 2015, **126**, 227–232.
56. Bougoffa, L. and Rach, R. Solving nonlocal initial-boundary value problems for linear and nonlinear parabolic and hyperbolic partial differential equations by the Adomian decomposition method. *Appl. Math. Comput.*, 2013, **225**, 50–61.
57. Martín-Vaquero, J. and Wade, B. A. On efficient numerical methods for an initial-boundary value problem with nonlocal boundary conditions. *Appl. Math. Model.*, 2012, **36**(8), 3411–3418.

### **Haari lainikute abil leitud mittelokaalsete ääritingimustega diferentsiaalvõrrandite numbrilised lahendid**

Imran Aziz, Muhammad Nisar ja Siraj-ul-Islam

Vaadeldakse elliptiliste, paraboolsete ja hüperboolsete diferentsiaalvõrrandite lahendeid mittelokaalsete ääritingimuste korral. Nende võrrandite numbriliste lahendite leidmisel kasutatakse ruumiliste tuletiste lähendamiseks Haari lainikutel baseeruvat kollokatsioonimeetodit ning ajaliste tuletiste lähendamiseks lõplike vahede meetodit. Sellist meetodit rakendatakse nii lineaarsete kui ka mittelineaarsete probleemide lahendamiseks. Meetodi tööd demonstreeritakse, kasutades mitut kirjandusel baseeruvat näidisülesannet. Täpsuse hindamiseks võrreldakse numbrilist tulemust analüütilise lahendiga. Numbrilised tulemused näitavad, et käesolev meetod on lihtne ja efektiivne.