

Research article

Applications of the nonlinear Klein/Sinh-Gordon equations in modern physics: a numerical study

Interam Ali¹ and Imtiaz Ahmad^{2,*}

¹ Department of Mathematics, Women University Swabi, Swabi 23430, Pakistan

² Institute of Informatics and Computing in Energy (IICE), Universiti Tenaga Nasional (UNITEN), Putrajaya Campus, Kajang 43000, Selangor, Malaysia

* **Correspondence:** Email: imtiaskakakhil@gmail.com.

Abstract: In this article, a hybrid numerical scheme based on Lucas and Fibonacci polynomials in combination with Störmer’s method for the solution of Klein/Sinh-Gordon equations is proposed. Initially, the problem is transformed to a time-discrete form by using Störmer’s technique. Then, with the help of Fibonacci polynomials, we approximate the derivatives of the function. The suggested technique is validated to both one and two-dimensional problems. The resultant findings are compared with existing numerical solutions and presented in a tabular form. The comparison reveals the superior accuracy of the scheme. The numerical convergence of the scheme is computed in each example.

Keywords: Klein/Sinh-Gordon equation; finite differences; Lucas polynomials; Fibonacci polynomials

1. Introduction

Nonlinear evolution equations (NEEs) are mathematical equations that describe the behavior of complex physical systems. These equations are often used in various fields of science, including material and biological sciences, physics, mathematics, and mechanics [1–5]. One of the most commonly used NEE is the nonlinear Schrödinger equation, which is used in quantum mechanics to describe the behavior of wave packets [6, 7]. Another example is the Klein-Gordon equation, which is used in particle physics to describe the behavior of spinless particles. The Cahn-Hilliard equation is another important equation used in materials science to describe the phase separation behaviour of binary mixtures. This equation has applications in the fields of metallurgy, polymer science, and biomaterials. The Navier-Stokes equation is a well-known NEE that describes the behavior of fluids, and is used in fluid mechanics to study the flow of fluids in various applications, including aerospace

engineering, civil engineering, and oceanography [8–11].

NEEs are important in many fields of science because they can provide a mathematical model to describe the behavior of complex physical systems. By studying these equations and developing new numerical techniques to solve them, scientists can gain a better understanding of the physical phenomena they describe, and develop new technologies and materials that are based on this knowledge. To study the behavior of the physical problem (see [12–15]), either analytical or numerical solutions are needed. However, finding an analytical solution for NNEs is not always simple. As a result, developing numerical methods for such problems is useful. In this article, we investigate the nonlinear Klein-Gordon and sinh-Gordon equations, which are given as follows:

$$u_{tt} + \alpha \Delta u + \psi(u) = g(x, t), \quad (1.1)$$

where $u = u(\mathbf{x}, t)$, α is constant, and ψ is some nonlinear

function. For

$$\psi(u) = \sinh(u),$$

the Eq (1.1) becomes the sinh-Gordon equation [16]. When

$$\psi(u) = \beta u + \gamma u^2,$$

the Eq (1.1) becomes the Klein-Gordon equation, which is given as follows:

$$u_{tt} + \alpha \Delta u + \beta u + \gamma u^2 = g(x, t), \quad (1.2)$$

with the following initial and boundary conditions:

$$u(x, 0) = g_1(x), \quad u_t(x, 0) = g_2(x), \quad a \leq x \leq b, \quad (1.3)$$

$$u(a, t) = f_1(t), \quad u(b, t) = f_2(t), \quad t > 0. \quad (1.4)$$

The Klein-Gordon equation has numerous applications in different areas of physics and engineering and plays a crucial role in our understanding of the behavior of particles and fields in the universe. The Klein-Gordon equation is a relativistic wave equation that describes the behavior of particles with a zero spin, such as the Higgs boson particle. It has several applications in different fields of physics; for example, the Klein-Gordon equation is fundamental in the quantum field theory. It describes the behavior of scalar fields, which are fields that have a single value at each point in space-time [17, 18]. Additionally, it can be used to describe the behavior of phonons, which are the quantized vibrations of a crystal lattice, and is used to describe the behavior of the inflaton field, which is thought to be responsible for the rapid expansion of the universe during the Big Bang [19]. Due to the wide range of applications, numerous researchers have investigated the solution of this equation utilizing various methodologies, including collocation points with the thin-plate splines radial basis function [20], the sine-cosine and tanh methods for travelling wave solutions [19], and the generalized auxiliary equation methods in [21]. Pseudospectral formulation with a perfectly matched layer has been suggested for the non-relativistic Klein-Gordon equation in [22]. Three different numerical methods were introduced to solve the rotating Klein-Gordon equation from relativistic regimes to non-relativistic regimes in polar and Cartesian coordinates [23]. Various numerical methods, including finite difference time

domain methods, the time-splitting method, an exponential wave integrator, a limit integrator, a multiscale time integrator, the two-scale formulation method, and an iterative exponential integrator, were compared in terms of their accuracy while solving the Klein-Gordon equation (see [24]). The authors used a finite element technique based on cubic B-splines, the decomposition method [25], the fourth-order implicit Runge-Kutta-Nystrom, the fourth-order compact finite difference method [26], the boundary integral equation method [27], and the cubic B-spline collocation approach [28] to solve the generalized form of the nonlinear Klein-Gordon equation. For the solution of this model, a variety of numerical techniques have been reported in [29, 30]. The two-dimensional sinh-Gordon equation is an essential partial differential equation (PDE) model with applications in a variety of fields.

The following is a concise review of the sinh-Gordon equation. In [31], the (G'/G)-expansion scheme was proposed for the generalized form of the double sinh-Gordon equations. For the analytical solutions to the sinh-Gordon equation, the tanh method was used in [32]. The element-free Galerkin technique was suggested in [33] to solve the generalized sinh-Gordon equation. The radial basis function was utilized by the authors in [34] to solve the sinh-Gordon equation. In [35], the author employed an implicit Lie-group iterative technique for the solution of the sinh-Gordon equation, whereas the Pseudo spectral and Kansa's methods based on radial basis functions were used in [36] to solve the two-dimensional sinh-Gordon equation. In [37], two effective approaches, namely the generalized leap-frog and the method of lines, were utilized for the sinh-Gordon equation, whereas the polynomial differential quadrature method was examined in [38].

In this work, we utilize the Fibonacci polynomials combined with Störmer's method to compute the numerical solutions of the one-dimensional Klein-Gordon and the two-dimensional sinh-Gordon equations. One of the key advantages of the suggested method is the ease implementation in higher-order derivatives using the relation of the Lucas and Fibonacci polynomials. Furthermore, the proposed approach improves the accuracy even for a small number of collocation points, thus lowering the computing costs. These polynomials have a wide range of applications

in differential equations; for instance, the relationship between Chebyshev and Lucas polynomials was discussed in [39], and accurate solutions for the boundary value problems were obtained. Higher-order differential equations were solved using the Lucas polynomial technique in [40], while the Volterra-Fredholm integral differential equations were solved using a Fibonacci polynomial approach in [41]. Delay difference equations were solved using a hybrid Taylor-Lucas polynomial approach [42]. The Fibonacci wavelet approach was implemented for non-linear reaction diffusion equations of a Fisher-type, non-linear Hunter-Saxton and time fractional telegraph equation in [43, 44]. The author and his co-worker used the Gegenbauer wavelet method for time fractional PDEs in [45, 46]. For the first time, the authors proposed a hybrid Lucas and Fibonacci polynomial scheme to solve the time-dependent PDEs in [16, 47]. More publications using Lucas polynomials and finite-differences to find efficient numerical solutions for different types of PDE models can be found in [48–51].

The rest of the paper is organized as follows: in Section 2, fundamental concepts and definitions are discussed; in Section 3, the proposed methodology of the scheme is described; the method is verified with the help of numerical experiment in Section 4, and finally, the concluding remarks are drawn at the end.

2. Fundamental concepts

In this section, we will go over several key definitions and properties related to the Fibonacci and Lucas polynomials.

2.1. Fibonacci polynomials

The Fibonacci polynomial is an extension of Fibonacci numbers described by the linear recurrence relation, which is shown as follows [52]:

$$F_k(x) = kF_{k-1}(x) + F_{k-2}(x), \quad k \geq 2, \quad (2.1)$$

with the initial values

$$F_0(x) = 0 \text{ and } F_1(x) = 1.$$

Their explicit form is as follows:

$$F_k(x) = \sum_{n=0}^{\lfloor \frac{k}{2} \rfloor} \binom{k-n}{n} x^{k-2n},$$

where

$$\binom{m}{n} = \frac{m!}{n!(m-n)!},$$

and $\lfloor k \rfloor$ denotes the integer floor function. Equation (2.1) generates a sequence for $x = 1$ of the Fibonacci numbers.

2.2. Lucas polynomials

The Lucas polynomials can be defined as follows [52]:

$$L_k(x) = kL_{k-1}(x) + L_{k-2}(x), \quad k \geq 2, \quad (2.2)$$

with

$$L_0(x) = 2 \text{ and } L_1(x) = x.$$

Their explicit form is

$$L_k(x) = \sum_{n=0}^{\lfloor \frac{k}{2} \rfloor} \frac{k}{k-n} \binom{k-n}{n} x^{k-2n},$$

when $x = 1$. Equation (2.2) generates a sequence of the Lucas numbers.

2.3. Function approximation

Let $u(x)$ be a continuous function that may be approximated in terms of the Lucas series in the following manner:

$$u(x) = \sum_{k=0}^M \lambda_k^{n+1} L_k(x), \quad (2.3)$$

where $L(x)$ are the Lucas polynomials and λ_k are the unknown coefficients. Utilizing the finite terms of the Lucas series, the m th-order derivative of a function $u(x)$ can be approximated as follows:

$$u^{(m)}(x) = \sum_{k=0}^M \lambda_k^{n+1} L_k^{(m)}(x). \quad (2.4)$$

Utilizing the differentiation matrix \mathbf{D} and Fibonacci polynomials, $L_k^{(m)}(x)$ can be approximated [52] as follows:

$$L_k^{(m)}(x) = kF_k(x)D^{m-1}, \quad (2.5)$$

where

$$D = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & & & \\ \vdots & & d & \\ 0 & & & \end{bmatrix},$$

and d is a $N \times N$ matrix

$$d_{m,n} = \begin{cases} m \sin \frac{(n-m)\pi}{2}, & \text{if } n > m, \\ 0, & \text{otherwise.} \end{cases} \quad (2.6)$$

3. Proposed methodology

To describe the proposed hybrid scheme, we consider Eq (1.1)

$$u_{tt} + \alpha \Delta u + \psi(u) = g(x, t) \tag{3.1}$$

with the following conditions:

$$u(x, 0) = g_1(x), \quad u_t(x, 0) = g_2(x), \quad a \leq x \leq b, \tag{3.2}$$

$$u(a, t) = f_1(t), \quad u(b, t) = f_2(t), \quad t > 0. \tag{3.3}$$

First, re-arrange Eq (3.1)

$$u_{tt} = g(x, t) - \alpha \Delta u - \psi(u). \tag{3.4}$$

Now, using Störmer’s method for the time discretization of Eq (3.4), we have the following:

$$u^{n+1} = 2u^n - u^{n-1} + dt^2(g(x, t) - \alpha \Delta u^n - \psi(u^n)), \tag{3.5}$$

where

$$u^{n+1} = u(t^{n+1}, x),$$

dt is time stepsize for $t \in [0, T]$.

Then, let us discretize the spacial domain $[a, b]$ into the $M + 1$ number of the nodes $x_i, i = 0, 1, 2, \dots, M$, where x_0, x_M are the boundary points while the remaining (i.e., $x_i, i = 1, 2, \dots, M - 1$) are interior points of the domain, which are computed as follows:

$$x_i = a + i * h,$$

where

$$h = (b - a)/M$$

is spatial step size. Here, we approximate the unknown function $u(x)$ by Lucas polynomials at the n th time level, which is denoted as $u^n(x)$:

$$u^n(x) = \sum_{j=0}^M c_j^n(t) L_j(x) = \mathbf{L}^T(x) \mathbf{C}, \tag{3.6}$$

where \mathbf{C} is the vector of unknown coefficients dependent on time, and \mathbf{L} is the $(M + 1) \times 1$ vector of Lucas polynomials, such that

$$\mathbf{L}(x) = [L_0(x), L_1(x), \dots, L_M(x)]^T. \tag{3.7}$$

Now, for the collocation points x_i ,

$$u^n(x_i) = \sum_{j=0}^M c_j^n(t) L_j(x_i) = \mathbf{L}(x_i) \mathbf{C}. \tag{3.8}$$

Let

$$u^n(x_i) = u_i;$$

then, in the matrix notation,

$$\mathbf{U} = \mathbf{A} \mathbf{C}, \tag{3.9}$$

where

$$\mathbf{U} = [u_0, u_1, \dots, u_M]^T, \quad \mathbf{C} = [c_0, c_1, \dots, c_M]^T$$

and

$$\mathbf{A} = \begin{bmatrix} L_0(x) \\ L_1(x) \\ \vdots \\ L_M(x) \end{bmatrix} = \begin{bmatrix} L_0(x_0) & L_0(x_1) & \cdots & L_0(x_M) \\ L_1(x_0) & L_1(x_1) & \cdots & L_1(x_M) \\ \vdots & \vdots & \ddots & \vdots \\ L_M(x_0) & L_M(x_1) & \cdots & L_M(x_M) \end{bmatrix}. \tag{3.10}$$

Similarly, the k th order derivative of unknown functions are approximated by the k th order derivatives of Lucas polynomials

$$u^{(k)}(x_i) = \sum_{j=0}^M c_j^n(t) L_j^{(k)}(x_i) = \mathbf{L}^{(k)}(x_i) \mathbf{C}. \tag{3.11}$$

In the matrix form,

$$\mathbf{U}^{(k)} = \mathbf{B} \mathbf{C}, \tag{3.12}$$

where

$$\mathbf{U}^{(k)} = [u_0^{(k)}, u_1^{(k)}, \dots, u_M^{(k)}]^T, \quad \mathbf{C} = [c_0, c_1, \dots, c_M]^T$$

and

$$\mathbf{B} = \begin{bmatrix} L_0^{(k)}(x) \\ L_1^{(k)}(x) \\ \vdots \\ L_M^{(k)}(x) \end{bmatrix} = \begin{bmatrix} L_0^{(k)}(x_0) & L_0^{(k)}(x_1) & \cdots & L_0^{(k)}(x_M) \\ L_1^{(k)}(x_0) & L_1^{(k)}(x_1) & \cdots & L_1^{(k)}(x_M) \\ \vdots & \vdots & \ddots & \vdots \\ L_M^{(k)}(x_0) & L_M^{(k)}(x_1) & \cdots & L_M^{(k)}(x_M) \end{bmatrix}. \tag{3.13}$$

Using the above approximation in Eq (3.5), we have the following:

$$\mathbf{u}^{n+1} = 2\mathbf{u}^n - \mathbf{u}^{n-1} + dt^2 (\mathbf{g}^n - \alpha \mathbf{D} \mathbf{u}^n - \psi(\mathbf{u}^n)), \tag{3.14}$$

where

$$\mathbf{g}^n = g(x_i, t^n) \quad (i = 0, 1, \dots, M)$$

and

$$\mathbf{D} = \mathbf{BA}^{-1}.$$

For $n=1$, \mathbf{u}^1 and \mathbf{u}^0 can be obtained from the initial conditions. To satisfy the boundary conditions we collocated the boundary conditions that corresponded to each of the following:

$$x_i \in \Gamma = \{a, b\}.$$

That is, for

$$u(x_i, t) = f(x_i, t), \quad \forall x_i \in \Gamma,$$

we then update the solution vector at boundary points to have

$$\mathbf{u}_{\Gamma}^{n+1} = \mathbf{f}(t^{n+1}).$$

Remark 3.1. *The stability condition for the second order Störmer method is as follows:*

$$\alpha \left(\frac{dt}{dx} \right) \leq 1,$$

where α is the coefficient of the Laplace operator. Then the method is then said to be stable, by the rule of thumb, if all the eigenvalues, λ , of the discretized operator,

$$\mathbf{D} = \mathbf{BA}^{-1},$$

scaled by dt^2 lies within the stability region of the time-stepping method [53], i.e.,

$$-4 \leq \Re(\lambda dt^2) \leq 0.$$

4. Numerical results and discussion

This section demonstrates the numerical solution of one dimension Klien-Gordon and two dimension sinh-Gordon equations utilizing the proposed scheme. All the computations are performed using MATLAB (R2012a) on Dell PC Laptop with an Intel(R) Core(TM)i5-2450M CPU 2.50 GHz 2.50 GHz 8 GB RAM. To check the accuracy and convergence, we used the following error norms:

$$L_{\infty} = \max |E_k - u_k|_{k=1}^{M+1},$$

$$L_2 = \left(dx \sum_{k=1}^{M+1} (E_k - u_k)^2 \right)^{1/2},$$

$$L_{rms} = \left(\frac{1}{M} \sum_{k=1}^{M+1} (E_k - u_k)^2 \right)^{1/2},$$

$$C - rate = \frac{L_{\infty}(dt_{\ell})}{L_{\infty}(dt_{\ell+1})},$$

where E is the exact solution.

4.1. (1+1)-dimensional case

Test Problem 4.1. *Consider Eq (1.2) with $\alpha = \beta = -1$, $\gamma = 0$, $g(x, t) = 0$; we have*

$$u_{tt} - \Delta u - u = 0, \tag{4.1}$$

with the following initial and boundary conditions:

$$u(x, 0) = 1 + \sin(x), \quad u_t(x, 0) = 0, \quad x \in [0, 1],$$

$$u(0, t) = \sin(0) + \cosh(t), \quad u(1, t) = \sin(1) + \cosh(t), \quad t > 0,$$

and the actual solution is as follows:

$$u(x, t) = \sin(x) + \cosh(t).$$

Using the proposed methodology for this particular equation the iterative scheme (3.14) becomes the following:

$$\mathbf{u}^{n+1} = 2\mathbf{u}^n - \mathbf{u}^{n-1} + dt^2(\mathbf{g}^n + \mathbf{D}\mathbf{u}^n + \mathbf{u}^n). \tag{4.2}$$

The above equation is used to generate numerical results, which are provided in Table 1 in terms of the L_{∞} and L_2 error norms. These findings are obtained using nodal points $M = 10$ and various values of the time step size dt while preserving the final time $T = 0.1$, and for various values of T and fixing $dt = 0.0001$. The accuracy increases as the value of dt decreases, as shown in this table. Based on the comparison with [29], it is obvious that the suggested scheme is more accurate than those reported in the recent literature. The numerical results in terms of L_{∞} , L_2 , and L_{rms} , as well as central processing unit (CPU) times in seconds, are computed using the suggested approach for various values of M in Table 2. The table illustrates that as the number of collocation points rises, the current approach provides an improved accuracy. Furthermore, the proposed technique may be seen to be quite efficient, whereas the comparison between the exact and approximate solutions are visualized in Figure 1.

Table 1. Results of the proposed method for Test Problem 4.1 with $M = 10$ and $T = 0.1$.

dt	Present method		MQ-RBF [29]		GA-RBF [29]	
	L_∞	L_2	L_∞	L_2	L_∞	L_2
0.01	5.65E-04	4.92E-04	5.06E-04	4.46E-04	1.36E-03	1.26E-03
0.001	5.50E-05	4.86E-05	5.55E-05	4.74E-05	5.08E-05	4.43E-05
0.0005	2.75E-05	2.43E-05	4.04E-05	2.61E-05	2.57E-05	2.21E-05
0.0001	5.49E-06	4.85E-06	2.83E-05	1.12E-05	6.80E-06	5.05E-06
T	L_∞	L_2	L_∞	L_2	L_∞	L_2
0.01	5.00E-07	4.78E-07	7.57E-07	-	5.35E-07	-
0.02	1.00E-06	9.49E-07	2.02E-06	-	1.13E-06	-
0.1	5.49E-06	4.85E-06	2.83E-05	-	6.80E-06	-
0.5	4.19E-05	3.42E-05	8.61E-05	-	4.63E-05	-
1	4.37E-04	2.14E-04	3.95E-05	-	4.20E-04	-

Table 2. Results of the proposed method for Test Problem 4.1 with $dt = 0.0001$ and $T = 0.1$.

M	L_∞	L_2	L_{rms}	CPU
5	5.03E-06	4.45E-06	3.15E-07	0.037078
7	4.32E-06	3.73E-06	2.95E-07	0.045391
10	2.49E-06	1.85E-06	1.85E-07	0.051433
15	9.03E-07	8.56E-07	9.81E-08	0.071921
18	7.12E-07	6.42E-07	7.88E-08	0.109775
20	5.31E-07	5.13E-07	6.36E-08	0.155763
22	2.90E-07	1.65E-07	3.38E-08	0.330819
25	2.35E-07	1.09E-07	1.47E-08	1.366762

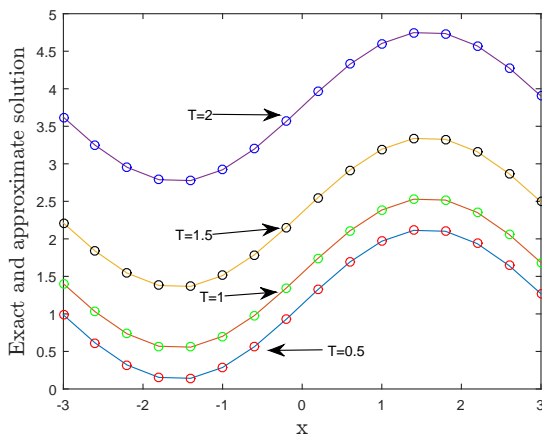


Figure 1. Comparison of exact and approximate solutions for Test Problem 4.1.

Test Problem 4.2. Now assume

$$\alpha = -1, \beta = -2, \gamma = 0$$

and

$$g(x, t) = -2 \sin x \sin t$$

in Eq (1.2) with the following conditions:

$$u(x, 0) = 0, \quad u_t(x, 0) = \sin x, \quad x \in [0, \pi/2],$$

$$u(0, t) = 0, \quad u(\pi/2, t) = \sin t, \quad t > 0.$$

The exact solution of Eq (1.2) for the above parameter is given in [29], which is

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

In Test Problem 4.2, the numerical results are computed utilizing the proposed technique. These results are shown in Table 3 using nodal points

$$M = 10$$

and various values of dt while preserving the final time

$$T = 0.1,$$

and for various values of T and fixing

$$dt = 0.0001.$$

In this situation, as indicated in the table, the method's accuracy improves as the value of dt declines. Furthermore, in comparison with [29], we came to know that the current method is more accurate than other articles published in the recent literature. Similarly to the previous test problem, the numerical results are computed in the form of L_∞ , L_2 , and L_{rms} , as well as the CPU times in seconds, using the suggested approach for various values of M in Table 4. As the number of collocation points increases, the current approach efficiently improves the accuracy, as seen in the table whereas profiles of the exact and approximate solutions are given in Figure 2.

Table 3. Results of the proposed method for Test Problem 4.2 with $M = 10$ and $T = 0.1$.

dt	Present method		MQ-RBF [29]		GA-RBF [29]	
	L_∞	L_2	L_∞	L_2	L_∞	L_2
0.01	1.59E-06	1.23E-03	2.50E-03	1.04E-03	2.44E-03	9.86E-04
0.001	1.59E-08	1.23E-04	1.53E-04	6.58E-05	5.05E-04	2.09E-04
0.0005	3.97E-09	6.17E-05	7.65E-05	3.24E-05	3.87E-04	1.63E-04
0.0001	1.57E-10	1.23E-05	1.50E-05	2.59E-05	2.92E-04	1.26E-04
T	L_∞	L_2	L_∞	L_2	L_∞	L_2
0.01	1.65E-11	3.91E-06	1.58E-07	-	5.45E-07	-
0.02	3.29E-11	5.53E-06	6.29E-07	-	3.32E-06	-
0.1	1.57E-10	1.23E-05	1.50E-05	-	2.92E-04	-
0.5	6.47E-10	2.27E-05	8.84E-05	-	1.07E-02	-
1	1.16E-09	1.39E-05	5.20E-05	-	2.34E-02	-

Table 4. Results of the proposed method for Test Problem 4.2 with $dt = 0.0001$ and $T = 1$.

M	L_∞	L_2	L_{rms}	CPU
5	4.23E-04	3.32E-03	1.87E-04	0.051816
7	2.99E-05	2.90E-04	3.65E-05	0.051183
10	9.76E-06	8.90E-05	7.09E-06	0.082319
15	6.51E-06	2.40E-05	2.34E-06	0.09778
18	1.64E-06	2.73E-05	6.92E-07	0.143512
20	8.01E-07	8.93E-06	1.01E-07	0.184772
22	9.55E-08	5.80E-06	8.68E-08	0.338581
25	7.12E-08	7.42E-07	7.70E-08	1.288726

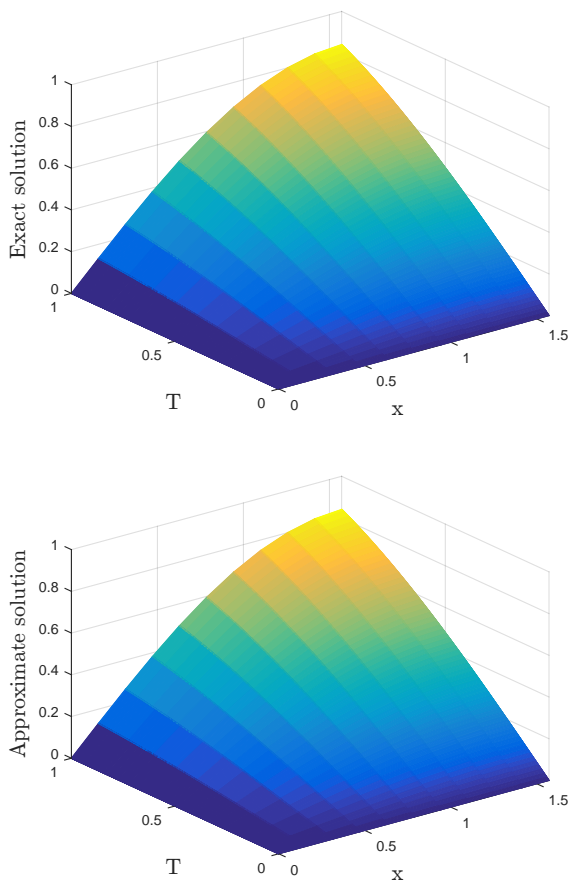


Figure 2. Exact versus approximate solution for Test Problem 4.2.

Test Problem 4.3. Consider Eq (1.2) with the exact solution $u(x, t) = x^3 t^3$, when

$$\alpha = -1, \beta = 0, \gamma = 1$$

and

$$g(x, t) = 6xt(x^2 - t^2) + x^6 t^6,$$

with associated conditions

$$\begin{aligned} u(x, 0) &= 0, & u_t(x, 0) &= 0, & x &\in [0, 1], \\ u(0, t) &= 0, & u(1, t) &= 0, & t &> 0. \end{aligned}$$

Tables 5 and 6 show the numerical results for Test Problem 4.3 for various values of dt and M , respectively. We compared our results with the methods in [29] for

$$M = 10 \text{ and } T = 1$$

in Table 5. We can see from this table that the method's accuracy improves as the number of iterations increases, with a very good accuracy of 10^{-8} for

$$dt = 0.0001.$$

In comparison to the method in [29], the proposed method produces more accurate results. In Table 6, the results are computed in terms of L_∞ , L_2 , and L_{rms} for

$$dt = 0.001$$

and time $T = 1$, as well as the CPU time in seconds. The suggested procedure is accurate and efficient, as shown in this table. The profile of the exact and approximate solutions are shown in Figures 3 and 4, respectively.

Table 5. Results of the proposed method for Test Problem 4.3 with $M = 10$ and $T = 1$.

dt	Present method		MQ-RBF [29]		GA-RBF [29]	
	L_∞	L_2	L_∞	L_2	L_∞	L_2
0.01	3.35E-04	1.78E-04	2.24E-02	9.59E-03	4.66E-02	1.96E-02
0.001	3.24E-06	1.64E-06	2.46E-03	1.05E-03	2.27E-03	9.71E-04
0.0005	8.08E-07	4.09E-07	1.36E-03	5.79E-04	1.17E-03	5.00E-04
0.0001	3.17E-08	1.66E-08	4.85E-04	2.07E-04	2.94E-04	1.25E-04

Table 6. Results of the proposed method for Test Problem 4.3 with $T = 1$ and $dt = 0.001$.

M	L_∞	L_2	L_{rms}	CPU
5	2.19E-04	1.37E-04	9.65E-05	0.050013
10	3.24E-05	6.36E-06	1.64E-06	0.102678
15	4.84E-06	4.88E-06	6.82E-07	0.088209
20	1.07E-06	1.3606E-06	1.92E-07	0.185567
22	8.11E-07	5.05E-07	7.13E-08	0.370034
25	6.08E-07	2.72E-07	4.21E-08	1.372918

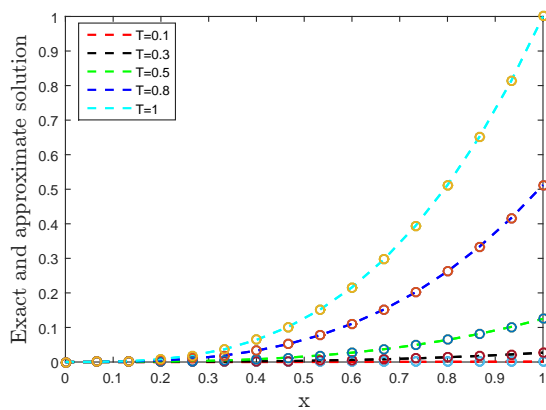


Figure 3. Comparison of exact and approximate solutions for Test Problem 4.3.

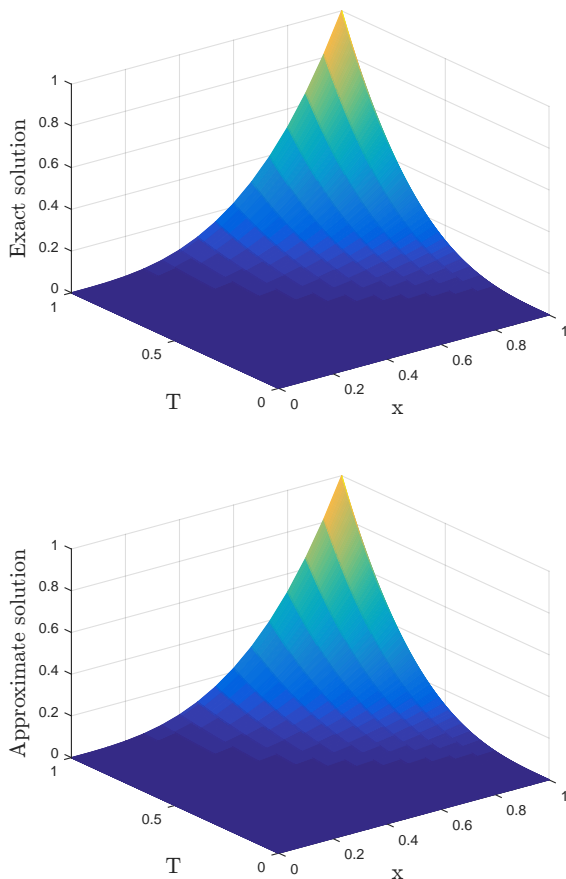


Figure 4. Exact versus approximate solution for Test Problem 4.3.

Test Problem 4.4. Consider

$$\alpha = -1, \beta = -1, \gamma = 0$$

and

$$g(x, t) = -\cos x \sin t$$

in Eq (1.2) with the following conditions:

$$u(x, 0) = 0, \quad u_t(x, 0) = \cos x, \quad x \in [0, 1],$$

$$u(0, t) = \sin t, \quad u(1, t) = \cos 1 \sin t, \quad t > 0.$$

The exact solution of Eq (1.2) for the above parameter given as follows:

$$u(x, t) = \cos x \sin t.$$

The solution is computed for the fixed point $T = 0.1$ and $M = 20$, while decreasing the time step. The results are reported in Table 7, which shows that the accuracy improves as the number of iteration increases. In Table 8, the results are recorded for various numbers of nodal points M , and notice that as number of collocation point increases, the numerical convergence accuracy improves. Along with, maximum error, the CPU time in seconds is also reported in the tables. The solution profiles of the exact and approximate solutions are illustrated in Figure 5, which shows that both the solutions are well matched.

Table 7. Results of the proposed method for Test Problem 4.4 with $M = 15$ and $T = 0.1$.

dt	L_∞	L_2	L_{rms}	CPU
0.01	4.28E-05	3.55E-05	4.79E-05	2.03E-01
0.001	4.88E-06	4.05E-06	1.80E-06	2.04E-01
0.0005	2.46E-06	2.04E-06	6.45E-07	1.77E-01
0.0001	4.97E-07	4.23E-07	5.98E-08	1.92E-01

Table 8. Results of the proposed method for Test Problem 4.4 with $dt = 0.0001$ and $T = 0.1$.

M	L_∞	L_2	L_{rms}	CPU
5	4.97E-07	4.88E-07	3.74E-08	0.04553
7	4.80E-07	3.94E-07	3.53E-08	0.048321
10	4.90E-07	3.76E-07	3.40E-08	0.063045
15	4.64E-07	3.66E-07	3.39E-08	0.081496
18	4.54E-07	3.68E-07	3.36E-08	0.115411
20	4.40E-07	3.43E-07	3.33E-08	0.167235
22	4.36E-07	3.36E-07	3.32E-08	0.372988
25	4.24E-07	3.18E-07	2.82E-08	1.378855

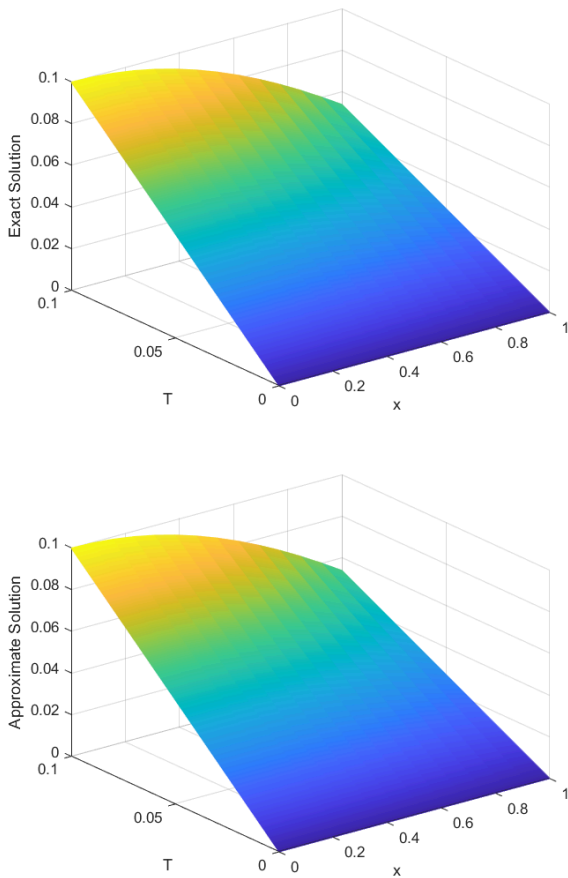


Figure 5. Exact versus approximate solution for Test Problem 4.4.

Test Problem 4.5. (1+2)-dimensional case: Finally, consider the two dimensional sinh-Gordon equation

$$u_{tt} - \Delta u + \sinh(u) = g(x, y, t) \tag{4.3}$$

with initial conditions

$$u(x, y, 0) = u_t(x, y, 0) = \sin(x + y), \quad (x, y) \in [0, 1]^2,$$

and boundary conditions

$$\begin{aligned} u(0, y, t) &= e^t \sin(0 + y), & u(1, y, t) &= e^t \sin(1 + y), & t > 0, \\ u(x, 0, t) &= e^t \sin(x + 0), & u(x, 1, t) &= e^t \sin(x + 1). \end{aligned}$$

The exact solution

$$u(x, y, t) = e^t \sin(x + y)$$

is considered, as given in [47]. Additionally the source term is as follows:

$$g(x, y, t) = 3e^t \sin(x + y) + \sinh(e^t \sin(x + y)). \tag{4.4}$$

The results of the suggested method for Test Problem 4.5 are shown in Table 9 for various values of dt with $M = 4$ and $T = 1$. This table shows that the method delivers a good accuracy for very coarse nodes, and that the method's accuracy improves to some extent as the number of iterations increases. Additionally in this table, we also compared the computed results to the approach in [47]. The results of the proposed approach in contrast to the method presented in [47] are shown in Table 10 for various values of M for the same test problem. This comparison reveals that the proposed approach is more accurate than [47]. The approximate solution of the suggested approach is compared to the exact solution in Figure 6. Although the absolute errors are indicated in Figure 7, the numerical solution is in a good agreement with the exact solution, as demonstrated in Figure 6.

Table 9. Results of the proposed method for Test Problem 4.5 with $M = 4$ and $T = 1$.

dt	Present method			[47]	
	L_∞	L_2	C-rate(L_∞)	L_∞	L_2
0.100	1.91E-02	8.66E-03	-	2.20E-02	4.19E-02
0.050	9.19E-03	3.93E-03	2.08E+00	1.15E-02	1.47E-02
0.025	4.52E-03	1.93E-03	2.03E+00	7.53E-03	1.23E-02
0.013	2.24E-03	9.51E-04	2.02E+00	4.88E-03	9.57E-03
0.006	1.11E-03	4.68E-04	2.02E+00	2.90E-03	6.04E-03
0.003	5.45E-04	2.27E-04	2.04E+00	1.60E-03	3.40E-03
0.002	2.63E-04	1.08E-04	2.07E+00	8.49E-04	1.80E-03
0.001	1.22E-04	5.13E-05	2.15E+00	4.43E-04	9.30E-04

Table 10. Results of the proposed method for Test Problem 4.5 with $dt = 0.001$ and $T = 1$.

M	Present method		[47]	
	L_∞	L_2	L_∞	L_2
2	5.85E-03	2.93E-03	5.53E-03	5.53E-03
3	2.22E-03	1.47E-03	1.22E-03	1.74E-03
4	1.62E-04	6.66E-05	5.59E-04	1.18E-03
5	1.63E-04	6.75E-05	5.38E-04	9.99E-04
6	1.23E-04	6.37E-05	-	-
7	1.32E-04	7.41E-05	-	-

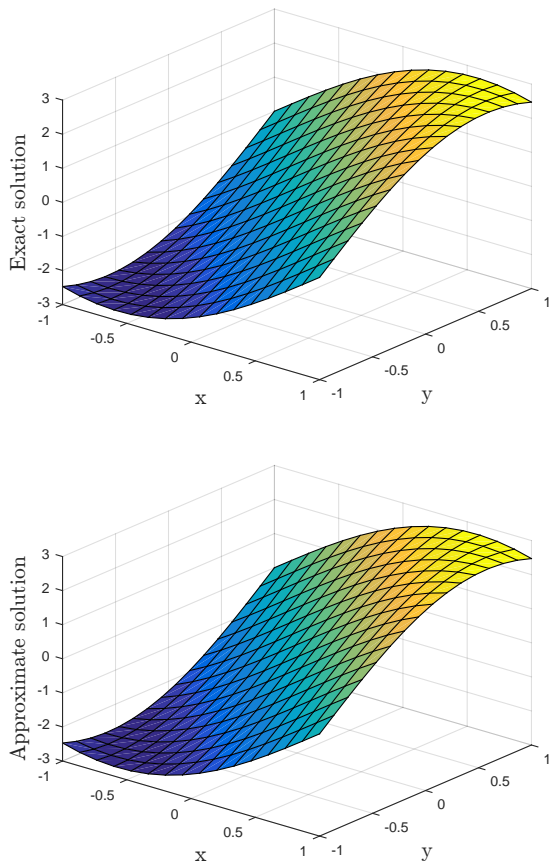


Figure 6. Exact versus approximate solution at $M = 15$, $T = 1$, for Test Problem 4.5.

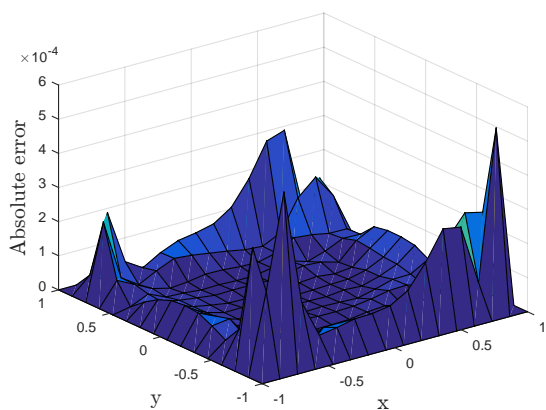


Figure 7. Error for Test Problem 4.5.

5. Conclusions

A numerical method based on Fibonacci polynomials combined with Störmer's method was developed to solve

the nonlinear Klein/Sinh-Gordon equation. On a variety of test cases, numerical evaluation exhibited an improved performance of the proposed scheme against the available results in the literature. As evidenced by the comparison, the proposed technique is simple to execute and faster in convergence. Furthermore, it was observed that by reducing the spatial and time steps size, the errors were reduced. With slight modifications, the proposed method can be applied to a variety of PDEs. This modification is mainly dictated by the problem at hand with the associated initial and boundary conditions that are dictated by the particular problem of interest. As such, we are interested in applying the proposed method to time-dependent Neumann and/or mixed boundary value problems with possible discontinuous initial condition(s). The limitations, as is the case in spectral methods, was the high-condition numbers of the collocated/system matrices. However, this can be alleviated by the state-of-the-art pre-conditioned algorithms developed for spectral methods. In the future, the proposed method can be expanded for applications to complex fractal-fractional problems.

Conflict of interest

The authors declare that there are no conflicts of interest in this paper.

References

1. N. H. Sweilam, M. M. Khader, M. Adel, On the stability analysis of weighted average finite difference methods for fractional wave equation, *Fract. Differ. Calc.*, **2** (2012), 17–29. <https://doi.org/10.7153/fdc-02-02>
2. M. Adel, M. Khader, H. Ahmad, T. Assiri, Approximate analytical solutions for the blood ethanol concentration system and predator-prey equations by using variational iteration method, *AIMS Math.*, **8** (2023), 19083–19096. <https://doi.org/10.3934/math.2023974>
3. Siraj-ul-Islam, I. Ahmad, Local meshless method for PDEs arising from models of wound healing, *Appl. Math. Model.*, **48** (2017), 688–710. <https://doi.org/10.1016/j.apm.2017.04.015>

4. M. Adel, M. M. Khader, S. Algelany, High-dimensional chaotic lorenz system: numerical treatment using changhee polynomials of the appell type, *Fractal Fract.*, **7** (2023), 398. <https://doi.org/10.3390/fractalfract7050398>
5. M. Adel, M. E. Ramadan, H. Ahmad, T. Botmart, Sobolev-type nonlinear Hilfer fractional stochastic differential equations with noninstantaneous impulsive, *AIMS Math.*, **7** (2022), 20105–20125. <https://doi.org/10.3934/math.20221100>
6. X. Liu, M. Ahsan, M. Ahmad, M. Nisar, X. Liu, I. Ahmad, et al., Applications of Haar wavelet-finite difference hybrid method and its convergence for hyperbolic nonlinear Schrödinger equation with energy and mass conversion, *Energies*, **14** (2021), 7831. <https://doi.org/10.3390/en14237831>
7. M. Ahsan, I. Ahmad, M. Ahmad, I. Hussian, A numerical Haar wavelet-finite difference hybrid method for linear and non-linear Schrödinger equation, *Math. Comput. Simul.*, **165** (2019), 13–25. <https://doi.org/10.1016/j.matcom.2019.02.011>
8. I. Ahmad, M. Ahsan, I. Hussain, P. Kumam, W. Kumam, Numerical simulation of PDEs by local meshless differential quadrature collocation method, *Symmetry*, **11** (2019), 394. <https://doi.org/10.3390/sym11030394>
9. F. Wang, J. Zhang, I. Ahmad, A. Farooq, H. Ahmad, A novel meshfree strategy for a viscous wave equation with variable coefficients, *Front. Phys.*, **9** (2021), 701512. <https://doi.org/10.3389/fphy.2021.701512>
10. M. Ahsan, S. Lin, M. Ahmad, M. Nisar, I. Ahmad, H. Ahmed, et al., A Haar wavelet-based scheme for finding the control parameter in nonlinear inverse heat conduction equation, *Open Phys.*, **19** (2021), 722–734. <https://doi.org/10.1515/phys-2021-0080>
11. P. Thounthong, M. N. Khan, I. Hussain, I. Ahmad, P. Kumam, Symmetric radial basis function method for simulation of elliptic partial differential equations, *Mathematics*, **6** (2018), 327. <https://doi.org/10.3390/math6120327>
12. J. F. Li, I. Ahmad, H. Ahmad, D. Shah, Y. M. Chu, P. Thounthong, et al., Numerical solution of two-term time-fractional PDE models arising in mathematical physics using local meshless method, *Open Phys.*, **18** (2020), 1063–1072. <https://doi.org/10.1515/phys-2020-0222>
13. I. Ahmad, M. Riaz, M. Ayaz, M. Arif, S. Islam, P. Kumam, Numerical simulation of partial differential equations via local meshless method, *Symmetry*, **11** (2019), 257. <https://doi.org/10.3390/sym11020257>
14. I. Ahmad, I. Ali, R. Jan, S. A. Idris, M. Mousa, Solutions of a three-dimensional multi-term fractional anomalous solute transport model for contamination in groundwater, *Plos One*, **18** (2023), e0294348. <https://doi.org/10.1371/journal.pone.0294348>
15. I. Ahmad, Siraj-ul-Islam, Mehnaz, S. Zaman, Local meshless differential quadrature collocation method for time-fractional PDEs, *Discrete Cont. Dyn. Syst.*, **13** (2020), 2641–2654. <https://doi.org/10.3934/dcds.2020223>
16. Ö. Oruç, A new algorithm based on Lucas polynomials for approximate solution of 1D and 2D nonlinear generalized Benjamin-Bona-Mahony-Burgers equation, *Comput. Math. Appl.*, **74** (2017), 3042–3057. <https://doi.org/10.1016/j.camwa.2017.07.046>
17. I. Ahmad, A. A. Bakar, R. Jan, S. Yussof, Dynamic behaviors of a modified computer virus model: insights into parameters and network attributes, *Alex. Eng. J.*, **103** (2024), 266–277. <https://doi.org/10.1016/j.aej.2024.06.009>
18. H. Ahmad, T. A. Khan, I. Ahmad, P. S. Stanimirović, Y. M. Chu, A new analyzing technique for nonlinear time fractional Cauchy reaction-diffusion model equations, *Results Phys.*, **19** (2020), 103462. <https://doi.org/10.1016/j.rinp.2020.103462>
19. A. M. Wazwaz, The tanh and the Sine-Cosine methods for compact and noncompact solutions of the nonlinear Klein-Gordon equation, *Appl. Math. Comput.*, **167** (2005), 1179–1195. <https://doi.org/10.1016/j.amc.2004.08.006>
20. M. Dehghan, A. Shokri, Numerical solution of the nonlinear Klein-Gordon equation using radial basis functions, *Comput. Appl. Math.*, **230** (2009), 400–410. <https://doi.org/10.1016/j.cam.2008.12.011>

21. E. Yomba, A generalized auxiliary equation method and its application to nonlinear Klein-Gordon and generalized nonlinear Camassa-Holm equations, *Phys. Lett. A*, **372** (2008), 1048–1060. <https://doi.org/10.1016/j.physleta.2007.09.003>
22. X. Antoine, X. Zhao, Pseudospectral methods with PML for nonlinear Klein-Gordon equations in classical and non-relativistic regimes, *J. Comput. Phys.*, **448** (2022), 110728. <https://doi.org/10.1016/j.jcp.2021.110728>
23. N. J. Mauser, Y. Zhang, X. Zhao, On the rotating nonlinear Klein-Gordon equation: nonrelativistic limit and numerical methods, *Multiscale Model. Simul.*, **18** (2020), 999–1024. <https://doi.org/10.1137/18M1233509>
24. W. Bao, X. Zhao, Comparison of numerical methods for the nonlinear Klein-Gordon equation in the nonrelativistic limit regime, *J. Comput. Phys.*, **398** (2019), 108886. <https://doi.org/10.1016/j.jcp.2019.108886>
25. S. M. El-Sayed, The decomposition method for studying the Klein-Gordon equation, *Chaos Solitons Fract.*, **18** (2003), 1025–1030. [https://doi.org/10.1016/S0960-0779\(02\)00647-1](https://doi.org/10.1016/S0960-0779(02)00647-1)
26. M. Dehghan, A. Mohebbi, Z. Asgari, Fourth-order compact solution of the nonlinear Klein-Gordon equation, *Numer. Algorithms*, **52** (2009), 523–540. <https://doi.org/10.1007/s11075-009-9296-x>
27. N. Wang, M. Li, C. Huang, Unconditional energy dissipation and error estimates of the SAV Fourier spectral method for nonlinear fractional generalized wave equation, *J. Sci. Comput.*, **88** (2021), 19. <https://doi.org/10.1007/s10915-021-01534-8>
28. S. A. Khuri, A. Sayfy, A spline collocation approach for the numerical solution of a generalized nonlinear Klein-Gordon equation, *Appl. Math. Comput.*, **216** (2010), 1047–1056. <https://doi.org/10.1016/j.amc.2010.01.122>
29. A. Hussain, S. Haq, M. Uddin, Numerical solution of Klein-Gordon and sine-Gordon equations by meshless method of lines, *Eng. Anal. Bound. Elem.*, **37** (2013), 1351–1366. <https://doi.org/10.1016/j.enganabound.2013.07.001>
30. I. Ahmad, M. Ahsan, I. Hussain, P. Kumam, W. Kumam, Numerical simulation of PDEs by local meshless differential quadrature collocation method, *Symmetry*, **11** (2019), 394. <https://doi.org/10.3390/sym11030394>
31. H. Kheiri, A. Jabbari, Exact solutions for the double sinh-Gordon and generalized form of the double sinh-Gordon equations by using (G'/G)-expansion method, *Turk. J. Phys.*, **34** (2011), 73–82. <https://doi.org/10.3906/fiz-0909-7>
32. A. M. Wazwaz, Exact solutions for the generalized sinh-Gordon and the generalized sinh-Gordon equations, *Chaos Solitons Fract.*, **28** (2006), 127–135. <https://doi.org/10.1016/j.chaos.2005.05.017>
33. X. Li, S. Zhang, Y. Wang, H. Chen, Analysis and application of the element-free galerkin method for nonlinear sinh-Gordon and generalized sinh-Gordon equations, *Comput. Math. Appl.*, **71** (2016), 1655–1678. <https://doi.org/10.1016/j.camwa.2016.03.007>
34. M. Dehghan, A. Shokri, A numerical method for solution of the two-dimensional sine-Gordon equation using the radial basis functions, *Math. Comput. Simul.*, **79** (2008), 700–715. <https://doi.org/10.1016/j.matcom.2008.04.018>
35. C. W. Chang, C. S. Liu, An implicit Lie-group iterative scheme for solving the nonlinear Klein-Gordon and sine-Gordon equations, *Appl. Math. Modell.*, **40** (2016), 1157–1167. <https://doi.org/10.1016/j.apm.2015.06.028>
36. M. Dehghan, M. Abbaszadeh, A. Mohebbi, The numerical solution of the two-dimensional sinh-Gordon equation via three meshless methods, *Eng. Anal. Bound. Elem.*, **51** (2015), 220–235. <https://doi.org/10.1016/j.enganabound.2014.10.015>
37. P. L. Christiansen, P. S. Lomdahl, Numerical study of 2+1 dimensional sine-Gordon solitons, *Phys. D*, **2** (1981), 482–494. [https://doi.org/10.1016/0167-2789\(81\)90023-3](https://doi.org/10.1016/0167-2789(81)90023-3)
38. R. Jiwari, S. Pandit, R. Mittal, Numerical simulation of two-dimensional sine-Gordon solitons by differential quadrature method, *Comput. Phys. Commun.*, **183** (2012), 600–616. <https://doi.org/10.1016/j.cpc.2011.12.004>

39. W. Abd-Elhameed, Y. Youssri, Connection formulae between generalized Lucas polynomials and some Jacobi polynomials: application to certain types of fourth-order BVPs, *Int. J. Appl. Comput. Math.*, **6** (2020), 45. <https://doi.org/10.1007/s40819-020-0799-4>
40. M. Çetin, M. Sezer, C. Güler, Lucas polynomial approach for system of high-order linear differential equations and residual error estimation, *Math. Probl. Eng.*, **2015** (2015), 625984. <https://doi.org/10.1155/2015/625984>
41. F. Mirzaee, S. F. Hoseini, Application of Fibonacci collocation method for solving Volterra-Fredholm integral equations, *Appl. Math. Comput.*, **273** (2016), 637–644. <https://doi.org/10.1016/j.amc.2015.10.035>
42. N. Bayku, M. Sezer, Hybrid Taylor-Lucas collocation method for numerical solution of high-order Pantograph type delay differential equations with variables delays, *Appl. Math. Inf. Sci.*, **11** (2017), 1795–1801. <https://doi.org/10.18576/amis/110627>
43. N. A. Nayied, F. A. Shah, M. A. Khanday, Fibonacci wavelet method for the numerical solution of nonlinear reaction-diffusion equations of Fisher-type, *J. Math.*, **2023** (2023), 1705607. <https://doi.org/10.1155/2023/1705607>
44. F. A. Shah, M. Irfan, K. S. Nisar, R. T. Matoog, E. E. Mahmoud, Fibonacci wavelet method for solving time-fractional telegraph equations with dirichlet boundary conditions, *Results Phys.*, **24** (2021), 104123. <https://doi.org/10.1016/j.rinp.2021.104123>
45. H. M. Srivastava, F. A. Shah, R. Abass, An application of the gegenbauer wavelet method for the numerical solution of the fractional Bagley-Torvik equation, *Russ. J. Math. Phys.*, **26** (2019), 77–93. <https://doi.org/10.1134/S1061920819010096>
46. K. S. Nisar, F. A. Shah, A numerical scheme based on gegenbauer wavelets for solving a class of relaxation-oscillation equations of fractional order, *Math. Sci.*, **17** (2023), 233–245. <https://doi.org/10.1007/s40096-022-00465-1>
47. Ö. Oruç, A new numerical treatment based on Lucas polynomials for 1D and 2D sinh-Gordon equation, *Commun. Nonlinear Sci. Numer. Simul.*, **57** (2018), 14–25. <https://doi.org/10.1016/j.cnsns.2017.09.006>
48. I. Ali, S. Haq, K. S. Nisar, D. Baleanu, An efficient numerical scheme based on Lucas polynomials for the study of multidimensional Burgers-type equations, *Adv. Differ. Equations*, **2021** (2021), 43. <https://doi.org/10.1186/s13662-020-03160-4>
49. S. Haq, I. Ali, Approximate solution of two-dimensional Sobolev equation using a mixed Lucas and Fibonacci polynomials, *Eng. Comput.*, **38** (2021), 2059–2068. <https://doi.org/10.1007/s00366-021-01327-5>
50. S. Haq, I. Ali, K. S. Nisar, A computational study of two-dimensional reaction-diffusion Brusselator system with applications in chemical processes, *Alex. Eng. J.*, **60** (2021), 4381–4392. <https://doi.org/10.1016/j.aej.2021.02.064>
51. I. Ahmad, A. A. Bakar, I. Ali, S. Haq, S. Yussof, A. H. Ali, Computational analysis of time-fractional models in energy infrastructure applications, *Alex. Eng. J.*, **82** (2023), 426–436. <https://doi.org/10.1016/j.aej.2023.09.057>
52. I. Ali, S. Haq, S. F. Aldosary, K. S. Nisar, F. Ahmad, Numerical solution of one-and two-dimensional time-fractional Burgers equation via lucas polynomials coupled with finite difference method, *Alex. Eng. J.*, **61** (2022), 6077–6087. <https://doi.org/10.1016/j.aej.2021.11.032>
53. J. S. Hesthaven, S. Gottlieb, D. Gottlieb, *Spectral methods for time-dependent problems*, Cambridge University Press, 2007. <https://doi.org/10.1017/CBO9780511618352>



AIMS Press

© 2024 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)