Numerical solution of the nonlinear Klein–Gordon equation using radial basis functions

Mehdi Dehghan *, Ali Shokri

Department of Applied Mathematics, Faculty of Mathematics and Computer Science, Amirkabir University of Technology, No. 424, Hafez Ave., Tehran, Iran

A R T I C L E   I N F O

Article history:
Received 15 April 2007
Received in revised form 10 December 2008

Keywords:
Nonlinear Klein–Gordon equation
Collocation
Radial basis functions (RBF)
Thin plate splines (TPS)

A B S T R A C T

The nonlinear Klein–Gordon equation is used to model many nonlinear phenomena. In this paper, we propose a numerical scheme to solve the one-dimensional nonlinear Klein–Gordon equation with quadratic and cubic nonlinearity. Our scheme uses the collocation points and approximates the solution using Thin Plate Splines (TPS) radial basis functions (RBF). The implementation of the method is simple as finite difference methods. The results of numerical experiments are presented, and are compared with analytical solutions to confirm the good accuracy of the presented scheme.

© 2008 Elsevier B.V. All rights reserved.

1. Introduction

Nonlinear phenomena, that appear in many areas of scientific fields such as solid state physics, plasma physics, fluid dynamics, mathematical biology and chemical kinetics, can be modeled by partial differential equations. A broad class of analytical solution methods and numerical solution methods were used to handle these problems.

The initial-value problem of the one-dimensional nonlinear Klein–Gordon equation is given by the following equation:

\[ u_{tt} + \alpha u_{xx} + g(u) = f(x, t), \]

(1.1)

where \( u = u(x, t) \) represents the wave displacement at position \( x \) and time \( t \), \( \alpha \) is a known constant and \( g(u) \) is the nonlinear force. In the well-known sine-Gordon equation, the nonlinear force is given by \( g(u) = \sin u \). In the physical applications, the nonlinear force \( g(u) \) has also other forms [1]. The cases \( g(u) = \sin u + \sin 2u \) and \( g(u) = \sinh u + \sinh 2u \) are called the double sine-Gordon equation and the double sinh-Gordon equation, respectively. The above nonlinear Klein–Gordon equations are Hamiltonian partial differential equations, and for a wide class of force \( g(u) \), it has the conserved Hamiltonian quantity (or energy)

\[ H = \int \left( \frac{1}{2} u_t^2 + \frac{1}{2} u_x^2 + G(u) \right), \]

where \( G'(u) = g(u) \).

In the present work we are dealing with the numerical approximation of the following nonlinear Klein–Gordon equation with quadratic nonlinearity

\[ u_{tt} + \alpha u_{xx} + \beta u + \gamma u^2 = f(x, t), \]

(1.2)
and with cubic nonlinearity
\[ u_{tt} + \alpha u_{xx} + \beta u + \gamma u^3 = f(x, t), \] (1.3)
in which \( \alpha, \beta \) and \( \gamma \) are known constants.

The nonlinear Klein–Gordon equation appears in many types of nonlinearities. The Klein–Gordon equation plays a significant role in many scientific applications such as solid state physics, nonlinear optics and quantum field theory [2]. As is mentioned in [2] many powerful methods, such as the inverse scattering method, Bäcklund transformation, the auxiliary equation method [3,4], the Wadati trace method, Hirota bilinear forms, pseudo-spectral method, the tanh–sech method, the sine–cosine method, Jacobi elliptic functions, and the Riccati equation expansion method were used to investigate these types of equations (see [2] and references therein).

The Klein–Gordon equation plays an important role in mathematical physics [5,6]. The equation has attracted much attention in studying solitons and condensed matter physics [7], in investigating the interaction of solitons in a collisionless plasma, the recurrence of initial states, and in examining the nonlinear wave equations [8].

Parkes et al. [9] and Fu et al. [10] used the Jacobi elliptic function expansion method to find double periodic solutions to Eqs. (1.2) and (1.3), respectively. Several methods are developed to solve the Klein–Gordon-type equations, such as the Weierstrass elliptic function method, the elliptic equation rational expansion method and the extended F-function method (see [3] and references therein). Sirendaoreji [3] used the auxiliary equation method to construct more types of new exact traveling wave solutions of the nonlinear Klein–Gordon equation with quadratic and cubic nonlinearity.

The numerical solutions to the nonlinear Klein–Gordon equation have received considerable attention in the literature. A variety of second order finite difference schemes have been presented (see [11] and references therein). We note that alternative approaches using spectral and pseudo-spectral methods have recently been presented [12,13]. As is mentioned in [11] in a paper in [14], four finite difference schemes for approximating the nonlinear Klein–Gordon equation were discussed. They observed undesirable characteristics in some of the numerical schemes, in particular a loss of spatial symmetry and the onset of instability for large values of a parameter in the initial condition of the equation.

Finite difference methods are known as the first technique for solving partial differential equations. Even though these methods are very effective for solving various kinds of partial differential equations, conditional stability of explicit finite difference procedures and the need to use large amount of CPU time in implicit finite difference schemes limit the applicability of these methods [15]. Furthermore, these methods provide the solution of the problem on mesh points only and accuracy of the techniques is reduced in non-smooth and non-regular domains.

To avoid the mesh generation, in recent years meshless techniques have attracted the attention of researchers. In a meshless (mesh free) method a set of scattered nodes are used instead of meshing the domain of the problem. Some meshless schemes are the element free Galerkin method, the reproducing kernel particle, the local point interpolation etc. For more descriptions see [16] and references therein.

For the last 20 years, the radial basis functions method was known as a powerful tool for the scattered data interpolation problem. The use of radial basis functions as a meshless procedure for numerical solution of partial differential equations is based on the collocation scheme. Due to the collocation technique, this method does not need to evaluate any integral. The main advantage of numerical procedures which use radial basis functions over traditional techniques is the meshless property of these methods. Radial basis functions are used actively for solving partial differential equations. For example see [17,18].

In the last decade, the development of the radial basis functions (RBFs) as a truly meshless method for approximating the solutions of PDEs has drawn the attention of many researchers in science and engineering. One of the domain-type meshless methods, the so-called Kansa’s method developed by Kansa in 1990 [17,19], is obtained by directly collocating the RBFs, particularly the multiquadric (MQ), for the numerical approximation of the solution.

Kansa’s method was recently extended to solve various ordinary and partial differential equations including the one-dimensional nonlinear Burgers equation [20] with shock wave, shallow water equations for tide and currents simulation [21], heat transfer problems [18], and free boundary problems [22,23]. Fasshauer [24] later modified Kansa’s method to a Hermite–type collocation method for the solvability of the resultant collocation matrix.

The traditional RBFs are globally defined functions which result in a full resultant coefficient matrix. This hinders the application of the RBFs to solve large scale problems due to severe ill-conditioning of the coefficient matrix. To tackle this ill-conditioning problem, a new class of compactly supported RBFs were constructed by [25]. For the theoretical developments of the RBFs in scattered data interpolation, Madych and Nelson [26,27] showed that the RBF–MQ interpolant employs exponential convergence with minimal semi-norm errors. Recently, Franke and Schaback [28,29] provided a theoretical proof in using the RBFs for the numerical solutions of PDEs. More recently, Hon and Wu [30] gave a theoretical justification in combining the RBFs with those advanced techniques of domain decomposition, multilevel/multigrid, Schwarz iterative schemes, and preconditioning in the FEM discipline. The RBFs used to solve several partial differential equations [33–36].

This paper presents a new numerical scheme to solve the nonlinear Klein–Gordon equation with quadratic and cubic nonlinearity using the collocation method and approximating directly the solution using thin plate splines radial basis function (Kansa’s method). The scheme is similar to finite difference methods. The Klein–Gordon equation is solved in [37] using the variational iteration method. It is worth to point out that the RBFs method developed in the current paper can be employed to solve the nonclassic boundary value problems investigated in [38,39].
The organization of the paper as follows. In Section 2 we show how we use the radial basis functions to approximate the solution. In Section 3 we apply the method on the nonlinear Klein–Gordon equation. The results of numerical experiments are presented in Section 4. Section 5 is dedicated to a brief conclusion. Finally some references are introduced at the end.

2. Radial basis function approximation

The approximation of a distribution $u(x)$, using radial basis functions, may be written as a linear combination of $N$ radial functions; usually it takes the following form:

$$u(x) \simeq \sum_{j=1}^{N} \lambda_j \varphi(x, x_j) + \psi(x) \quad \text{for } x \in \Omega \subset \mathbb{R}^d,$$

(2.1)

where $N$ is the number of data points, $x = (x_1, x_2, \ldots, x_d)$, $d$ is the dimension of the problem, $\lambda_j$'s are coefficients to be determined and $\varphi$ is the radial basis function. Eq. (2.1) can be written without the additional polynomial $\psi$. In that case $\varphi$ must be unconditionally positive definite to guarantee the solvability of the resulting system (e.g. Gaussian or inverse multiquadrics). However, $\psi$ is usually required when $\varphi$ is conditionally positive definite, i.e., when $\varphi$ has a polynomial growth towards infinity. Examples are thin plate splines and multiquadrics. We will use thin plate splines for the numerical scheme introduced in Section 3. The generalized Thin Plate Splines (TPS) are defined as:

$$\varphi(x, x_j) = \varphi(r_j) = r_j^{2m} \log(r_j), \quad m = 1, 2, 3, \ldots,$$

(2.2)

where $r_j = \|x - x_j\|$ is the Euclidean norm.

Since $\varphi$ in (2.2) is $C^{2m-1}$ continuous, higher order thin plate splines must be used for higher order partial differential operators. For the nonlinear sine-Gordon equation, an $m = 2$ is used for thin plate splines (i.e. second order thin plate splines).

If $\mathcal{P}_q^d$ denotes the space of $d$-variate polynomials of order not exceeding $q$, and letting the polynomials $P_1, \ldots, P_m$ be the basis of $\mathcal{P}_q^d$ in $\mathbb{R}^d$, then the polynomial $\psi(x)$, in Eq. (2.1), is usually written in the following form:

$$\psi(x) = \sum_{i=1}^{m} \xi_i P_i(x),$$

(2.3)

where $m = (q - 1 + d)!/(d!(q - 1)!)$.

To determine the coefficients $(\lambda_1, \ldots, \lambda_N)$ and $(\xi_1, \ldots, \xi_m)$, the collocation method is used. However, in addition to the $N$ equations resulting from collocating (2.1) at the $N$ points, an extra $m$ equations are required. This is ensured by the $m$ conditions for (2.1),

$$\sum_{j=1}^{N} \lambda_j P_i(x_j) = 0, \quad i = 1, \ldots, m.$$  

(2.4)

In a similar representation to (2.1), for any linear partial differential operator $\mathcal{L}$, $\mathcal{L}u$ can be approximated by

$$\mathcal{L}u(x) \simeq \sum_{j=1}^{N} \lambda_j \mathcal{L} \varphi(x, x_j) + \mathcal{L} \psi(x).$$

(2.5)

3. The nonlinear Klein–Gordon equation

Let us consider the following one-dimensional nonlinear Klein–Gordon equation:

$$\frac{\partial^2 u}{\partial t^2} + \alpha \frac{\partial^2 u}{\partial x^2} + \beta u + \gamma u^k = f(x, t), \quad x \in \Omega = [a, b] \subset \mathbb{R}, \quad 0 < t \leq T,$$

(3.1)

with the initial conditions

$$\begin{cases}
    u(x, 0) = g_1(x), & x \in \Omega \\
    u_t(x, 0) = g_2(x), & x \in \Omega,
\end{cases}$$

(3.2)

and Dirichlet boundary condition

$$u(x, t) = h(x, t), \quad x \in \partial \Omega, \quad 0 < t \leq T,$$

(3.3)

where $\alpha$, $\beta$ and $\gamma$ are known constants, $k = 2$ when we have quadratic nonlinearity and $k = 3$ when we have cubic nonlinearity. $f$, $g_1$, $g_2$ and $h$ are known functions, and the function $u$ is unknown.
First, let us discretize (3.1) according to the following $\theta$-weighted scheme
\[
\frac{u(x, t + \delta t) - 2u(x, t) + u(x, t - \delta t)}{(\delta t)^2} + \theta \left( \alpha \nabla^2 u(x, t + \delta t) + \beta u(x, t + \delta t) \right)
+ (1 - \theta) \left[ \alpha \nabla^2 u(x, t) + \beta u(x, t) \right] + \gamma u(x, t)^k = f(x, t + \delta t),
\]
where $\nabla$ is the gradient differential operator, $0 \leq \theta \leq 1$, and $\delta t$ is the time step size. Rearranging (3.4), using the notation $u^n = u(x, t^n)$ where $t^n = t^{n-1} + \delta t$, we obtain
\[
\begin{align*}
(1 + \theta(\delta t)^2) u^{n+1} + & \alpha \theta(\delta t)^2 \nabla^2 u^{n+1} = (2 - \beta (1 - \theta)(\delta t)^2) u^n \\
- & \alpha (1 - \theta)(\delta t)^2 \nabla^2 u^n - \gamma (\delta t)^2 (u^n)^k - u^{n-1} + (\delta t)^2 f^{n+1}.
\end{align*}
\] (3.5)
Assuming that there are a total of $(N - 2)$ interpolation points, $u(x, t^n)$ can be approximated by
\[
u^n(x) \simeq \sum_{j=1}^{N-2} \lambda^n_j \varphi_j(r_j) + \lambda^n_{N-1} x + \lambda^n_N.
\] (3.6)
To determine the interpolation coefficients ($\lambda_1, \lambda_2, \ldots, \lambda_{N-1}, \lambda_N$), the collocation method is used by applying (3.6) at every point $x_i, i = 1, 2, \ldots, N - 2$. Thus we have
\[
u^n(x_i) \simeq \sum_{j=1}^{N-2} \lambda^n_j \varphi_j(r_j) + \lambda^n_{N-1} x_i + \lambda^n_N,
\] (3.7)
where $r_j = (x_i - x_j)^2$. The additional conditions due to (2.4) can be written as
\[
\sum_{j=1}^{N-2} \varphi_j(x_i) = 0.
\] (3.8)
Writing (3.7) together with (3.8) in a matrix form we have
\[
[u^n] = A[\lambda]^n,
\] (3.9)
where $[u^n] = [u^n_1 \; u^n_2 \; \ldots \; u^n_{N-2} \; 0 \; 0]^T$, $[\lambda]^n = [\lambda^n_1 \; \lambda^n_2 \; \ldots \; \lambda^n_{N-1}]^T$ and $A = [a_{ij}, 1 \leq i, j \leq N]$ is given by
\[
A = \begin{bmatrix}
\varphi_1 & \cdots & \varphi_{N-2} & x_1 & 1 \\
\vdots & \ddots & \vdots & \vdots & \vdots \\
\varphi_{N-2} & \cdots & \varphi_{N-1} & x_{N-2} & 1 \\
x_1 & \cdots & x_{N-2} & 0 & 0 \\
1 & \cdots & 1 & 0 & 0
\end{bmatrix}.
\] (3.10)
Assuming that there are $p < (N - 2)$ internal points and $(N - 2 - p)$ boundary points, then the $(N \times N)$ matrix $A$ can be split into: $A = A_d + A_b + A_e$, where
\[
A_d = [a_{ij} \text{ for } (1 \leq i \leq p, 1 \leq j \leq N) \text{ and } 0 \text{ elsewhere}],
A_b = [a_{ij} \text{ for } (p + 1 \leq i \leq N - 2, 1 \leq j \leq N) \text{ and } 0 \text{ elsewhere}],
A_e = [a_{ij} \text{ for } (N - 1 \leq i \leq N, 1 \leq j \leq N) \text{ and } 0 \text{ elsewhere}].
\] (3.11)
Using the notation $\mathcal{L}A$ to designate the matrix of the same dimension as $A$ and containing the elements $\tilde{a}_{ij}$, where $\tilde{a}_{ij} = \mathcal{L}a_{ij}, 1 \leq i, j \leq N$, then Eq. (3.5) together with (3.3) can be written, in the matrix form, as
\[
B[\lambda]^{n+1} = C[\lambda]^{n} - \gamma (\delta t)^2 \left( [u_d]^{n+1} - [u_d]^{n-1} + (\delta t)^2 [f]^{n+1} + [H]^{n+1} \right),
\] (3.12)
where
\[
B = (1 + \theta(\delta t)^2) A_d + \alpha \theta(\delta t)^2 \nabla^2 A_d + A_b + A_e,
C = (2 - \beta (1 - \theta)(\delta t)^2) A_d - \alpha (1 - \theta)(\delta t)^2 \nabla^2 A_d,
[u_d]^{n} = [u^n_1 \; \ldots \; u^n_p \; 0 \; 0]^T,
[f]^{n+1} = [f_1^{n+1} \; \ldots \; f_p^{n+1} \; 0 \; 0]^T,
[\lambda]^{n+1} = [\lambda_1^{n+1} \; \ldots \; \lambda_p^{n+1} \; 0 \; 0]^T, [H]^{n+1} = [0 \; \ldots \; 0 \; h_{p+1}^{n+1} \; \ldots \; h_{N-2}^{n+1} \; 0 \; 0]^T.
\] Eq. (3.12) is obtained by combining (3.5), which applies to the domain points, while (3.3) applies to the boundary points.
Fig. 1. Analytical and estimated functions in $t = 10$ s, with $dt = 0.0001$ and $dx = 0.02$, for Example 1.

At $n = 0$, Eq. (3.12) has the following form:

$$B[λ]^1 = C[λ]^0 - γ(δt)^2 \left([u_d]^0\right)^k - [u_d]^{-1} + (δt)^2[f]^1 + [H]^1.$$  \hfill (3.13)

To approximate $u^{-1}$ the second initial condition is used. For this we discretize the second initial condition as

$$\frac{u^1(x) - u^{-1}(x)}{2δt} = g_2(x), \quad x \in Ω.$$  \hfill (3.14)

Writing (3.13) together with (3.14) we have

$$(B + A_d)[λ]^1 = C[λ]^0 - γ(δt)^2 \left([u_d]^0\right)^k + 2δt[G] + (δt)^2[f]^1 + [H]^1,$$ \hfill (3.15)

where $[G] = [(g_2)_1 ... (g_2)_p 0 ... 0]^T$.

Since the coefficient matrix is unchanged in time steps, we use the LU factorization to the coefficient matrix only once and use this factorization in our algorithm.

**Remark.** Although Eq. (3.12) is valid for any value of $θ \in [0, 1]$, we will use $θ = 1/2$ (the famous Crank–Nicholson scheme).

4. Test problems and numerical results

In this section we present some numerical results of our scheme for the nonlinear Klein–Gordon equation.

4.1. Example 1

In this example, we consider the nonlinear Klein–Gordon equation (3.1) with quadratic nonlinearity. The constants are $α = -1$, $β = 0$ and $γ = 1$ in the interval $-1 \leq x \leq 1$. The initial conditions are given by

$$\begin{cases} u(x, 0) = x, & -1 \leq x \leq 1 \\ u_t(x, 0) = 0, & -1 \leq x \leq 1 \end{cases}$$ \hfill (4.1)

and the right-hand side function is

$$f(x, t) = -x \cos t + x^2 \cos^2 t.$$ \hfill (4.2)

The analytical solution is given in [31] as

$$u(x, t) = x \cos t.$$ \hfill (4.3)

We extract the boundary function $h(x, t)$ from the exact solution. The $L_∞$, $L_2$ errors and Root-Mean-Square (RMS) of errors are obtained in Table 1 for $t = 1, 3, 5, 7$ and 10. The graph of analytical and estimated functions for $t = 10$ is given in Fig. 1. We also draw the space–time graph of the estimated solution in Fig. 2.
Fig. 2. Space–time graph of the solution up to $t = 10$ s, with $dt = 0.0001$ and $dx = 0.02$, for Example 1.

### Table 1
Computational domain is $[-1, 1]$.

<table>
<thead>
<tr>
<th>$t$</th>
<th>$L_{\infty}$-error</th>
<th>$L_2$-error</th>
<th>RMS</th>
<th>Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$1.2540 \times 10^{-5}$</td>
<td>$6.5422 \times 10^{-5}$</td>
<td>$6.5097 \times 10^{-6}$</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>$1.5554 \times 10^{-5}$</td>
<td>$1.1717 \times 10^{-4}$</td>
<td>$1.1659 \times 10^{-5}$</td>
<td>22</td>
</tr>
<tr>
<td>5</td>
<td>$3.3792 \times 10^{-5}$</td>
<td>$2.2011 \times 10^{-4}$</td>
<td>$2.1902 \times 10^{-5}$</td>
<td>49</td>
</tr>
<tr>
<td>7</td>
<td>$3.7753 \times 10^{-5}$</td>
<td>$2.5892 \times 10^{-4}$</td>
<td>$2.5763 \times 10^{-5}$</td>
<td>83</td>
</tr>
<tr>
<td>10</td>
<td>$1.3086 \times 10^{-5}$</td>
<td>$7.9854 \times 10^{-5}$</td>
<td>$7.9458 \times 10^{-6}$</td>
<td>150</td>
</tr>
</tbody>
</table>

$L_{\infty}$, $L_2$ and RMS errors, with $dt = 0.0001$, $dx = 0.02$.

### Table 2
Computational domain is $[0, 1]$.

<table>
<thead>
<tr>
<th>$t$</th>
<th>$L_{\infty}$-error</th>
<th>$L_2$-error</th>
<th>RMS</th>
<th>Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$1.1012 \times 10^{-5}$</td>
<td>$5.4998 \times 10^{-5}$</td>
<td>$5.4725 \times 10^{-6}$</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>$1.6496 \times 10^{-4}$</td>
<td>$1.1522 \times 10^{-3}$</td>
<td>$1.1465 \times 10^{-4}$</td>
<td>14</td>
</tr>
<tr>
<td>3</td>
<td>$5.9728 \times 10^{-4}$</td>
<td>$3.2588 \times 10^{-3}$</td>
<td>$3.2426 \times 10^{-4}$</td>
<td>25</td>
</tr>
<tr>
<td>4</td>
<td>$1.8264 \times 10^{-3}$</td>
<td>$9.8191 \times 10^{-3}$</td>
<td>$9.7704 \times 10^{-4}$</td>
<td>37</td>
</tr>
<tr>
<td>5</td>
<td>$3.6915 \times 10^{-3}$</td>
<td>$1.9139 \times 10^{-2}$</td>
<td>$1.9044 \times 10^{-3}$</td>
<td>52</td>
</tr>
</tbody>
</table>

$L_{\infty}$, $L_2$ and RMS errors, with $dt = 0.0001$, $dx = 0.02$.

#### 4.2. Example 2

Consider the nonlinear Klein–Gordon equation (3.1) with quadratic nonlinearity in the interval $0 \leq x \leq 1$ with $\alpha = -1$, $\beta = 0$ and $\gamma = 1$. The initial conditions are given by

\[
\begin{align*}
  u(x, 0) &= 0, \quad 0 \leq x \leq 1 \\
  u_t(x, 0) &= 0, \quad 0 \leq x \leq 1.
\end{align*}
\]  

(4.4)

The analytical solution is given in [31] as

\[
u(x, t) = x^3 t^3.
\]  

(4.5)

In this case $f(x, t) = 6xt(x^2 - t^2) + x^6 t^6$. The $L_{\infty}$, $L_2$ errors and RMS of errors are obtained in Table 2 for $t = 1, 2, 3, 4$ and 5. The graph of analytical and estimated functions for $t = 5$ and the space–time graph of the estimated solution are given in Figs. 3 and 4.

#### 4.3. Example 3

In this example, we consider the nonlinear Klein–Gordon Eq. (3.1) with cubic nonlinearity with constants $\alpha = -2.5$, $\beta = 1$ and $\gamma = 1.5$ in the interval $0 \leq x \leq 1$. The initial conditions are given by

\[
\begin{align*}
  u(x, 0) &= B \tan(Kx), \quad 0 \leq x \leq 1 \\
  u_t(x, 0) &= Bc K \sec^2(Kx), \quad 0 \leq x \leq 1,
\end{align*}
\]  

(4.6)
and the exact solution by [32] is

\[ u(x, t) = B \tan(K(x + ct)), \]  

where \( B = \sqrt{\frac{\beta}{\gamma}} \) and \( K = \sqrt{-\frac{\beta}{2(\alpha + c^2)}} \).

In this example \( f(x, t) = 0 \). We extract the boundary function \( h(x, t) \) from the exact solution. In Table 3, the \( L_\infty \), \( L_2 \) errors and RMS of errors are obtained for two values of \( c \) (\( c = 0.5 \) and \( c = 0.05 \)) for \( t = 1, 2, 3 \) and \( 4 \). The graph of analytical and estimated functions for \( t = 4 \) and the space–time graph of the estimated solution for each value of \( c \) are given in Figs. 5 and 6.

4.4. Example 4

Similarly to the previous example, we consider the nonlinear Klein–Gordon equation (3.1) with cubic nonlinearity in the interval \(-1 \leq x \leq 1\) and constants \( \alpha = -1, \beta = 1 \) and \( \gamma = 1 \). The initial conditions are given by

\[
\begin{align*}
    u(x, 0) &= x^2 \cosh(x), & -1 \leq x \leq 1 \\
    u_t(x, 0) &= x^2 \cosh(x), & -1 \leq x \leq 1,
\end{align*}
\]  

and the right-hand side function is

\[ f(x, t) = (x^2 - 2) \cosh(x + t) - 4x \sinh(x + t) + x^6 \cosh^3(x + t). \]  

The exact solution is

\[ u(x, t) = x^2 \cosh(x + t). \]
Fig. 5. Analytical and estimated functions in $t = 4$ s, with $dt = 0.001$ and $dx = 0.01$, for Example 3. (Right) $c = 0.5$, (Left) $c = 0.05$.

Fig. 6. Space-time graph of the solution up to $t = 4$ s, with $dt = 0.001$ and $dx = 0.01$, for Example 3. (Right) $c = 0.5$, (Left) $c = 0.05$.

Table 3

<table>
<thead>
<tr>
<th>$c$</th>
<th>$L_\infty$-error</th>
<th>$L_2$-error</th>
<th>RMS</th>
<th>Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>$5.9964 \times 10^{-6}$</td>
<td>$4.0761 \times 10^{-5}$</td>
<td>$4.0559 \times 10^{-6}$</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>$2.1973 \times 10^{-5}$</td>
<td>$1.5769 \times 10^{-4}$</td>
<td>$1.5691 \times 10^{-5}$</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>$9.0893 \times 10^{-5}$</td>
<td>$6.4792 \times 10^{-4}$</td>
<td>$6.4470 \times 10^{-5}$</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>$8.2945 \times 10^{-4}$</td>
<td>$5.3572 \times 10^{-3}$</td>
<td>$5.3306 \times 10^{-4}$</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>$3.6497 \times 10^{-7}$</td>
<td>$1.7861 \times 10^{-6}$</td>
<td>$1.7772 \times 10^{-7}$</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>$3.8952 \times 10^{-7}$</td>
<td>$1.5383 \times 10^{-6}$</td>
<td>$1.5306 \times 10^{-7}$</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>$4.2123 \times 10^{-7}$</td>
<td>$1.7275 \times 10^{-6}$</td>
<td>$1.7190 \times 10^{-7}$</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>$4.5928 \times 10^{-7}$</td>
<td>$2.0097 \times 10^{-6}$</td>
<td>$1.9997 \times 10^{-7}$</td>
<td>2</td>
</tr>
</tbody>
</table>

$L_\infty$, $L_2$ and RMS errors, with $dt = 0.001$, $dx = 0.01$.

We extract the boundary function $h(x, t)$ from the exact solution. The $L_\infty$, $L_2$ errors and Root-Mean-Square (RMS) of errors are obtained in Table 4 for $t = 1, 2, 3, 4$ and 5. The graph of analytical and estimated functions for $t = 5$ is given in Fig. 7. We also draw the space–time graph of the estimated solution in Fig. 8.

4.5. Example 5

To compare the accuracy of the proposed numerical scheme with four finite difference schemes mentioned in [14], we consider the nonlinear Klein–Gordon equation (3.1) with cubic nonlinearity with constants $\alpha = -1$, $\beta = 1$ and $\gamma = 1$ in
Fig. 7. Analytical and estimated functions in $t = 5\,\text{s}$, with $dt = 0.0001$ and $dx = 0.01$, for Example 4.

Fig. 8. Space–time graph of the solution up to $t = 5\,\text{s}$, with $dt = 0.0001$ and $dx = 0.01$, for Example 4.

Table 4
Computational domain is $[-1, 1]$.

<table>
<thead>
<tr>
<th>$t$</th>
<th>$L_{\infty}$-error $\times 10^{-5}$</th>
<th>$L_2$-error $\times 10^{-4}$</th>
<th>RMS $\times 10^{-4}$</th>
<th>Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.0705</td>
<td>2.9474</td>
<td>2.0789</td>
<td>23</td>
</tr>
<tr>
<td>2</td>
<td>5.0260</td>
<td>2.7082</td>
<td>1.9102</td>
<td>51</td>
</tr>
<tr>
<td>3</td>
<td>2.0612</td>
<td>9.7246</td>
<td>6.8592</td>
<td>84</td>
</tr>
<tr>
<td>4</td>
<td>6.5720</td>
<td>2.7881</td>
<td>1.9666</td>
<td>120</td>
</tr>
<tr>
<td>5</td>
<td>1.9067</td>
<td>7.7337</td>
<td>5.4549</td>
<td>160</td>
</tr>
</tbody>
</table>

$L_{\infty}$, $L_2$ and RMS errors, with $dt = 0.0001$, $dx = 0.01$.

the interval $0 \leq x \leq 1.28$. We consider the initial data

$$u(x, 0) = A \left[ 1 + \cos \left( \frac{2\pi x}{1.28} \right) \right],$$

$$u_t(x, 0) = 0,$$

(4.11)

where the amplitude $A$ is taken in $[0.1, 100]$. The right-hand side function is $f(x, t) = 0$. The boundary conditions are given by

$$u_x(x, t) = 0, \quad x = 0, 1.28, \quad t > 0.$$  

(4.12)

For these data, and due to the periodic boundary conditions, the continuous solutions remain always symmetric with respect to the center of the spatial interval [14].
Jiminez and Vazquez [14] studied this problem and found undesirable characteristics in some of the numerical schemes, in particular a loss of spatial symmetry and the onset of instability for large values of a parameter $A$ in the initial condition of the equation.

We solved problem (4.11) and (4.12) with $dx = 0.01$, $dt = 0.0001$ and the results obtained were compared with the relevant results in [14]. We deduced:

- accurate results for amplitude $A \leq 1$ (Fig. 9) similar to the results of [14] when $t \in [0, 1000]$, were observed,
- $U$ remained bounded for $A = 100$ and $t \in [0, 15]$ (Fig. 10) instead of the known $A = 100$ and $t \in [0, 0.8]$.

Then, it may be concluded that the proposed numerical scheme gives more accurate results than the already known ones for problem (4.11) and (4.12).

5. Conclusion

In this paper, we discussed the nonlinear Klein–Gordon equation with quadratic and cubic nonlinearity. We proposed a numerical scheme to solve this nonlinear equation using collocation points and approximating directly the solution using the thin plate splines (TPS) radial basis function. The implementation of the method is simple as finite difference methods. The numerical results given in the previous section demonstrate the good accuracy of this scheme.

Acknowledgments

The authors are very grateful to both reviewers for carefully reading the paper and for their comments and suggestions which have improved the paper. The authors are also very much thankful to the Principal Editor, Professor Taketomo Mitsui.
References