Computer Systems Science & Engineering
DOI: 10.32604/csse.2023.032404
Article

Particle Swarm Optimization for Solving Sine-Gordan Equation

Geeta Arora[1], Pinkey Chauhan[2], Muhammad Imran Asjad[3], Varun Joshi[1], Homan Emadifar[4] and Fahd Jarad[5,6,7,*]

[1]Lovely Professional University, Punjab, 144118, India
[2]Jaypee Institute of Information Technology, Noida, 201304, India
[3]Department of Mathematics, University of Management and Technology, Lahore, 54770, Pakistan
[4]Department of Mathematics, Hamedan Branch, Islamic Azad University, Hamedan, Iran
[5]Department of Mathematics, Cankaya University, Etimesgut, Ankara, 06790, Turkey
[6]Department of Mathematics, King Abdulaziz University, Saudi Arabia
[7]Department of Medical Research, China Medical University Hospital, China Medical University, Taichung, Taiwan
*Corresponding Author: Fahd Jarad. Email: fahd@cankaya.edu.tr
Received: 17 May 2022; Accepted: 14 July 2022

Abstract: The term 'optimization' refers to the process of maximizing the beneficial attributes of a mathematical function or system while minimizing the unfavorable ones. The majority of real-world situations can be modelled as an optimization problem. The complex nature of models restricts traditional optimization techniques to obtain a global optimal solution and paves the path for global optimization methods. Particle Swarm Optimization is a potential global optimization technique that has been widely used to address problems in a variety of fields. The idea of this research is to use exponential basis functions and the particle swarm optimization technique to find a numerical solution for the Sine-Gordan equation, whose numerical solutions show the soliton form and has diverse applications. The implemented optimization technique is employed to determine the involved parameter in the basis functions, which was previously approximated as a random number in the work reported till now in the literature. The obtained results are comparable with the results obtained in the literature. The work is presented in the form of figures and tables and is found encouraging.

Keywords: Differential quadrature method; B-spline; particle swarm optimization; Sine-Gordan equation

1 Introduction

Whenever there is a need to optimize any one of the parameters involved in a problem there needs a tool or technique that can be implemented to solve it. The main focus is to obtain the best feasible collection of values to achieve a given goal while adhering to a set of constraints. The majority of real-world situations can be modelled as optimization problems that include large dimensions, non-linearity, multimodality, constraints, and other factors. Since traditional optimization approaches are generally incapable of resolving these complex problems, non-traditional efficient optimization tools/techniques are implemented to deal with a wide range of such problems. Nature-inspired algorithms belong to the category of optimization methods that have gained much popularity since their inception. Nature-inspired optimization algorithms are the approaches that replicate an existing natural process to find an optimum solution to a problem that can't be handled using classical techniques. Swarm intelligence paradigm is an emerging field which simulates the social behavior of organisms. Since its inception, various algorithms have been proposed to handle complex problems arising in different spheres. Particle swarm optimization (PSO) is one of the potential global optimization techniques in the category of nature-inspired algorithms. PSO simulates the foraging process of swarm analogies such as bird flocks and fish schools. It is inspired by the well-informed social behaviour of swarms and was firstly proposed by Kennedy et al. in 1995 [1]. Fast convergence to the global optimum, a simple to implement code, and a sophisticated computation-free environment are all advantages of employing PSO. The search process in PSO is driven by the velocity and position update equations.

The present work is intended to obtain the numerical solution for the Sine-Gordan equation, which is a well-known nonlinear partial differential equation using exponential basis functions implementing the particle swarm optimization technique. The implemented optimization technique is to find the involved parameter in the exponential B-spline basis functions that was just approximated as a random number in the work reported till now in the literature.

Sine-Gordon (SG) equation is a second-order hyperbolic partial differential equation whose numerical solutions depict the soliton form and have intense applications in science and engineering. It appears in the study of optics as a solution to the classical Maxwell systems [2]. This equation also appears in the literature in the geometrical study of the soliton in view of the canonical field [3]. This study also depicts a relation between the black hole temperature and the soliton velocity. SG equation also presents a mathematical model to illustrate the fault dynamics of the phenomena of strain waves and earthquakes [4]. It plays a significant role in understanding the seismic distortion effects on the earth's crust and the theory behind the cause of faulty natural substances. The soliton solution of the kinks form of the equation makes it a suitable equation to understand the concepts related to the different phenomena.

The equation is given by:

$$u_{tt} + \alpha\, u_t = \beta\ u_{xx} + \eta(x)\sin(u) \tag{1}$$

To be solved with set of initial conditions:

$$u(x,\ 0) = \phi_1(x)\ and\ u_t(x,\ 0) = \phi_2(x)$$

And values defined at the boundaries.

Here, α and β are real constants and $\eta(x,\ y)$ depicts the Josephson current density. The constant α represents the dissipative term that plays an important role in converting the equation from damped $(\alpha \geq 0)$ to undamped for $(\alpha = 0)$.

Researchers have investigated the SG equation to study the solution based on the properties shown by the equation. Ben-yu et al. [5] have implemented finite difference based on the concept of conserved discrete energy to discuss the solution of the equation. Albowitz et al. [6,7] discussed the equation for the unstable nature using the nonlinear spectrum. Various analytical and numerical approaches have been applied by the researchers to solve this equation for its soliton solutions including the modified decomposition method [8,9] for solving this equation 1D and 2D, the modified Adomian decomposition method [10] to solve the SG equation in (N + 1)-dimensions, homotopy analysis method [11], boundary element and boundary integral approach [12,13], Compact finite difference of order-6 (CFD6) scheme [14], tension spline-based approximation scheme [15], Modified cubic B-spline (MCB) collocation technique [16], localized method of approximate particular solutions [17], Legendre spectral element method [18], virtual element method [19], Barycentric rational interpolation and local radial basis functions [20], fourth-order collocation scheme [21] and rational radial basis function [22].

In this work, exponential basis functions with the differential quadrature method and the particle swarm optimization technique are implemented to find a numerical solution of the Sine-Gordan equation. To the authors' knowledge, there is no such optimization method reported in the literature to calculate the value of the parameter involved in the exponential B-spline which plays a crucial role in finding the solutions.

This paper is organized in the following manner: Section 2 is concerned with the numerical scheme for the implementation of the PSO in the exponential B-spline which is implemented in the differential quadrature approach. In Section 3, two test problems of the SG equation are demonstrated with the different sets of parameters. The summary of this research paper is stated in Section 4.

2 Numerical Scheme

2.1 Differential Quadrature Method

Numerical methods have always proved as an efficient tool to solve differential equations in a programmable approach. There is a lot of mathematical software that assists researchers in obtaining the numerical solution of differential equations by a set of algorithms. Among the numerous available numerical approaches, the Differential quadrature method (DQM) is a higher-order method proposed by Bellman and Casti [23] that provides accurate results with a smaller number of grid points by discretization of the domain. One of the important characteristics of the DQM is the basis functions. Different forms of basis functions have been successfully employed to find the underlying weighting coefficients to obtain the solution [24,25].

Consider the distribution of domain $x \in [a, b]$as $a = x_1 < x_2 < \cdots < x_N = b$ with N number of grid points. If $u(x, t)$ is a smooth function over the solution domain, it's r^{th} derivative with respect to x at a grid point x_i can be approximated by a linear summation of all the functional values evaluated using weighing coefficients $w_{ij}^{(r)}$as follows:

$$\frac{d^{(r)}u}{dx^{(r)}}|_{x_i} = \sum_{j=1}^{N} w_{i,j}^{(r)} u(x_j), \qquad i = 1, 2, \ldots, N, \qquad r = 1, 2, \ldots, N-1. \tag{2}$$

2.2 Exponential B-Spline Basis Function

In the last few years, the B-Spline basis functions have been used in their different forms successfully in DQM to obtain numerical solutions to differential equations [25,26]. These functions are very popular because of the properties of continuity, compact support, orthogonality, and capability to handle the local phenomenon. B-spline functions of standard [27] and trigonometric forms [27] are successfully implemented to solve the well-known differential equations but the work reported in the literature using the exponential form of B-spline is less because of the involved parameter, whose value is taken as a hit and trial for reducing the errors [28,29].

As known the exponential B-spline functions are the generalization of the polynomial B-splines with a free parameter. The exponential B-spline of the third degree can be defined as follows [30]:

$$B_m(x)=\frac{1}{h^3}\begin{cases}\beta_2(x_{m-2}-x)-\frac{\beta_2}{\omega}\left(sinh(\omega(x_{m-2}-x))\right), & x\epsilon[x_{m-2},\ x_{m-1}]\\ \alpha_1+\beta_1(x_m-x)+\gamma_1 e^{\omega(x_m-x)}+\delta_1 e^{-\omega(x_m-x)}, & x\epsilon[x_{m-1},\ x_m]\\ \alpha_1+\beta_1(x-x_m)+\gamma_1 e^{\omega(x-x_m)}+\delta_1 e^{-\omega(x-x_m)}, & x\epsilon[x_m,\ x_{m+1}]\\ \beta_2(x-x_{m+2})-\frac{\beta_2}{\omega}\left(sinh(\omega(x-x_{m+2}))\right), & x\epsilon[x_{m+1},\ x_{m+2}]\end{cases} \tag{3}$$

here h is the uniform space partition and other parameters are reported as:

$$\alpha_1=\frac{\omega hc}{\omega hc-s}\ ,\ \ \beta_1=\frac{\omega}{2}\left(\frac{c(c-1)+s^2}{(\omega hc-s)(1-c)}\right),\ \ \beta_2=\frac{\omega}{2\ (\omega hc-s)},\ \ \gamma_1=\frac{1}{4}\left(\frac{(1-c+s)e^{-\omega h}-s}{(\omega hc-s)(1-c)}\right),$$

$$\delta_1=\frac{1}{4}\left(\frac{(-1+c+s)e^{\omega h}-s}{(\omega hc-s)(1-c)}\right),\ \ c=cosh(\omega h),\ \ s=sinh(\omega h).$$

The numerical values of the function and the derivatives at nodal point can be obtained as:

$$B_m(x_{m-1})=B_m(x_{m+1})=\frac{s-\omega h}{2(\omega hc-s)};\ \ B_m(x_m)=1$$

$$B'_m(x_{m-1})=\frac{\omega(c-1)}{2(\omega hc-s)};\ B'_m(x_{m+1})=\frac{\omega(1-c)}{2(\omega hc-s)};\ B'_m(x_m)=0$$

$$B''_m(x_{m-1})=B''_m(x_{m+1})=\frac{\omega^2 s}{2(\omega hc-s)};\ B''_m(x_m)=\frac{-\omega^2 s}{(\omega hc-s)}$$

Further before implementation of the basis function, it is modified at the boundary mesh points to satisfy the condition of a diagonally dominant matrix as follows

$$M_1(x)=B_1(x)+2B_0(x),\ M_2(x)=B_2(x)-B_0(x),$$

$$M_k(x)=B_k(x)\ for\ k=3,\ 4,\ \ldots,\ N-2,\ (5)$$

$$M_{N-1}(x)=B_{N-1}(x)-B_{N+1}(x),\ M_N(x)=B_N(x)+2B_{N+1}(x),$$

2.3 Particle Swarm Optimization Algorithm

Particle swarm optimization (PSO) algorithm is developed on the communal actions of birds in searching for food [31,32]. In this algorithm, particles are entities whose routine is measured by their locations. Each location presents a part of the solution that needs to be optimized. The search process is driven by changing particle's velocity and positions at every time step. In the swarm, position of each particle is a solution in D-dimensional space. The updating rules for velocity and position of each particle is given by

$$v_{id}^{t+1}=\chi[v_{id}^t+c_1\ r_1(p_{id}^t-x_{id}^t)+c_2\ r_2(p_{gd}^t-x_{id}^t)] \tag{4}$$

$$x_{id}^{t+1}=x_{id}^t+v_{id}^{t+1} \tag{5}$$

where, x_{id}^t, represents c particle's position and v_{id}^t represents i^{th} particle's velocity in d dimension at time step t, p_{gd} represents the particle having the best fitness value, p_{id} is the particle's best position visited so far, $c_1,\ c_2$ are acceleration coefficients which quantifies particle personal and global experience respectively, χ is called constriction coefficient which evaluates a value in the range [0, 1] and is given by

$$\chi = \frac{2\kappa}{\left|2 - \varphi - \sqrt{\varphi(\varphi - 4)}\right|}$$

With $\varphi = \varphi_1 + \varphi_2,\ \varphi_1 = c_1 r_1,\ \ \varphi_2 = c_2 r_2$ and $\kappa \approx 1$.

To start with the parameter optimization the first step is to initialize the parameter by assigning random positions to the particles in the defined range. The next step is to randomly assign velocity for all particles. After this, the values of the function are evaluated for the particles and then the information is collected from the particles regarding their updated values and the velocity and position are again updated to search for the global optimum value. The working of PSO is shown below by the algorithm.

Algorithm: Particle Swarm Optimization

```
Begin
   t→0              // iteration
      Initialize a D–dimensional swarm, S
      Evaluate fitness of each particle of swarm
      For t = 1 to Max iteration
          For i = 1 to S
             For d = 1 to D
               Update velocity using basic velocity update Eq. (4)
                Apply basic position update Eq. (5)
                end
             End–for-d;
           Evaluate fitness of updated positions
           Update Pbest and Gbest
         End-for-i;
        Iteration (t)++;
   End-for-t;
End
```

The solution thus obtained by the implementation of the approach is reliable as it is being searched for the number of iterations with the selected number of parameters with a predetermined population size. This approach has been successfully implemented in calculating the solution of equations using radial basis functions [31]. The parameters considered here are as: swarm size: 20; maximum iterations: 50; inertia weight is linearly decreased and social and cognitive coefficients are taken as $c_1 = c_2 = 2.05$. Some recent and interesting studies on the theory, parameters and application of PSO are presented in [33–35].

In the present work selection of the value of the parameter in the basis function rests on certain factors such as the degree of the function, the number of knot points and the precision of computations. It plays a vital role in the accuracy of numerical methods. The objective of this study is to obtain the parameter based on the optimization of the L_∞ error.

2.4 The Scheme's Implementation

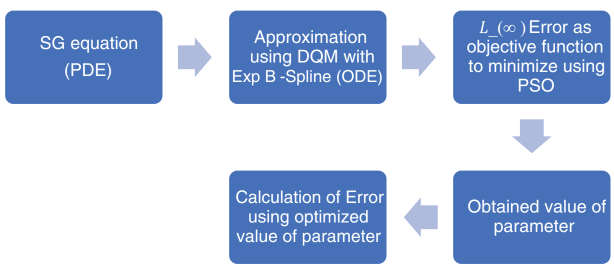
To solve the SG equation a transformation $u_t = v$ is used to reduce the equation in the following system of differential equations:

$$u_t = v$$

$$v_t + \alpha\, v = \beta\ \ u_{xx} + \eta(x)\ sin(u)$$

Substituting the approximations of the space derivatives using the DQM with exponential B-Spline basis functions results in an ordinary differential equation (ODE) that can be solved by any appropriate numerical method. Once the solutions are obtained for the equation with the known initial condition, then after the PSO technique is implemented to reduce the obtained error by comparing the numerical and exact solutions to minimize the errors. Once the value of the parameter is obtained for the minimum error, numerical results can be calculated on predefined domain and time intervals.

The numerical scheme can be summarized and visualized as follows:



3 Numerical Observations

The Sine-Gordon equation has been solved numerically for two different problems to authenticate the effectiveness and precision of the proposed method by computing the errors.

Example 1:

The numerical solutions of Sine-Gordon equation are obtained in the computational domain $x \in\ [-3,\ 3]$ for $\alpha = 0,\ \ \beta = 1$ and $\eta(x) = -1$ with initial conditions:

$$\phi_1(x) = 4tan^{-1}(exp(\gamma x\ ))$$

and

$$\phi_2(x) = \frac{-2\gamma exp(\gamma x\ )}{1 + exp(2\gamma x\ )}$$

The boundary conditions are computed from the exact solution given as:

$$u(x,\ t) = 4tan^{-1}(exp(\gamma x - 0.5t))$$

Here γ is a parameter that depends on velocity of solitary wave given as:

$$\gamma = \frac{1}{\sqrt{(1 - c^2)}}$$

For numerical computation, c is taken as 0.5 with the time step as $k = 0.0001$and space step size is $h = 0.02$ and 0.04 compared with results as reported in the literature [16,21,36]. The number of iteration used in the PSO is 50. As presented in Tab. 1, the results show the efficiency of the present approach as compared to that available in the literature for the parameter $\omega = 0.5579$.

Table 1: Comparisons of obtained results with results reported in literature

At h = 0.04	Present results		Mittal et al. [16]		Singh et al. [21]		Houssein et al. [34]	
Time (t)	L_2	L_∞	L_2	L_∞	L_2	L_∞	L_2	L_∞
0.25	6.47E−06	1.37E−05	3.66E−05	4.90E−05	1.60E−05	2.73E−05	5.67E−06	9.61E−06
0.50	8.68E−06	1.41E−05	9.00E−05	7.55E−05	2.37E−05	3.09E−05	8.39E−06	1.10E−05
0.75	9.99E−06	1.42E−05	1.60E−04	1.43E−04	2.91E−05	3.52E−05	1.05E−05	1.26E−05
1.0	1.08E−05	1.41E−05	2.27E−04	2.10E−04	3.25E−05	4.01E−05	1.24E−05	1.44E−05
At h = 0.02								
0.25	3.74E−06	8.34E−06			7.10E−06	6.62E−06		
0.50	4.92E−06	9.63E−06			1.23E−05	7.54E−06		
0.75	5.68E−06	1.11E−05			1.60E−05	1.01E−05		
1.0	6.21E−06	1.28E−05			1.79E−05	1.18E−05		

It can be concluded that obtained results are in good agreement and even superior as compared to results given by other researchers. The program is created and compiled on MATLAB 2014b on Intel Processor 64 bit, the CPU time for the algorithm is presented in Tab. 2. Numerical and the exact solutions are also depicted in the Fig. 1 at different time levels.

Table 2: Time elapsed in seconds for different knot partitions

Time (t)	CPU time (s)	
	h = 0.04	h = 0.02
0.25	0.2328	0.2345
0.50	0.3857	0.4023
0.75	0.5157	0.5234
1.0	0.6599	0.6672

Example 2:

Another example of Sine-Gordon equation to validate the algorithm is considered with $x \in [-20, 20]$ for $\alpha = 0, \quad \beta = 1$ and $\eta(x) = -1$ with initial conditions:

$$\phi_1(x) = 4tan^{-1}(c\sinh(\gamma x))$$

and

$$\phi_2(x) = 0$$

Here,

$$\gamma = \frac{1}{\sqrt{(1 - c^2)}}$$

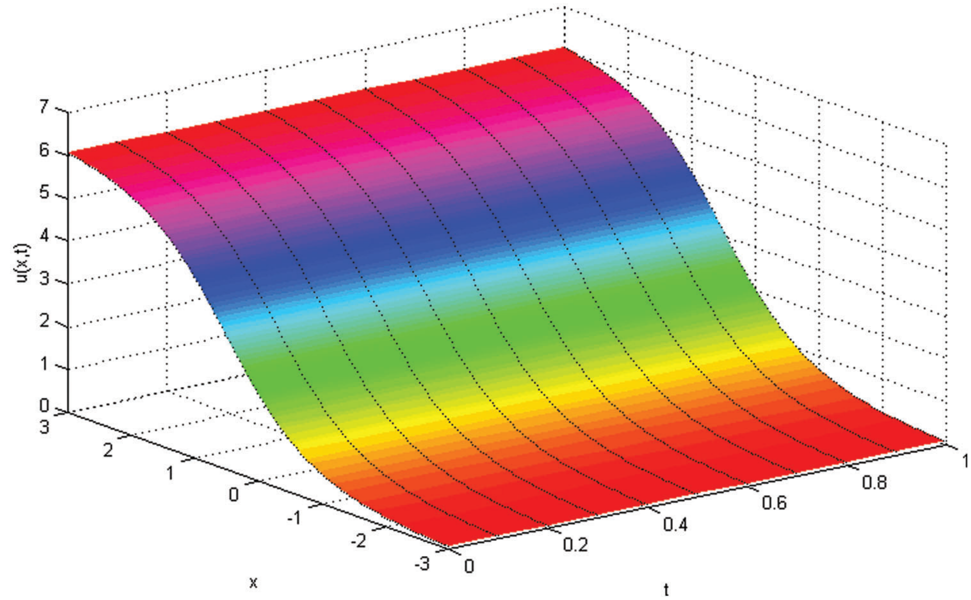


Figure 1: Physical profile of the SG equation example 1

The boundary conditions are computed from the exact solution given as:

$u(x,\ t) = 4tan^{-1}(c\sinh(\gamma x)\ \mathrm{sech}(\gamma ct))$

The results are calculated at $c = 0.5,\ \ k = 0.001$ and number of node points as $N = 300$. From the results given in Tab. 3 it can be seen that the present approach is efficiency and is comparable to results in literature for the parameter $\omega = 1$. The results are obtained for 1oo iterations with 5 as particle swarm size. To validate the obtained results the errors are presented and compared with results given in the literature [16,37]. To present the behavior of the solution the results are presented at various time in Figs. 2–5. A three-dimensional plot is given to showcase the behavior for time $1 \leq t \leq 20$.

Table 3: Comparisons of obtained results with results reported in literature

Time (t)	CPU time (Sec)	Present results			Mittal et al. [16]		Uddin et al. [37]
		L_2	L_∞	RMS	L_2	L_∞	L_∞
1	0.4135	1.048E−5	6.845E−6	1.652E−6	-	-	
2	0.5261	1.077E−5	7.919E−6	1.698E−6	2.564E−5	1.818E−5	1.568E – 3
10	0.6827	3.190E−5	2.047E−5	5.027E−6	8.850E−5	5.228E−5	3.151E−3
20	0.8215	6.289E−5	3.616E−5	9.911E−6	1.713E−4	9.438E−5	1.828E−3

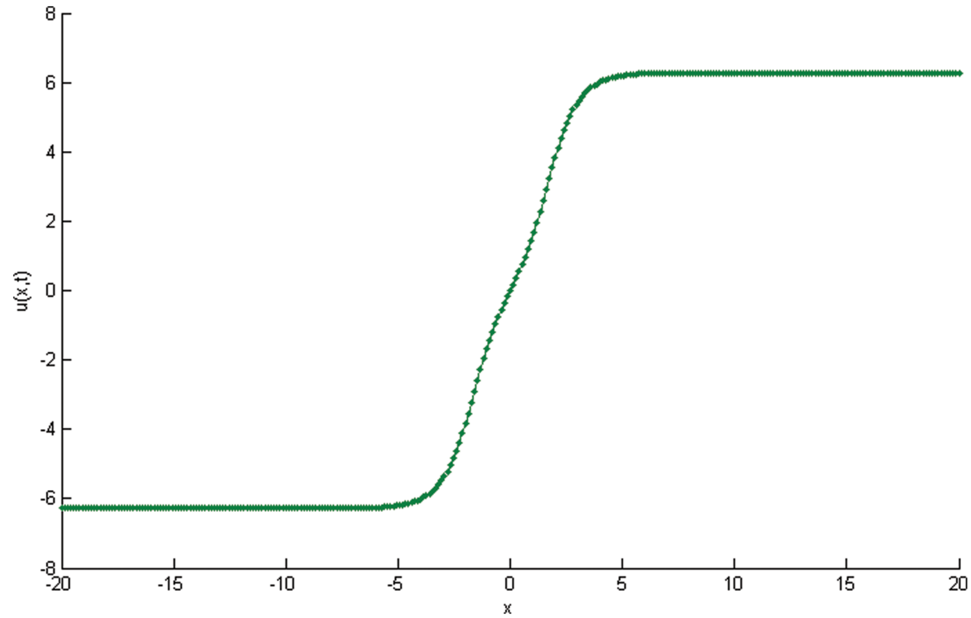


Figure 2: Physical profile of the SG equation example 2 at t = 2

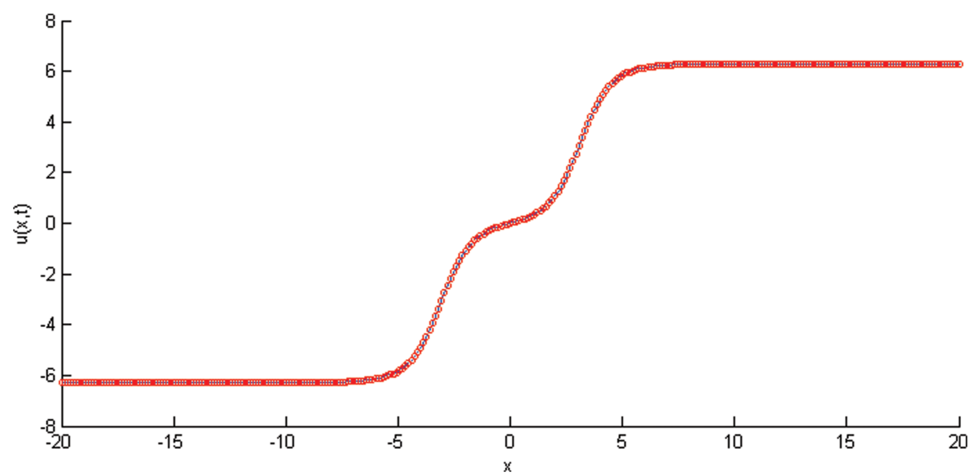


Figure 3: Physical profile of the SG equation example 2 at t = 5

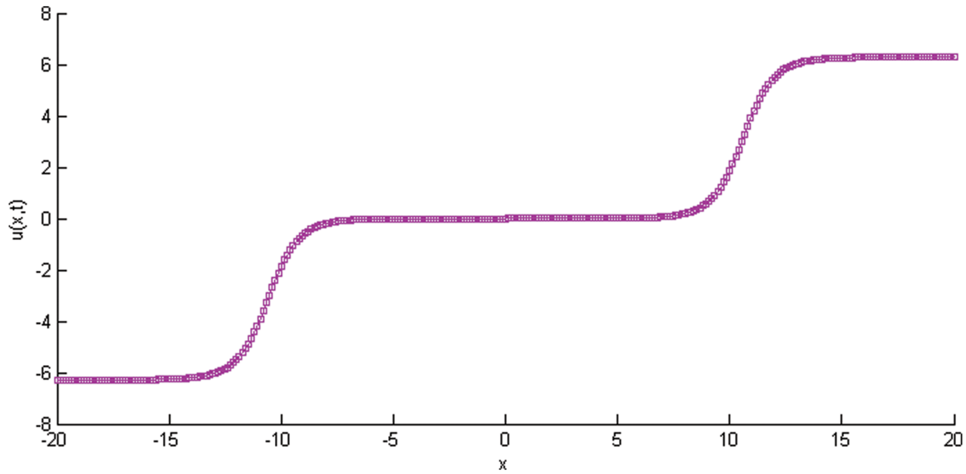


Figure 4: Physical profile of the SG equation example 2 at t = 20

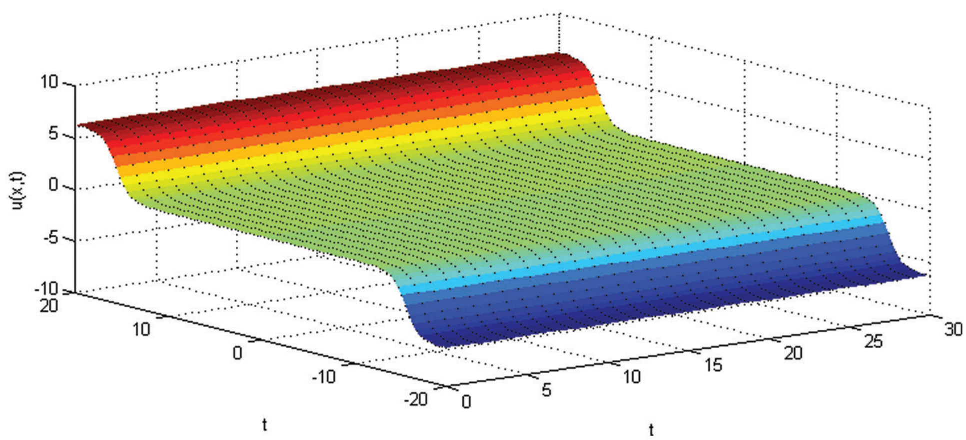


Figure 5: Physical profile of the SG equation example 2 at t = 2

4 Conclusion

Inspired by the success of PSO in optimization tasks, this approach is implemented to obtain an optimal value of the parameter involved in the exponential B-spline basis function to solve the SG equation. The advantage of PSO is to find the parameter value that aids in reducing the error involved in the function evaluations. The PSO-based method is capable of evaluating the parameter in a search space with global search ability. The obtained results are found encouraging and the scheme can be implemented to solve linear and nonlinear partial differential equations efficiently.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proc. of ICNN'95-Int. Conf. on Neural Networks*, Perth, WA, Australia, vol. 4, pp. 1942–1948, 1995.

[2] T. Povich and J. Xin, "A numerical study of the light bullets interaction in the (2 + 1) Sine-Gordon equation," *Journal of Nonlinear Science*, vol. 15, no. 1, pp. 11–25, 2005.

[3] L. D. M. Villari, G. Marcucci, M. C. Braidotti and C. Conti, "Sine-Gordon soliton as a model for hawking radiation of moving black holes and quantum soliton evaporation," *Journal of Physics Communications*, vol. 2, no. 55016, pp. 1–10, 2018.

[4] V. G. Bykov, "Sine-Gordon equation and its application to tectonic stress transfer," *Journal of Seismology*, vol. 18, no. 3, pp. 497–510, 2014.

[5] G. Ben-Yu, P. J. Pascual, M. J. Rodriguez and L. V´azquez, "Numerical solution of the Sine-Gordon equation," *Applied Mathematics and Computation*, vol. 18, no. 1, pp. 1–14, 1986.

[6] M. J. Ablowitz, B. M. Herbst and C. M. Schober, "On the numerical solution of the Sine-Gordon equation, I. Integrable discretization and homoclinic manifolds," *Journal of Computational Physics*, vol. 126, no. 2, pp. 299–314, 1996.

[7] M. J. Ablowitz, B. M. Herbst and C. M. Schober, "On the numerical solution of the Sine-Gordon equation. II. Performance of numerical schemes," *Journal of Computational Physics*, vol. 131, no. 2, pp. 354–367, 1997.

[8] D. Kaya, "An application of the modified decomposition method for two dimensional Sine-Gordon equation," *Applied Mathematics and Computation*, vol. 159, no. 1, pp. 1–9, 2004.

[9] S. S. Ray, "A numerical solution of the coupled Sine-Gordon equation using the modified decomposition method," *Applied Mathematics and Computation*, vol. 175, no. 2, pp. 1046–1054, 2006.

[10] Q. Wang, "An application of the modified adomian decomposition method for (N + 1)-dimensional Sine-Gordon field," *Applied Mathematics and Computation*, vol. 181, no. 1, pp. 147–152, 2006.

[11] U. Yucel, "Homotopy analysis method for the Sine-Gordon equation with initial conditions," *Applied Mathematics and Computation*, vol. 203, no. 1, pp. 387–395, 2008.

[12] M. Dehghan and D. Mirzaei, "The dual reciprocity boundary element method (DRBEM) for two-dimensional Sine-Gordon equation," *Computer Methods in Applied Mechanics and Engineering*, vol. 197, no. 6–8, pp. 476–486, 2008.

[13] M. Dehghan and A. Shokri, "A numerical method for one dimensional nonlinear Sine-Gordon equation using collocation and radial basis functions," *Numerical Methods for Partial Differential Equations*, vol. 24, no. 2, pp. 687–698, 2008.

[14] M. Sari and G. Gurarslan, "A sixth-order compact finite difference method for the one-dimensional Sine-Gordon equation," *International Journal for Numerical Methods in Biomedical Engineering*, vol. 27, no. 7, pp. 1126–1138, 2011.

[15] J. Rashidinia and R. Mohammadi, "Tension spline solution of nonlinear Sine-Gordon equation," *Numerical Algorithms*, vol. 56, no. 1, pp. 129–142, 2011.

[16] R. C. Mittal and R. Bhatia, "Numerical solution of nonlinear Sine-Gordon equation by modified cubic B-spline collocation method," *International Journal of Partial Differential Equations*, vol. 2014, pp. 1–8, 2014.

[17] L. Su, "Numerical solution of two-dimensional nonlinear Sine-Gordon equation using localized method of approximate particular solutions," *Engineering Analysis with Boundary Elements*, vol. 108, no. 2, pp. 95–107, 2019.

[18] M. Lotfi and A. Alipanah, "Legendre spectral element method for solving Sine-Gordon equation," *Advances in Difference Equations*, vol. 2019, no. 1, pp. 1–15, 2019.

[19] D. Adak and S. Natarajan, "Virtual element method for semi linear Sine–Gordon equation over polygonal mesh using product approximation technique," *Mathematics and Computers in Simulation*, vol. 172, pp. 224–243, 2020.

[20] R. Jiwari, "Barycentric rational interpolation and local radial basis functions based numerical algorithms for multidimensional Sine-Gordon equation," *Numerical Methods for Partial Differential Equations*, vol. 37, no. 3, pp. 1965–1992, 2021.

[21] B. K., Singh and M. A. Gupta, "New efficient fourth order collocation scheme for solving sine–Gordon equation," *International Journal of Applied and Computational Mathematics*, vol. 7, no. 4, pp. 1–18, 2021.

[22] M. Shiralizadeh, A. Alipanah and M. Mohammadi, "Numerical solution of one-dimensional Sine-Gordon equation using rational radial basis functions," *Journal of Mathematical Modeling*, Article in press, pp. 1–19, 2021.

[23] R. Bellman, B. G. Kashef and J. Casti, "Differential quadrature: A technique for the rapid solution of nonlinear differential equations," *Journal of Computational Physics*, vol. 10, no. 1, pp. 40–52, 1972.

[24] G. Arora and V. Joshi, "A computational approach for one and two dimensional fisher's equation using quadrature technique," *American Journal of Mathematical and Management Sciences*, vol. 40, no. 2, pp. 145–162, 2021.

[25] G. Arora and B. K. Singh, "Numerical solution of burgers' equation with modified cubic B-spline differential quadrature method," *Applied Mathematics and Computation*, vol. 224, pp. 166–177, 2013.

[26] P. Devshali and G. Arora, "Solution of two-dimensional fractional diffusion equation by a novel hybrid D(TQ) method," *Nonlinear Engineering*, vol. 11, no. 1, pp. 135–142, 2022.

[27] R. C. Mittal and G. Arora, "Efficient numerical solution of fisher's equation by using B-spline method," *International Journal of Computer Mathematics*, vol. 87, no. 13, pp. 3039–3051, 2010.

[28] W. Imtiaz, A. Muhammad, I. M. Kashif and H. A. Mubashir, "Exponential B-spline collocation method for solving the generalized Newell-Whitehead-Segel equation," *Journal of Mathematics and Computer Science*, vol. 20, no. 4, pp. 313–324, 2020.

[29] M. Reza, "Exponential B-spline collocation method for numerical solution of the generalized regularized long wave equation," *Chinese Physics B*, vol. 24, no. 5, pp. 050206, 2015.

[30] I. Dag and O. Ersoy, "The exponential cubic B-spline algorithm for Fisher equation," *Chaos, Solitons and Fractals*, vol. 86, pp. 101–106, 2016.

[31] J. A. Koupaei, M. Firouznia and S. M. Hosseini, "Finding a good shape parameter of RBF to solve PDEs based on the particle swarm optimization algorithm," *Alexandria Engineering Journal*, vol. 57, no. 4, pp. 3641–3652, 2018.

[32] J. Kennedy, "The particle swarm: Social adaptation of knowledge," in *Proc. of IEEE Int. Conf. on Evolutionary Computation*, Indianapolis, IN, USA, pp. 303–308, 1997.

[33] Y. Belkourchia, L. Azrar and E. M. Zeriab, "Hybrid optimization procedure applied to optimal location finding for piezoelectric actuators and sensors for active vibration control," *Applied Mathematical Modelling*, vol. 62, pp. 701–716, 2018.

[34] E. H. Houssein, A. G. Gad, K. Hussain and P. N. Suganthan, "Major advances in particle swarm optimization: Theory, analysis, and application," *Swarm and Evolutionary Computation*, vol. 63, pp. 100868, 2021.

[35] A. P. Piotrowski, J. J. Napiorkowski and A. E. Piotrowska, "Population size in particle swarm optimization," *Swarm and Evolutionary Computation*, vol. 58, pp. 100718, 2020.

[36] H. S. Shukla and M. Tamsir, "Numerical solution of nonlinear sine–Gordon equation by using the modified cubic B-spline differential quadrature method," *Beni-Suef University Journal of Basic and Applied Sciences*, vol. 7, no. 4, pp. 359–366, 2018.

[37] M. Uddin, S. Haqand and G. Qasim, "A meshfree approach for the numerical solution of nonlinear Sine-Gordon equation," *International Mathematical Forum*, vol. 7, no. 24, pp. 1179–1186, 2012.