

Numerical results for the nonlinear Schrödinger equation via a cubic Hermite spline method

Murat Arı 

Department of Mathematics, Karamanoğlu Mehmetbey University, Karaman, Turkey 

Article History:

- received October 15, 2025
- revised March 18, 2026
- accepted May 20, 2026

Abstract. In this paper, we investigate the numerical solution of the nonlinear Schrödinger (NLS) equation using a cubic Hermite spline collocation method. The proposed scheme is applied to various test problems, including single soliton propagation, soliton collisions, and both standing and traveling soliton solutions, in order to evaluate its accuracy and efficiency. A Von Neumann stability analysis is performed for the linearized form of the scheme, demonstrating that the method is unconditionally stable in the linear sense. Numerical experiments demonstrate that the method provides stable and accurate approximations, with good agreement between the computed and exact solutions. The results confirm that the cubic Hermite spline approach is a reliable tool for solving the NLS equation, particularly in capturing the qualitative behavior of the wave profiles. Comparisons with existing numerical methods further highlight the effectiveness of the proposed technique.

Keywords: cubic Hermite spline; collocation method; nonlinear Schrödinger equation; Legendre and Chebyshev roots.

AMS Subject Classification: 65D07; 65N20.

✉ Corresponding author. E-mail: muratari@kmu.edu.tr

1 Introduction

The nonlinear Schrödinger (NLS) equation is one of the most important nonlinear partial differential equations in mathematical physics, with wide applications in fields such as nonlinear optics, plasma physics, fluid dynamics. It describes the evolution of slowly varying wave packets in nonlinear dispersive media and has been extensively studied both analytically and numerically. Due to its ability to capture soliton dynamics and other nonlinear wave phenomena, the NLS equation has become a prototype model in nonlinear science.

Despite the availability of some analytical solutions for specific cases, most realistic problems require efficient and accurate numerical methods. Over the past decades, numerous numerical schemes have been developed for solving the

Copyright © 2026 The Author(s). Published by Vilnius Gediminas Technical University

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

NLS equation. Early numerical approaches heavily relied on finite-difference and finite-element methods; for instance, Delfour et al. [4] introduced a finite-difference method that satisfies a discrete analogue of the equation's conservation laws to model soliton formation. Similarly, Twizell et al. [28] developed a family of finite-difference methods that transform the problem into a first-order linear initial-value problem, providing detailed stability and convergence analyses.

B-spline finite element methods have also been extensively utilized to ensure continuous derivatives across the solution range. Dağ [2] obtained numerical solutions using a quadratic B-spline finite element method, while Gardner et al. [11,12] proposed an approach based on the collocation of cubic B-splines over spatial finite elements. In these studies, time integration was handled using a Crank-Nicolson approximation [11] and a leapfrog algorithm [12], both demonstrating excellent conservation properties. Higher-order approximations, such as the quintic B-spline collocation method developed by Saka [27], have further proven their efficiency and robustness in modeling single and interacting solitons. In addition to splines, orthogonal spline collocation semidiscretizations have been examined by Robinson and Fairweather [26] for the cubic Schrödinger equation, which Robinson [25] later extended to NLS equations with general power nonlinearities, emphasizing the preservation of theoretical invariants. Differential quadrature (DQ) algorithms have also shown high accuracy for NLS simulations; Korkmaz and Dağ [15,16] successfully simulated soliton dynamics using both cosine expansion-based DQ [15] and polynomial-based DQ coupled with a fourth-order Runge-Kutta scheme [16].

From an alternative standpoint, meshless methods have gained significant traction for their flexibility. Dereli et al. [6] presented a meshless method based on collocation with radial basis functions (RBFs), which Dereli [5] expanded upon by evaluating the meshless kernel-based method of lines with Multiquadric, Gaussian, and Wendland's functions. Furthermore, Kaplan and Dereli [13] applied a moving least-squares approximation meshless method, conducting thorough stability analyses and calculating error norms. Collectively, these studies provide accurate, stable and conservation-compatible approximations for simulating soliton propagation and interaction in the NLS equation.

In addition to numerical approaches, analytical techniques have been developed to obtain exact soliton solutions. Khuri [14] introduced a complex tanh-function method for constructing travelling wave and multiple soliton solutions of Schrödinger-type equations. Öziş and Yıldırım [23] applied He's modified Lindstedt-Poincaré method to derive dispersion-managed exact soliton solutions expressed in trigonometric and hyperbolic forms.

Overall, the existing literature demonstrates that finite-difference, spline-based finite element, orthogonal spline collocation, meshless radial basis function and differential quadrature methods are all effective for solving the NLS equation. Among these, spline-based and meshless collocation techniques stand out due to their smoothness, high-order convergence and favorable conservation properties, making them particularly suitable for accurately simulating soliton dynamics.

While the above discussion has primarily focused on classical spline formu-

lations, in the present study we turn our attention to Hermite spline functions, which have proven highly effective in the numerical solution of various other nonlinear evolution equations. In particular, cubic Hermite splines are advantageous due to their built-in continuity properties and their suitability for approximating both the function and its derivatives. Several works in the literature have reported promising results using hermite spline functions for nonlinear wave equations [1, 7, 8, 9, 10, 17, 18, 19, 20, 21, 22, 24], motivating further exploration of this approach for the NLS equation.

In this paper, we apply a cubic Hermite spline (CHS) method to the NLS equation and present detailed numerical results. The selection of CHS is motivated by the necessity to balance numerical precision with computational efficiency. While lower-order approximations often provide insufficient convergence for capturing complex wave phenomena, the CHS approach naturally preserves both the solution and its gradient at each node. Furthermore, CHS avoids the excessive algorithmic complexity and additional boundary constraints associated with higher-order alternatives, maintaining a more efficient matrix structure that is better suited for the NLS equation.

The main aim is to demonstrate the applicability, accuracy, and efficiency of the proposed scheme through a set of benchmark problems. Numerical experiments are carried out to validate the method and to compare its performance with existing approaches. The results confirm that the cubic Hermite spline method provides reliable and accurate approximations, making it a useful tool for the numerical treatment of the NLS equation.

The general form of the nonlinear Schrödinger (NLS) equation can be expressed as

$$\begin{aligned} iU_t + U_{xx} + q|U|^2U &= 0, \quad (x, t) \in [a, b] \times [0, T], \\ U(x, 0) &= g(x), \end{aligned} \quad (1.1)$$

subject to the boundary conditions

$$U(a, t) = 0, \quad U(b, t) = 0,$$

where $i = \sqrt{-1}$ and $q \geq 0$ is a real constant [3]. The function $U = U(x, t)$ represents a complex-valued field variable.

To simplify the numerical analysis of Equation (1.1), the complex variable $U(x, t)$ is partitioned into its real and imaginary parts as

$$U(x, t) = r(x, t) + i s(x, t),$$

where $r(x, t)$ and $s(x, t)$ represent real-valued functions that define the real and imaginary components, respectively. Substituting this representation into Equation (1.1) and separating the real and imaginary parts yield the following coupled system of real partial differential equations:

$$\begin{aligned} r_t + s_{xx} + q(r^2 + s^2)s &= 0, \\ s_t - r_{xx} - q(r^2 + s^2)r &= 0. \end{aligned} \quad (1.2)$$

The initial and boundary conditions prescribed for the system (1.2) are formulated as follows,

$$\begin{aligned} r(x, 0) &= g_r(x), & s(x, 0) &= g_s(x), & x &\in [a, b], \\ r(a, t) &= r(b, t) = 0, & s(a, t) &= s(b, t) = 0, & t &\in [0, T], \end{aligned}$$

where $g_r(x)$ and $g_s(x)$ are respectively the real and imaginary components of the initial function $g(x)$.

After this transformation, the cubic Hermite spline approximation is applied separately to the real-valued functions $r(x, t)$ and $s(x, t)$. In this way, the complex-valued NLS equation is handled through a coupled system of real equations while preserving the polynomial approximation framework.

2 The cubic Hermite collocation method

Cubic Hermite splines are piecewise cubic polynomial functions that interpolate both the function values and their first derivatives at the grid points. Let $a = x_1 < x_2 < \dots < x_{N+1} = b$ be a partition of the interval $[a, b]$. The cubic Hermite interpolant of a function f is a function $s(x)$ such that on each subinterval $[x_k, x_{k+1}]$ it coincides with a cubic polynomial $s_k(x)$.

The spline satisfies the interpolation conditions

$$s(x_k) = f(x_k), \quad s'(x_k) = f'(x_k), \quad k = 1, 2, \dots, N+1,$$

which ensures that both the function and its first derivative are matched at the nodes. Moreover, the continuity conditions

$$s_k(x_{k+1}) = s_{k+1}(x_{k+1}), \quad s'_k(x_{k+1}) = s'_{k+1}(x_{k+1})$$

guarantee that $s(x)$ and its first derivative are continuous on the entire interval $[a, b]$, that is, $s(x) \in C^1[a, b]$.

Consequently, the cubic polynomial on each subinterval can be expressed in terms of the function and derivative values at the end points as

$$\begin{aligned} s_k(x) &= H_1^k(x)f(x_k) + H_2^k(x)f'(x_k) \\ &\quad + H_3^k(x)f'(x_{k+1}) + H_4^k(x)f(x_{k+1}), \end{aligned}$$

where H_j are the cubic Hermite basis functions that satisfy the essential interpolation properties defined by [7, 8, 9]:

$$\begin{aligned} H_{2j-1}(x) &= \frac{1}{h^3} \begin{cases} (x - x_{j-1})^2 (3h - 2(x - x_{i-1})), & x \in [x_{j-1}, x_j], \\ (h - (x - x_j))^2 (h - 2(x - x_j)), & x \in [x_j, x_{j+1}], \\ 0, & \text{otherwise,} \end{cases} \\ H_{2j}(x) &= \frac{1}{h^3} \begin{cases} -h(x - x_{j-1})^2 (h - x - x_{j-1}), & x \in [x_{j-1}, x_j], \\ h(h - (x - x_j))^2 (x - x_j), & x \in [x_j, x_{j+1}], \\ 0, & \text{otherwise,} \end{cases} \\ j &= 1, 2, 3, \dots, N+1. \end{aligned}$$

These properties provide smooth C^1 approximations with local support, which makes cubic Hermite splines particularly suitable for numerical schemes applied to nonlinear wave equations.

The approximate solution to $u(x, t)$ is represented in terms of cubic Hermite basis functions as

$$u(x, t) = \sum_{j=1}^{N+1} a_{j+2k-2}(t) H_{ij},$$

where $a_j(t)$ are time-dependent coefficients to be determined, k denotes the number of elements, and $i = 1, 2$. Within each sub-interval $[x_j, x_{j+1}]$, the Gauss-Legendre quadrature points $\eta_{j,r}$ of order two are employed as collocation points. These roots are selected as collocation points because they minimize the numerical integration error and enhance the overall accuracy of the method. Unlike equally spaced nodes, these points provide better control of interpolation error growth and significantly reduce spurious oscillations. Their non-uniform distribution, characterized by clustering near the boundaries, improves numerical stability and enhances the conditioning of the resulting discrete system. These properties make them particularly suitable for nonlinear wave equations, where stability and accurate resolution of wave profiles are essential.

$$\eta_{j,r} = \left(\frac{x_{j-1} + x_j}{2} \right) + (-1)^r \frac{h_j}{2\sqrt{3}}, 2 \leq j \leq n+1, 1 \leq r \leq 2. \quad (2.1)$$

By taking $\lambda_1 = \frac{1}{2} \left(1 - \frac{1}{\sqrt{3}} \right)$ and $\lambda_2 = \frac{1}{2} \left(1 + \frac{1}{\sqrt{3}} \right)$ (shifted Legendre roots) in Equation (2.1), we obtain

$$\begin{aligned} \frac{\eta_{j,1} - x_j}{h_j} &= -\lambda_1, & \frac{\eta_{j,2} - x_j}{h_j} &= -\lambda_2, \\ \frac{\eta_{j,1} - x_{j-1}}{h_j} &= \lambda_1, & \frac{\eta_{j,2} - x_{j-1}}{h_j} &= \lambda_2. \end{aligned}$$

For Chebyshev polynomial $\lambda_1 = \frac{1}{2} \left(1 - \frac{1}{\sqrt{2}} \right)$ and $\lambda_2 = \frac{1}{2} \left(1 + \frac{1}{\sqrt{2}} \right)$.

Following the discretization process, each element is associated with a local coordinate ξ , where the relation between x and ξ is given by $\xi = \frac{x-x_k}{h}$. Through this transformation, the varying global intervals $[x_k, x_{k+1}]$ are consistently projected onto a normalized local interval of $[0, 1]$ for all elements. Finally, the transformation $x = h\xi + x_k$ is used to express the subsequent equations in terms of the local coordinate system.

$$\begin{aligned} H_1(\xi) &= (1 + 2\xi)(1 - \xi)^2, & H_2(\xi) &= \xi(1 - \xi)^2 h, \\ H_3(\xi) &= \xi^2(3 - 2\xi), & H_4(\xi) &= \xi^2(\xi - 1)h, \\ A_1(\xi) &= 6\xi^2 - 6\xi, & A_2(\xi) &= (1 - 4\xi + 3\xi^2) h, \\ A_3(\xi) &= 6\xi - 6\xi^2, & A_4(\xi) &= (3\xi^2 - 2\xi) h, \\ B_1(\xi) &= 12\xi - 6, & B_2(\xi) &= (6\xi - 4)h, \\ B_3(\xi) &= 6 - 12\xi, & B_4(\xi) &= (6\xi - 2)h. \end{aligned}$$

Thus, the trial function over the k^{th} element are defined as

$$u_N(x, t) = \sum_{j=1}^N a_{j+2k-2}(t) H_{ji}.$$

The trial functions with their first and second order derivatives at the collocation points in terms of local variable ξ are defined as follows

$$\begin{aligned} u_N(\xi, t) &= \sum_{j=1}^4 a_{j+2k-2}(t) H_j(\xi), \\ &= a_{2k-1} H_1(\xi) + a_{2k} H_2(\xi) + a_{2k+1} H_3(\xi) + a_{2k+2} H_4(\xi), \\ u'_N(\xi, t) &= \frac{1}{h} \sum_{j=1}^4 a_{j+2k-2}(t) A_j(\xi), \\ &= \frac{1}{h} [a_{2k-1} A_1(\xi) + a_{2k} A_2(\xi) + a_{2k+1} A_3(\xi) + a_{2k+2} A_4(\xi)], \\ u''_N(\xi, t) &= \frac{1}{h^2} \sum_{j=1}^4 a_{j+2k-2}(t) B_j(\xi), \\ &= \frac{1}{h^2} [a_{2k-1} B_1(\xi) + a_{2k} B_2(\xi) + a_{2k+1} B_3(\xi) + a_{2k+2} B_4(\xi)]. \end{aligned}$$

Here A_1, A_2, A_3, A_4 ve B_1, B_2, B_3, B_4 represent the first and second derivatives of the Hermite basis functions, respectively. By applying Equations (2.4) and (2.5) at the nodal points, one obtains the following approximate solutions:

$$\begin{aligned} u_i &= u_N(\xi_i, t) = a_{2k-1} H_{1i} + a_{2k} H_{2i} + a_{2k+1} H_{3i} + a_{2k+2} H_{4i}, \\ hu'_i &= u'_N(\xi_i, t) = a_{2k-1} A_{1i} + a_{2k} A_{2i} + a_{2k+1} A_{3i} + a_{2k+2} A_{4i}, \\ h^2 u''_i &= u''_N(\xi_i, t) = a_{2k-1} B_{1i} + a_{2k} B_{2i} + a_{2k+1} B_{3i} + a_{2k+2} B_{4i}, \end{aligned}$$

where $H_{ji} = H_j(\xi)$, $A_{ji} = A_j(\xi)$ and $B_{ji} = B_j(\xi)$ for $i = 1, 2$.

In the developed numerical scheme, the time integration is carried out by applying a Crank-Nicolson finite difference formulation, while the spatial discretization is achieved through a collocation technique utilizing cubic Hermite spline basis functions. The use of these functions ensures high computational efficiency owing to their compact support, simple structure, and minimal storage requirements. Moreover, the resulting discrete algebraic equations exhibit favorable numerical properties such as stability, good conditioning, and reduced sensitivity to rounding errors. The associated system matrices are sparse and banded, making them particularly suitable for fast and efficient implementation.

Although cubic Hermite splines theoretically provide fourth-order accuracy in space for sufficiently smooth functions, the overall convergence rate of the proposed scheme is influenced by the time discretization and the treatment of the nonlinear term. In particular, the Crank-Nicolson scheme is second order accurate in time, and the applied linearization further affects the global accuracy. Therefore, the observed convergence rate in the numerical experiments is approximately $O(h^2)$, which is consistent with similar studies in the literature.

3 Implementation of the method

Integrating the Crank-Nicolson discretization into Equation (1.2) yields the following expression:

$$\begin{aligned} \frac{s^{n+1} - s^n}{\Delta t} - \frac{r_{xx}^{n+1} + r_{xx}^n}{2} - q \left((r^n)^2 + (s^n)^2 \right) \left(\frac{r^{n+1} + r^n}{2} \right) &= 0, \\ \frac{r^{n+1} - r^n}{\Delta t} - \frac{s_{xx}^{n+1} + s_{xx}^n}{2} + q \left((r^n)^2 + (s^n)^2 \right) \left(\frac{s^{n+1} + s^n}{2} \right) &= 0. \end{aligned}$$

It should be noted that the nonlinear terms contain the factor $(r^2 + s^2)$, which depends on the unknown solution. A direct Crank-Nicolson discretization would therefore result in a nonlinear algebraic system at each time step. To avoid solving a nonlinear system, the nonlinear coefficient is evaluated using the solution at the previous time level. Introducing

$$z_m = q \left((r^n)^2 + (s^n)^2 \right),$$

the nonlinear terms can be linearized by means of a Taylor series expansion. After this linearization, the resulting system can be written as

$$\begin{aligned} 2s^{n+1} - \Delta t r_{xx}^{n+1} - \Delta t z_m r^{n+1} &= 2s^n + \Delta t r_{xx}^n + \Delta t z_m r^n, \\ 2r^{n+1} + \Delta t s_{xx}^{n+1} + \Delta t z_m s^{n+1} &= 2r^n - \Delta t s_{xx}^n - \Delta t z_m s^n. \end{aligned}$$

The stability of the time discretization should also be noted. The Crank-Nicolson method is well known to be second order accurate in time and unconditionally stable for linear problems. In the present formulation, the nonlinear coefficient z_m is evaluated using the solution at the previous time level, which results in a semi-implicit scheme. This approach avoids solving a nonlinear system at each time step while preserving the favorable stability properties of the Crank-Nicolson discretization. The numerical experiments presented in the following section further confirm the stable behavior of the proposed scheme. For each time iteration, by substituting

$$\begin{aligned} r(x, t) &= \sum_{j=1}^{N+1} a_{j+2k-2}(t) H_{ji}, \quad i = 1, 2, \\ s(x, t) &= \sum_{j=1}^{N+1} b_{j+2k-2}(t) H_{ji}, \quad i = 1, 2, \end{aligned}$$

the following form of the linear system of equations is obtained:

$$\begin{aligned}
 & 2H_{1i}a_{2k-1}^{n+1} + 2H_{2i}a_{2k}^{n+1} + 2H_{3i}a_{2k+1}^{n+1} + 2H_{4i}a_{2k+2}^{n+1} + \left(\frac{\Delta t}{h^2}B_{1i} + \Delta t z_m H_{1i}\right)b_{2k-1}^{n+1} \\
 & + \left(\frac{\Delta t}{h^2}B_{2i} + \Delta t z_m H_{2i}\right)b_{2k}^{n+1} + \left(\frac{\Delta t}{h^2}B_{3i} + \Delta t z_m H_{3i}\right)b_{2k+1}^{n+1} + \left(\frac{\Delta t}{h^2}B_{4i} \right. \\
 & + \Delta t z_m H_{4i})b_{2k+2}^{n+1} = 2H_{1i}a_{2k-1}^n + 2H_{2i}a_{2k}^n + 2H_{3i}a_{2k+1}^n + 2H_{4i}a_{2k+2}^n \\
 & + \left(-\frac{\Delta t}{h^2}B_{1i} - \Delta t z_m H_{1i}\right)b_{2k-1}^n + \left(-\frac{\Delta t}{h^2}B_{2i} - \Delta t z_m H_{2i}\right)b_{2k}^n \\
 & + \left(-\frac{\Delta t}{h^2}B_{3i} - \Delta t z_m H_{3i}\right)b_{2k+1}^n + \left(-\frac{\Delta t}{h^2}B_{4i} - \Delta t z_m H_{4i}\right)b_{2k+2}^n, \\
 & \left(-\frac{\Delta t}{h^2}B_{1i} - \Delta t z_m H_{1i}\right)a_{2k-1}^{n+1} + \left(-\frac{\Delta t}{h^2}B_{2i} - \Delta t z_m H_{2i}\right)a_{2k}^{n+1} \left(-\frac{\Delta t}{h^2}B_{3i} \right. \\
 & - \Delta t z_m H_{3i})a_{2k+1}^{n+1} + \left(-\frac{\Delta t}{h^2}B_{4i} - \Delta t z_m H_{4i}\right)a_{2k+2}^{n+1} + 2H_{1i}b_{2k-1}^{n+1} + 2H_{2i}b_{2k}^{n+1} \\
 & + 2H_{3i}b_{2k+1}^{n+1} + 2H_{4i}b_{2k+2}^{n+1} = \left(\frac{\Delta t}{h^2}B_{1i} + \Delta t z_m H_{1i}\right)a_{2k-1}^n + \left(\frac{\Delta t}{h^2}B_{2i} \right. \\
 & + \Delta t z_m H_{2i})a_{2k}^n + \left(\frac{\Delta t}{h^2}B_{3i} + \Delta t z_m H_{3i}\right)a_{2k+1}^n + \left(\frac{\Delta t}{h^2}B_{4i} + \Delta t z_m H_{4i}\right)a_{2k+2}^n \\
 & + 2H_{1i}b_{2k-1}^n + 2H_{2i}b_{2k}^n + 2H_{3i}b_{2k+1}^n + 2H_{4i}b_{2k+2}^n, \quad i = 1, 2.
 \end{aligned} \tag{3.1}$$

A solvable $4N \times 4N$ matrix system is obtained by imposing boundary conditions on the recursive system (3.1), which originally contains $4N$ equations and $4N + 4$ unknown variables. This process effectively eliminates the four redundant parameters, resulting in:

$$Lx^{n+1} = Rx^n,$$

where L, R coefficient matrices and, $x^n = (a_1^n, a_2^n, \dots, a_{2N}^n, b_1^n, b_2^n, \dots, b_{2N}^n)^T$ are time dependent unknown vectors to be determined. Given that each cubic Hermite basis function spans a maximum of four sub-intervals, a band matrix with a bandwidth of four is generated (see Figure 1). In order to start the iteration, the initial vectors a^0 and b^0 are needed. This vectors are calculated by the initial condition presented by the governing equation.

3.1 Algorithm of the method

The computational steps for executing the proposed algorithm are structured in the following manner:

1. The spatial domain is discretized into a finite number of elements, and suitable collocation points (Legendre or Chebyshev roots) are selected.
2. The complex-valued problem is transformed into a coupled real system and approximated using cubic Hermite spline basis functions.

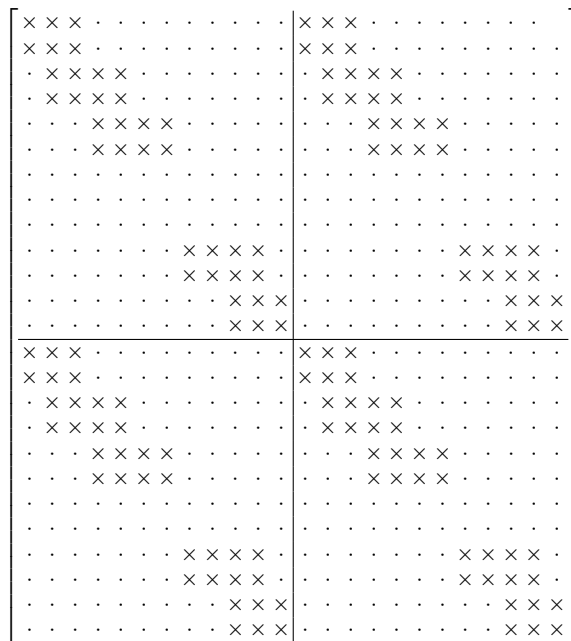


Figure 1. Pattern of nonzero elements of banded matrices L and R .

3. The corresponding system matrices L and R are assembled. Using the prescribed initial condition $u(x, 0)$, the initial coefficient vector x^0 is computed.
4. Set the initial time level $n = 0$ and start the time-marching process.
5. At each time step, update (or recompute when necessary) the matrices L , R , and the coefficient vector according to the numerical formulation, including the treatment of the nonlinear terms.
6. Compute the unknown coefficients at the next time level by solving the linear system

$$Lx^{n+1} = Rx^n.$$

7. Advance the time level ($n \leftarrow n + 1$) and repeat the time-marching procedure until the prescribed final time is reached.
8. After completing the time integration, reconstruct the complex solution from its real and imaginary components, and evaluate the numerical results through appropriate error norms and graphical analysis.

4 Stability analysis

In this section, Von-Neumann stability analysis is to be used to investigate stability of the linear numerical scheme. Since this stability technique applies only

linear differential equations, nonlinear terms in Equation (1.2) is changed by a constant λ and the coupled system of is written as

$$\frac{\partial U}{\partial t} + A \frac{\partial^2 U}{\partial x^2} + \lambda AU = 0, \quad (4.1)$$

where $U = \begin{bmatrix} r \\ s \end{bmatrix}$ and $A = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$. By applying the Crank-Nicolson method to Equation (4.1) and reorganizing we get

$$U^{n+1} + \frac{A\Delta t}{2} \left(\frac{\partial^2 U}{\partial x^2} \right)^{n+1} + \frac{\lambda A\Delta t}{2} U^{n+1} = U^n - \frac{A\Delta t}{2} \left(\frac{\partial^2 U}{\partial x^2} \right)^n - \frac{\lambda A\Delta t}{2} U^n. \quad (4.2)$$

Now we apply cubic Hermite collocation

$$U(x, t) = \sum_{j=1}^{N+1} c_{j+2k-2}(t) H_{ji}, \quad i = 1, 2$$

to the Equation (4.2) we get

$$\sigma_1 c_{2k-1}^{n+1} + \sigma_2 c_{2k}^{n+1} + \sigma_3 c_{2k+1}^{n+1} + \sigma_4 c_{2k+2}^{n+1} = \tau_1 c_{2k-1}^n + \tau_2 c_{2k}^n + \tau_3 c_{2k+1}^n + \tau_4 c_{2k+2}^n, \quad (4.3)$$

where

$$\begin{aligned} \sigma_m &= IH_{mi} + \frac{A\Delta t}{2} B_{mi} + \frac{\lambda A\Delta t}{2} H_{mi}, \\ \tau_m &= IH_{mi} - \frac{A\Delta t}{2} B_{mi} - \frac{\lambda A\Delta t}{2} H_{mi}, \quad m = 1, 2, 3, 4. \end{aligned}$$

In the von Neumann stability analysis, we assume the solution to be of the form:

$$c_j^n = \xi^n \gamma e^{ij\varphi}, \quad (4.4)$$

where $i = \sqrt{-1}$, $\gamma \in \mathbb{R}^2$ and $\xi \in \mathbb{R}^{2 \times 2}$ is the amplification matrix.

Substituting Equation (4.4) into Equation (4.3) we obtain the following relationship:

$$\xi^{n+1} \left(\sigma_1 e^{-i\varphi} + \sigma_2 + \sigma_3 e^{i\varphi} + \sigma_4 e^{i2\varphi} \right) = \xi^n \left(\tau_1 e^{-i\varphi} + \tau_2 + \tau_3 e^{i\varphi} + \tau_4 e^{i2\varphi} \right).$$

Using the Euler identities the above expressions can be written as:

$$\begin{aligned} S_1 + iS_2 &= \sigma_2 + (\sigma_1 + \sigma_3) \cos \varphi + \sigma_4 \cos 2\varphi + i[(\sigma_3 - \sigma_1) \sin \varphi + \sigma_4 \sin 2\varphi] \\ S_3 + iS_4 &= \tau_2 + (\tau_1 + \tau_3) \cos \varphi + \tau_4 \cos 2\varphi + i[(\tau_3 - \tau_1) \sin \varphi + \tau_4 \sin 2\varphi]. \end{aligned}$$

Hence the amplification matrix is obtained as

$$\xi = (S_1 + iS_2)^{-1} (S_3 + iS_4).$$

Since the coefficients σ_m and τ_m contain the matrix

$$A = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix},$$

which is skew symmetric, the matrices involved in the amplification relation preserve this property. Consequently, the matrices $(S_1 + iS_2)$ and $(S_3 + iS_4)$ possess the same eigenvalue structure. Hence the spectral radius of the amplification matrix satisfies

$$\rho(\xi) = 1.$$

Therefore, the proposed linearized scheme is unconditionally stable.

5 Numerical examples

In this section, numerical experiments are carried out for four benchmark problems governed by the nonlinear Schrödinger (NLS) equation. To ensure the reliability of the computed solutions, it is crucial that the numerical scheme preserves the discrete analogues of the continuous conservation laws associated with the equation. These invariants play a key role in maintaining the smoothness and stability of the soliton profiles during time evolution. The NLS equation admits the following conserved quantities:

$$C_1 = \int_a^b |U|^2 dx, \quad C_2 = \int_a^b \left(|U_x|^2 - \frac{1}{2}q|U|^4 \right) dx.$$

The performance and accuracy of the proposed method are assessed through two standard error norms: the discrete L_2 and L_∞ measures, defined respectively as

$$L_2 = \sqrt{h \sum_{j=1}^N |u_j^{\text{exact}} - u_j^{\text{num}}|^2}, \quad L_\infty = \max_{1 \leq j \leq N} |u_j^{\text{exact}} - u_j^{\text{num}}|.$$

Example 1. As an initial validation test, we examine the analytical one-soliton solution of the nonlinear Schrödinger (NLS) equation. Following the formulation presented in [11], the exact solution can be expressed as

$$U(x, t) = \alpha \left(\frac{2}{q} \right)^{1/2} \exp \left[i \left(\frac{1}{2}Sx - \frac{1}{4}(S^2 - \alpha^2)t \right) \right] \text{sech}(\alpha(x - St)),$$

where S represents the velocity of the soliton, determined by the amplitude parameter α . Numerical experiments were performed with the parameter set $q = 2$, $S = 4$, and $\alpha = 1$ over the spatial domain $-20 \leq x \leq 20$. With these parameters, the envelope of the soliton is given by

$$|U| = \text{sech}(x - 4t),$$

which travels rightward with constant shape and velocity equal to 4. The solitary wave profile, along with its real and imaginary parts, is displayed in Figure 2. The numerical outcomes obtained using Legendre basis functions are visually identical to those produced with Chebyshev functions.

The analytical expressions of the conserved invariants yield $C_1 = 2$ and $C_2 = 7.333333$. The computation is advanced up to the final time $t = 2.5$, and the corresponding error norms together with the invariant quantities are listed in Table 1. The numerical results demonstrate excellent agreement with the analytical predictions, confirming the reliability and precision of the proposed approach.

Table 1. Comparison of error norms of NLS for two methods (QHCM-L and QHCM-C) with some previous methods at $t = 2.5$.

Method	h	Δt	L_2 Error	L_∞ Error	C_1	C_2
CHCM-L	0.05	0.0005	8.147×10^{-3}	3.711×10^{-3}	1.999999	7.312040
CHCM-C	0.05	0.0005	3.749×10^{-3}	1.576×10^{-3}	2.000000	7.312118
CHCM-L	0.1	0.0001	1.581×10^{-3}	7.213×10^{-4}	1.999999	7.329006
CHCM-C	0.1	0.0001	2.492×10^{-2}	1.095×10^{-2}	2.000000	7.329084
CHCM-L	0.3125	0.0001	4.264×10^{-3}	1.839×10^{-3}	1.999999	7.323042
CHCM-C	0.3125	0.0001	2.392×10^{-1}	1.027×10^{-1}	2.000027	7.322541
G [6]	0.3125	0.001	4.6×10^{-5}	2.8×10^{-5}	1.999908	7.333177
MQ [6]	0.3125	0.001	4.434×10^{-3}	2.165×10^{-3}	1.999472	7.331960
IMQ [6]	0.3125	0.001	6.68×10^{-4}	4.86×10^{-4}	1.999137	7.329795
IQ [6]	0.3125	0.001	5.652×10^{-3}	2.037×10^{-3}	1.999812	7.329801
Quin. B-Spline [27]	0.05	0.005	9×10^{-4}	6×10^{-4}	2.000000	7.333333

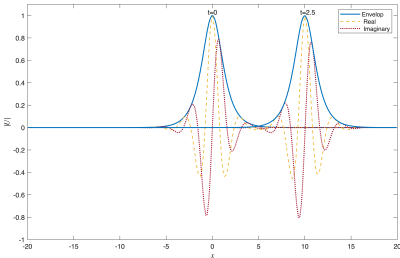


Figure 2. Single soliton simulation.

Example 2. As the second test problem, the nonlinear interaction between two solitons is considered. The initial condition describing the soliton collision is expressed as

$$U(x, 0) = U_1(x, 0) + U_2(x, 0),$$

where each soliton component $U_j(x, 0)$ is defined by

$$U_j(x, 0) = \alpha_j \left(\frac{2}{q}\right)^{1/2} \exp\left[i\frac{S}{2}(x - x_j)\right] \operatorname{sech}(\alpha_j(x - x_j)), \quad j = 1, 2.$$

The computational parameters are assigned as $q = 2$, $\alpha_1 = 1$, $S_1 = -4$, $x_1 = -10$, $\alpha_2 = 1$, $S_2 = 4$, $x_2 = 10$, $h = 0.4$, and $\Delta t = 0.001$ within the spatial interval $-20 \leq x \leq 20$.

This configuration is employed to examine the nonlinear behavior and phase variation that take place when two solitons interact with each other. The numerical experiment is performed up to the final time $t = 6$. The prescribed initial profile describes two solitary waves that are initially separated by a distance of 20 units. One soliton with amplitude 1.0 is centered at $x = -10$, while the other of identical amplitude is located at $x = 10$. The time evolution of the elastic collision between two waves is illustrated in Figure 3. As the simulation advances, both solitons propagate in opposite directions, undergo a collision, and then move apart again. After the interaction, each soliton retains its original amplitude and shape.

The computed invariant quantities are summarized in Table 2, and compared some previous results.

Table 2. Comparison of invariants of two soliton problem at time $t = 6$.

Method	h	Δt	C_1	C_2
CHCM-L	0.3125	0.0001	4.00000	14.63634
CHCM-C	0.3125	0.0001	4.00005	14.62701
Gaussian [13]	0.1	0.01	3.99999	14.66715
W(4,2) [13]	0.1	0.01	3.99999	14.66874
W(5,3) [13]	0.1	0.01	4.00000	14.66654
W(6,4) [13]	0.1	0.01	4.00000	14.66679
W(7,5) [13]	0.1	0.01	3.99999	14.66665
G [6]	0.3125	0.001	4.00040	14.66846
MQ [6]	0.3125	0.001	4.00637	14.69381
IMQ [6]	0.3125	0.001	4.00616	14.69505
IQ [6]	0.3125	0.001	3.99613	14.65291
Quin. B-Spline [27]	0.3125	0.001	4.00000	14.66672

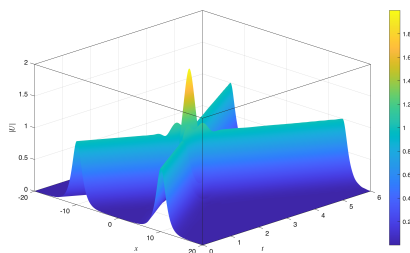


Figure 3. Collision of two solitons.

Example 3. In this numerical experiment, the formation of a standing soliton is examined. The formation of such a soliton occurs when the following condition is satisfied:

$$\int_{-\infty}^{\infty} U(x, 0) dx \geq \pi.$$

This phenomenon is investigated using the following Maxwellian-type initial

condition

$$U(x, 0) = Ae^{-x^2}. \quad (5.1)$$

When the amplitude A of the Maxwellian initial condition is smaller than $\sqrt{\pi}$, the solution gradually diminishes over time, indicating that no soliton is generated. This phenomenon, known as soliton decay, has been verified by several numerical methods in [2, 5, 6, 11, 27]. Conversely, when $A \geq \sqrt{\pi} = 1.7725$, a stable soliton structure is formed. In this example, we will also observe this characteristic behavior by employing our proposed method.

In this example, the evolution of a stationary solution is observed by adopting the initial condition (5.1) with the parameters $q = 2$, $h = 0.25$, and $\Delta t = 0.001$ within the computational domain $-45 \leq x \leq 45$ up to the final time $t = 6$. Simulations are carried out for two amplitude values, $A = 1$ and $A = 1.78$. The temporal evolution of the standing wave formation for these amplitudes is depicted in Figure 4.

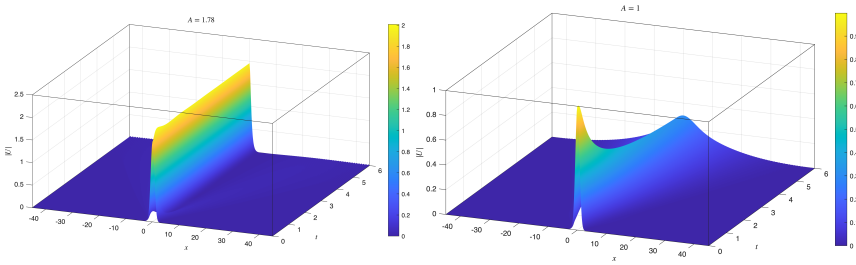


Figure 4. Formation of a standing soliton for $A = 1.78$ and $A = 1$.

The analytical expressions of the conserved quantities are given by

$$C_1 = A^2 \sqrt{\frac{\pi}{2}}, \quad C_2 = \frac{1}{4} A^2 (2\sqrt{2} - qA^2) \sqrt{\pi}.$$

Accordingly, for $A = 1$, the computed results are $C_1 = 1.2533141373$ and $C_2 = 0.3670872118$, whereas for $A = 1.78$, they are $C_1 = 3.9710005126$ and $C_2 = -4.9256176213$. The corresponding numerical estimations and a comparative analysis of the invariants, along with their relative variations at $t = 6$, are summarized in Table 5. The numerical results in this table indicate that soliton formation depends on the magnitude of the initial pulse amplitude and is associated with a physically meaningful threshold. From a physical standpoint, the emergence of a stable soliton requires a balance between dispersive spreading and nonlinear self-focusing effects. When the initial power or norm of the pulse remains below a certain level, dispersion dominates and the wave packet spreads. However, above this regime, nonlinear effects are sufficiently strong to counteract dispersion, leading to the formation of a localized structure. The preservation of analytical invariants in our simulations confirms that the method accurately captures this delicate balance and the associated transition behavior.

Table 3. Comparison of invariants of birth of standing soliton problem at time $t = 6$.

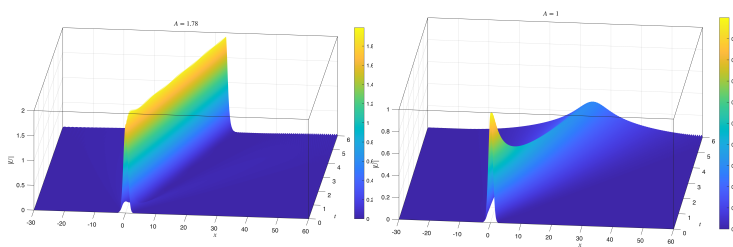
Method	A	h	Δt	C_1	C_2
CHCM-L	1	0.25	0.0001	1.2533141373	0.3671196920
CHCM-C	1	0.25	0.0001	1.2533143988	0.3708031109
G [5]	1	0.25	0.01	1.2533141373	0.3670872106
MQ [5]	1	0.25	0.01	1.2533141398	0.3670872071
$W(7, 5)$ [5]	1	0.25	0.01	1.2533141374	0.3670872107
$W(6, 4)$ [5]	1	0.25	0.01	1.2533141207	0.3670872197
$W(5, 3)$ [5]	1	0.25	0.01	1.2533141284	0.3670868861
$W(4, 2)$ [5]	1	0.25	0.01	1.2533204063	0.3670656847
CHCM-L	1.78	0.25	0.001	3.9710005126	-4.9071789067
CHCM-C	1.78	0.25	0.001	3.9710027553	-4.9513646014
G [5]	1.78	0.25	0.01	3.9710005127	-4.9256176957
MQ [5]	1.78	0.25	0.01	3.9710005126	-4.9256221031
$W(7, 5)$ [5]	1.78	0.25	0.01	3.9710005127	-4.9256176321
$W(6, 4)$ [5]	1.78	0.25	0.01	3.9710005071	-4.9256176397
$W(5, 3)$ [5]	1.78	0.25	0.01	3.9710005116	-4.9256187196
$W(4, 2)$ [5]	1.78	0.25	0.01	3.9710005049	-4.9256646071
B-Spline [27]	1.78	0.25	0.04	3.97100	-4.93298

Example 4. In this numerical experiment, the generation and propagation of a traveling soliton are investigated. The initial profile

$$U(x, 0) = A \exp(-x^2 + 2ix)$$

gives rise to a soliton that exhibits translational motion. Initially, the solitary wave is centered at $x = 0$, and as time evolves, it propagates toward the positive x -direction.

For this experiment, the spatial step size and time increment are chosen as $h = 0.25$ and $\Delta t = 0.0001$, respectively. The simulations are performed within the computational region $[-30, 60]$ for the time interval $[0, 6]$, considering amplitude parameters $A = 1$ and $A = 1.78$. The temporal development of the traveling soliton is illustrated in Figure 5.


Figure 5. Formation of a travelling soliton for $A = 1.78$ and $A = 1$.

The analytical expressions of the conserved quantities yield $C_1 = 1.2533141373$ and $C_2 = 5.3803437645$ for $A = 1$, while for $A = 1.78$, the corresponding values are $C_1 = 3.9710005126$ and $C_2 = 10.9583844401$. The computed numerical invariants together with their relative variations at the final simulation time are

summarized in Table 4. It can be observed that both invariants remain nearly constant throughout the evolution, confirming the stability and accuracy of the proposed numerical scheme.

Table 4. Comparison of invariants of birth of mobile soliton problem at time $t = 6$.

Method	A	h	Δt	C_1	C_2
CHCM-L	1	0.25	0.0001	1.2533141373	5.3814444602
CHCM-C	1	0.25	0.0001	1.2533185889	5.3938277599
G [5]	1	0.25	0.01	1.2533141373	5.3803436429
MQ [5]	1	0.25	0.01	1.2533141961	5.3803433917
$W(7, 5)$ [5]	1	0.25	0.01	1.2533141375	5.3803434416
$W(6, 4)$ [5]	1	0.25	0.01	1.2533141345	5.3803407068
$W(5, 3)$ [5]	1	0.25	0.01	1.2533141391	5.3802912465
$W(4, 2)$ [5]	1	0.25	0.01	1.2533140994	5.3792124676
CHCM-L	1.78	0.25	0.0001	3.9710005126	10.6661827472
CHCM-C	1.78	0.25	0.0001	3.9710629787	10.6176351877
G [5]	1.78	0.25	0.01	3.9710005126	10.9583840548
MQ [5]	1.78	0.25	0.01	3.9710006989	10.9583823585
$W(7, 5)$ [5]	1.78	0.25	0.01	3.9710005132	10.9583834148
$W(6, 4)$ [5]	1.78	0.25	0.01	3.9710005044	10.9583747700
$W(5, 3)$ [5]	1.78	0.25	0.01	3.9709993206	10.9582284948
$W(4, 2)$ [5]	1.78	0.25	0.01	3.9710005433	10.9547877730
B-Spline [27]	1.78	0.25	0.04	3.97100	10.97592

5.1 Discussion on the performance of CHCM-L and CHCM-C

It is observed from the numerical results that CHCM-L outperforms CHCM-C in certain test cases. This behavior can be attributed to the different distribution of collocation nodes. While Chebyshev nodes are strongly clustered near the boundaries, Legendre nodes exhibit a comparatively more uniform distribution over the computational domain. For problems where the solution remains sufficiently smooth and does not exhibit strong boundary-layer behavior, the Legendre-based discretization may provide a more balanced error distribution across the domain. Consequently, improved overall accuracy may be obtained in such cases.

6 Conclusions

In this study, the Cubic Hermite Collocation Method has been effectively utilized to obtain accurate numerical solutions of the nonlinear Schrödinger (NLS) equation. The method not only captures the strong nonlinear and dispersive effects characteristic of the NLS equation but also maintains excellent stability over long-time simulations. The numerical results reveal that the scheme preserves the smoothness and essential features of the soliton structures while efficiently satisfying boundary conditions. Owing to its compact formulation and high-order accuracy, the method requires relatively low computational ef-

fort compared to other techniques commonly used in the literature. These advantages make the Cubic Hermite Collocation Method a powerful and flexible approach for modeling nonlinear wave propagation phenomena. Furthermore, its adaptable framework suggests that it can be extended to a broad class of nonlinear partial differential equations exhibiting similar complex dynamics.

While the developed cubic Hermite spline approach offers notable precision and stability, it is not without specific constraints, particularly regarding its reliance on solution smoothness and increased computational complexity for higher-dimensional extensions. The method may also require careful discretization in problems involving steep gradients. Future research will aim to enhance the efficiency and robustness of the scheme by incorporating adaptive discretization strategies and extending the method to multidimensional nonlinear Schrödinger equations and other nonlinear dispersive models.

References

- [1] S. Arora, R. Jain and V.K. Kukreja. Solution of Benjamin-Bona-Mahony-Burgers equation using collocation method with quintic Hermite splines. *Applied Numerical Mathematics*, **154**:1–16, 2020. <https://doi.org/10.1016/j.apnum.2020.03.015>.
- [2] İ. Dağ. A quadratic B-spline finite element method for solving nonlinear Schrödinger equation. *Computer Methods in Applied Mechanics and Engineering*, **174**:247–258, 1999. [https://doi.org/10.1016/S0045-7825\(98\)00257-6](https://doi.org/10.1016/S0045-7825(98)00257-6).
- [3] L. Debnath. *Nonlinear Partial Differential Equations for Scientist and Engineers*. Birkhauser, Boston, 1997. <https://doi.org/10.1007/978-1-4899-2846-7>.
- [4] M. Delfour, M. Fortin and G. Payr. Finite-difference solutions of a non-linear Schrödinger equation. *Journal of Computational Physics*, **44**(2):277–288, 1981. [https://doi.org/10.1016/0021-9991\(81\)90052-8](https://doi.org/10.1016/0021-9991(81)90052-8).
- [5] Y. Dereli. The meshless kernel-based method of lines for the numerical solution of the nonlinear Schrödinger equation. *Engineering Analysis with Boundary Elements*, **36**(9):1416–1423, 2012. <https://doi.org/10.1016/j.enganabound.2012.02.018>.
- [6] Y. Dereli, D. Irk and İ. Dağ. Soliton solutions for NLS equation using radial basis functions. *Chaos, Solitons & Fractals*, **42**(2):1227–1233, 2009. <https://doi.org/10.1016/j.chaos.2009.03.030>.
- [7] I.A. Ganaie, S. Arora and V.K. Kukreja. Cubic Hermite collocation method for solving boundary value problems with Dirichlet, Neumann, and Robin conditions. *International Journal of Engineering Mathematics*, **2014**:365209, 2014. <https://doi.org/10.1155/2014/365209>.
- [8] I.A. Ganaie, S. Arora and V.K. Kukreja. Cubic Hermite collocation solution of Kuramoto–Sivashinsky equation. *International Journal of Computer Mathematics*, **93**(1):223–235, 2016. <https://doi.org/10.1080/00207160.2014.999052>.
- [9] I.A. Ganaie, B. Gupta, N. Parumasur, P. Singh and V.K. Kukreja. Asymptotic convergence of cubic Hermite collocation method for parabolic partial differential equation. *Applied Mathematics and Computation*, **220**:560–567, 2013. <https://doi.org/10.1016/j.amc.2013.05.073>.

- [10] I.A. Ganaie and V.K. Kukreja. Numerical solution of Burgers' equation by cubic Hermite collocation method. *Applied Mathematics and Computation*, **237**:571–581, 2014. <https://doi.org/10.1016/j.amc.2014.03.102>.
- [11] L.R.T. Gardner, G.A. Gardner, S.I. Zaki and Z. El Sahrawi. B-spline finite element studies of the non-linear Schrödinger equation. *Computer Methods in Applied Mechanics and Engineering*, **108**(3–4):303–318, 1993. [https://doi.org/10.1016/0045-7825\(93\)90007-K](https://doi.org/10.1016/0045-7825(93)90007-K).
- [12] L.R.T. Gardner, G.A. Gardner, S.I. Zaki and Z. El Sahrawi. A leapfrog algorithm and stability studies for the nonlinear Schrödinger equation. *Arabian Journal for Science and Engineering*, **18**:23–32, 1993.
- [13] A.G. Kaplan and Y. Dereli. A meshless method and stability analysis for the non-linear Schrödinger equation. *Waves in Random and Complex Media*, **27**(4):602–614, 2017. <https://doi.org/10.1080/17455030.2017.1290301>.
- [14] S.A. Khuri. A complex tanh-function method applied to nonlinear equations of Schrödinger type. *Chaos, Solitons & Fractals*, **20**(5):1037–1040, 2004. <https://doi.org/10.1016/j.chaos.2003.09.042>.
- [15] A. Korkmaz and İ. Dağ. A differential quadrature algorithm for simulations of nonlinear Schrödinger equation. *Computers & Mathematics with Applications*, **56**(9):2222–2234, 2008. <https://doi.org/10.1016/j.camwa.2008.03.047>.
- [16] A. Korkmaz and İ. Dağ. A differential quadrature algorithm for nonlinear Schrödinger equation. *Nonlinear Dynamics*, **56**:69–83, 2009. <https://doi.org/10.1007/s11071-008-9380-0>.
- [17] A. Kumari and V.K. Kukreja. Septic Hermite collocation method for the numerical solution of Benjamin–Bona–Mahony–Burgers equation. *Journal of Difference Equations and Applications*, **27**(8):1193–1217, 2021. <https://doi.org/10.1080/10236198.2021.1972985>.
- [18] A. Kumari and V.K. Kukreja. Robust septic Hermite collocation technique for singularly perturbed generalized Hodgkin–Huxley equation. *International Journal of Computer Mathematics*, **99**(5):909–923, 2022. <https://doi.org/10.1080/00207160.2021.1939317>.
- [19] S. Kutluay, M. Yağmurlu and A.S. Karakaş. A robust quintic Hermite collocation method for one-dimensional heat conduction equation. *Journal of Mathematical Sciences and Modelling*, **7**(2):82–89, 2024. <https://doi.org/10.33187/jmsm.1475294>.
- [20] S. Kutluay, N.M. Yağmurlu and A.S. Karakaş. A powerful robust cubic Hermite collocation method for the numerical calculations and simulations of the equal width wave equation. *arXiv preprint*, 2023. <https://doi.org/10.48550/arXiv.2309.02439>.
- [21] S. Kutluay, N.M. Yağmurlu and A.S. Karakaş. A novel perspective for simulations of the modified equal-width wave equation by cubic Hermite B-spline collocation method. *Wave Motion*, **129**:103342, 2024. <https://doi.org/10.1016/j.wavemoti.2024.103342>.
- [22] A.K. Mittal, I.A. Ganaie, V.K. Kukreja, N. Parumasur and P. Singh. Solution of diffusion–dispersion models using a computationally efficient technique of orthogonal collocation on finite elements with cubic Hermite as basis. *Computers and Chemical Engineering*, **58**:203–210, 2013. <https://doi.org/10.1016/j.compchemeng.2013.07.007>.

- [23] T. Öziş and A. Yıldırım. Reliable analysis for obtaining exact soliton solutions of nonlinear Schrödinger (NLS) equation. *Chaos, Solitons & Fractals*, **38**(1):209–212, 2008. <https://doi.org/10.1016/j.chaos.2006.11.006>.
- [24] Priyanka, S. Arora, F. Mebrek-Oudina and S. Sahani. Super convergence analysis of fully discrete Hermite splines to simulate wave behaviour of Kuramoto–Sivashinsky equation. *Wave Motion*, **121**:103187, 2023. <https://doi.org/10.1016/j.wavemoti.2023.103187>.
- [25] M.P. Robinson. The solution of nonlinear Schrödinger equations using orthogonal spline collocation. *Computers & Mathematics with Applications*, **33**(7):39–57, 1997. [https://doi.org/10.1016/S0898-1221\(97\)00042-4](https://doi.org/10.1016/S0898-1221(97)00042-4).
- [26] M.P. Robinson and G. Fairweather. Orthogonal spline collocation methods for Schrödinger-type equation in one space variable. *Numerische Mathematik*, **68**:355–376, 1994. <https://doi.org/10.1007/s002110050067>.
- [27] B. Saka. A quintic B-spline finite-element method for solving the nonlinear Schrödinger equation. *Solitons and Chaos, Physics of Wave Phenomena*, **20**:107–117, 2012. <https://doi.org/10.3103/S1541308X12020033>.
- [28] E.H. Twizell, A. Bratros and J.C. Newby. A finite-difference method for solving the cubic Schrödinger equation. *Mathematics and Computers in Simulation*, **43**(1):67–75, 1997. [https://doi.org/10.1016/S0378-4754\(96\)00056-0](https://doi.org/10.1016/S0378-4754(96)00056-0).