



# A new numerical approach to the solution of the nonlinear Kawahara equation by using combined Taylor–Dickson approximation

D. Priyadarsini, M. Routaray and P.K. Sahu\*

## Abstract

This article presents a novel numerical approach to the solution of the nonlinear Kawahara equation. The desired approximations are obtained from the combination of Dickson polynomials and Taylor’s expansion. The combined approach is based on Taylor’s expansion for discretizing the time

\*Corresponding author

Received 28 June 2023; revised 28 August 2023; accepted 10 September 2023

Dibyajyoti Priyadarsini

Department of Mathematics, School of Applied Sciences, KIIT University, Odisha-751024, India. e-mail: pdibyajyoti7@gmail.com

Mitali Routaray

Department of Mathematics, School of Applied Sciences, KIIT University, Odisha-751024, India. e-mail: mitaray8@gmail.com

Prakash Kumar Sahu

Department of Mathematics, School of Applied Sciences, KIIT University, Odisha-751024, India. e-mail: prakash.2901@gmail.com

## How to cite this article

Priyadarsini, D., Routaray, M., and Sahu, P.K., A new numerical approach to the solution of the nonlinear Kawahara equation by using combined Taylor–Dickson approximation. *Iran. J. Numer. Anal. Optim.*, 2024; 14(1): 20-43. <https://doi.org/10.22067/ijnao.2023.83184.1287>

derivative and Dickson polynomials for space derivatives. The problem will be converted into a system of linear algebraic equations for each time step via some suitable collocation points. Error estimation is presented after obtaining the approximate solution. The newly proposed technique is compared with some existing numerical methods to show the method's applicability, accuracy, and efficacy. Two problems are solved to demonstrate the method's power and effect, and the results are presented as a table and graphics.

**AMS subject classifications (2020):** 35C11; 35G20; 65N22.

**Keywords:** Taylor's expansion; Dickson polynomials; Kawahara equation; Discretization.

## 1 Introduction

Partial differential equations (PDEs), including linear and nonlinear PDEs, are used to describe many real-world phenomena [9, 25, 16]. We must solve differential equations to see the nature of the background of these phenomena. In general, we are better at dealing with linear problems than nonlinear problems. However, in practice, we face an increasing number of nonlinear problems. Nonlinear partial differential equations (NPDEs) have a significant role in science and engineering [27, 26]. In the last few decades, many researchers have solved numerous numerical approximations for solving NPDEs. The radial basis function partition of the unity method was used by Nikan and Avazzadeh [22] to solve the sine-Gordon system. Bařhan et al. [5] proposed a numerical solution of the complex modified Korteweg–de Vries equation by the differential quadrature method (DQM). Also, Bařhan et al. [6] proposed a finite difference method combined with DQM for numerical computation of the modified equal-width wave equation. Many numerical methods, such as finite difference, decomposition, homotopy perturbation, finite element, radial basis function, finite volume, and B-spline collocation methods, have been developed in recent years to solve these NPDEs [23, 29, 33, 30, 6]. Nonlinear evolution equations are employed in many engineering and mathematics disciplines, including solid state physics, fluid mechanics, chemical

physics, plasma physics, geochemistry, and chemical kinematics [14, 4]. One of the well-known nonlinear evolution equations is the fifth-order Kawahara equation, which appears in the theories of shallow water waves having surface tension, magneto-acoustic waves in a plasma, and capillary gravity equations [8, 31].

Consider the nonlinear Kawahara equation [16] is of the form

$$u_t + uu_s + u_{3s} + u_{5s} = 0, \quad (s, t) \in [0, 1] \times [0, T], \quad (1)$$

based on the boundary conditions

$$u(0, t) = -\frac{72}{169} + \frac{105}{169} \operatorname{sech}^4 kt, \quad (2)$$

$$u(1, t) = -\frac{72}{169} + \frac{105}{169} \operatorname{sech}^4 k(1 + ct), \quad (3)$$

as well as the initial condition

$$u(s, 0) = -\frac{72}{169} + \frac{105}{169} \operatorname{sech}^4 ks, \quad (4)$$

where  $k = \frac{1}{2\sqrt{13}}$  and  $c = \frac{36}{169}$ .

Researchers have developed a wide range of approximate algorithms and analytical techniques over the last few decades to solve the Kawahara equation. In scholarly literature, there are numerous effective methods for approximating the solutions of nonlinear PDE. The homotopy analysis method was used by Abbasbandy [1] to solve the Kawahara equation. Turgut and Karakoc [2] proposed a numerical approach based on the collocation method to solve the modified Kawahara problem. To address the Kawahara problem numerically, Rasoulizadeh and Rashidinia [28] developed a local radial basis function-finite dimension (RBF-FD) meshless approach. Haq and Uddin [13] looked at the RBFs approximation method for the Kawahara problem. Soltanalizadeh [32] used the differential transformation approach to solve the Kawahara equation numerically. Lahiji and Aziz [21] solved the Kawahara equation numerically using spectral methods. Karakoç, Zeybek, and Ak [15] to compute numerical solutions of the Kawahara equation using the septic B-spline collocation method. Zara et al. [34] applied the kernel smoothing method to approximate the modified Kawahara equation numerically.

Several scholars have also developed different approaches for solving this equation.

In this article, we describe a newly developed, extremely simple numerical technique based on Dickson approximation. There are some studies, such as [20, 19, 18], that employed Dickson polynomials as a basis for numerically solving integral equations, but the present method is different from them. This study seeks to develop a newly combined approximation method that combines Dickson polynomial approximation with Taylor-series approximation to solve the nonlinear Kawahara equation. We discretize the time variables after first estimating the time-dependent term with Taylor's series. The underlying model problem is then assumed to have a solution for the novel Dickson series expansion of the unknown function at each time step. Then, using an appropriate set of collocation points, all associated unknown functions are represented in the Dickson matrix form. Error analysis of the combined technique is also presented here. There are few works on solving polynomial approximation since it is difficult to solve the NPDEs. So we combine Taylor's scheme for time derivative and Dickson polynomial for space derivative to create a new technique to solve the Kawahara equation. The obtained results have been compared with the same obtained by some existing methods. It manifests that the present method is more accurate than other methods and the computational time taken by the present technique is much less (less than a second).

The rest of the paper is organized as follows: Section "Introduction" highlights the importance of the Kawahara equation and its application to a certain field of study. It also mentions the main objectives or goals of the research, as well as any background information required for understanding the next parts. The "Preliminaries" Section emphasizes Taylor's time discretization scheme and the Dickson polynomials. The next part discusses the "Error analysis" to quantify how closely the numerical solution approximates the exact solution. The section "Numerical methods for the Kawahara equation" details the suggested numerical technique for approximating solutions to the Kawahara equation. It also explains the step-by-step procedure of the numerical method, possibly making use of the earlier discussed Dickson polynomials and Taylor's time discretization scheme. The proposed method

is used to solve nonlinear Kawahara equations problems about their accuracy and applicability in Section “Illustrative Examples.” The contributions of the work are summarized in the “Conclusion” section, which highlights the achievements of the proposed method and identifies possible directions for further study.

## 2 Preliminaries

### Taylor’s time discretization scheme:

First, the NPDE is attempted to be discretized with respect to the time variable; that is, the interval  $[0, T]$  can be divided into  $(M + 1)$  grid points as

$$0 =: t_0 < t_1 = \Delta t < \cdots < t_m = M\Delta t = T, \quad (5)$$

with the uniform step size  $\Delta t = t_n - t_{n-1}$ . To obtain a discretization scheme that is time accurate, we get the following from the Taylor series representation for  $u^n = u(s, t_n)$ :

$$u_t^n = \frac{u^{n+1} - u^n}{\Delta t} - \frac{\Delta t u_{tt}^n}{2} + O(\Delta t^2). \quad (6)$$

### Dickson polynomial:

The Dickson polynomial was first introduced by Dickson [11] in 1896; afterward, it was rediscovered by Brewer [10] in 1961. For integer  $n > 0$  and  $\alpha$  is a commutative ring  $\mathbb{R}$  with identity. The first kind Dickson polynomial  $D_n(s, \alpha)$  of degree  $n$  with the parameter  $\alpha \in \mathbb{R}$  is defined by

$$D_n(s, \alpha) = \sum_{p=0}^{\lfloor \frac{n}{2} \rfloor} \frac{n}{n-p} (-\alpha)^p s^{n-2p}, \quad n \geq 1, -\infty < t < \infty.$$

The first few Dickson polynomials are

1.  $D_0(s, \alpha) = 2,$
2.  $D_1(s, \alpha) = s,$
3.  $D_2(s, \alpha) = s^2 - 2\alpha,$
4.  $D_3(s, \alpha) = s^3 - 3s\alpha,$
5.  $D_4(s, \alpha) = s^4 - 4s^2\alpha,$
6.  $D_5(s, \alpha) = s^5 - 5s^3\alpha + 5s\alpha^2.$

They may also be generated by the recurrence relation for  $n \geq 2$  as

$D_n(s, \alpha) = sD_{n-1}(s, \alpha) - \alpha D_{n-2}(s, \alpha), n \geq 2.$  The polynomials  $u = D_n(s, \alpha)$  read the second-order differential equation:

$$(s^2 - 4\alpha)u'' + su' - n^2u = 0, \quad n = 0, 1, 2, 3, \dots$$

### Function approximation:

Since  $f(s) \in L^2(\mathbb{R})$  is a continuous function defined over  $[0, 1]$ , then the Dickson polynomials can be represented as

$$f(s) = \sum_{i=0}^N c_i D_i(s), \quad s \in [0, 1], \quad (7)$$

where  $D_n(s, \alpha) = D_n(s)$  with any arbitrary constant  $\alpha$ .

In this case, it is necessary to figure out the unknown coefficients,  $c_i, i = 0, 1, \dots, N.$  The Dickson vector  $D_N(s)$  and the unknown vector  $C_N$  will be introduced in the following ways:

$$D_N(s) = \left[ D_0(s) \ D_1(s) \ \dots \ D_n(s) \right]^T,$$

and

$$C_N = \left[ c_0 \ c_1 \ \dots \ c_n \right]^T.$$

We can express (7) in a compact representation using these vectors as follows:

$$f(s) = C_N^T D_N(s). \quad (8)$$

Additionally, by introducing the matrix  $\Psi$ ,

$$\Psi = \begin{bmatrix} 2 & 0 & 0 & \dots & 0 \\ 0 & \frac{1}{1} \binom{1}{0} (-\alpha)^0 & 0 & \dots & 0 \\ \frac{2}{1} \binom{2}{1} (-\alpha)^1 & 0 & \frac{2}{2} \binom{2}{0} (-\alpha)^0 & \dots & 0 \\ 0 & \frac{3}{2} \binom{2}{1} (-\alpha)^1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{N}{\frac{N}{2}} \binom{\frac{N}{2}}{\frac{N}{2}} (-\alpha)^{\frac{N}{2}} & 0 & \frac{N}{\frac{N}{2}+1} \binom{\frac{N}{2}+1}{\frac{N}{2}-1} (-\alpha)^{\frac{N}{2}-1} & \dots & 0 \end{bmatrix}_{(N+1) \times (N+1)},$$

and the monomial vector

$$S_n(s) = \left[ 1 \ s \ s^2 \ \dots \ s^n \right]^T,$$

we shall express  $D_N(s)$  as follows:

$$D_N(s) = \Psi S_n(s). \tag{9}$$

For finding the operational matrix of the derivative, a straightforward calculation shows that

$$\begin{aligned} \frac{d}{ds} D_N(s) &= \psi \frac{d}{ds} S_n(s) \\ &= \psi \lambda S_n(s), \end{aligned} \tag{10}$$

where

$$\lambda = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 2 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & 0 \\ 0 & 0 & \dots & 0 & N & 0 \\ 0 & 0 & \vdots & 0 & 0 & 0 \end{bmatrix}_{(N+1) \times (N+1)}. \tag{11}$$

### 3 Error analysis:

Kürkcü, Aslan, and Sezer [19] employed the residual function in the Banach space method to do convergence research on the Dickson polynomial solution for integro-differential equations. We can find the Dickson polynomial-based

error analysis for approximation for the nonlinear Kawahara equation. Finally, using the Dickson and Taylor's approximation, we derive the error analysis of approximation.

**Theorem 1.** [17] Suppose that  $H$  is a Hilbert space and that  $Y$  is a closed subspace of  $H$  such that  $\dim Y < \infty$  and  $y_1, y_2, \dots, y_n$  is any basis for  $Y$ . Let  $x$  be an arbitrary element in  $H$  and let  $y_0$  be the most accurate approximation of  $x \in Y$ . Then  $\|x - y_0\|_2^2 = \frac{G(x, y_1, y_2, \dots, y_n)}{G(y_1, y_2, \dots, y_n)}$ , where

$$G(x, y_1, y_2, \dots, y_n) = \begin{vmatrix} \langle x, x \rangle & \langle x, y_1 \rangle & \langle x, y_2 \rangle & \cdots & \langle x, y_n \rangle \\ \langle y_1, x \rangle & \langle y_1, y_1 \rangle & \langle y_1, y_2 \rangle & \cdots & \langle y_1, y_n \rangle \\ \langle y_2, x \rangle & \langle y_2, y_1 \rangle & \langle y_2, y_2 \rangle & \cdots & \langle y_2, y_n \rangle \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \langle y_n, x \rangle & \langle y_n, y_1 \rangle & \langle y_n, y_2 \rangle & \cdots & \langle y_n, y_n \rangle \end{vmatrix}.$$

**Theorem 2.** Let  $u \in C([0, 1] \times [0, 1])$  and let  $Y = \text{Span}\{D_0, D_1, D_2, \dots, D_n\}$ , where  $D_i$ 's are Dickson polynomials. If  $u_N = c^T \phi(x)$  is the most accurate approximation to  $u$  for particular time step  $\Delta t$ , then the error bound is as follows:

$$\|u - u_N\|_2 \leq O(\Delta t)^2 + \left( \frac{G(u, D_0, D_1, \dots, D_n)}{G(D_0, D_1, \dots, D_n)} \right)^{\frac{1}{2}},$$

*Proof.* Let  $u(x, t)$  be the exact solution and let  $u(x, t_n)$  be an approximate solution after time discretization. For a particular time step  $\Delta t$ , from (6), we have

$$\|u(x, t) - u(x, t_n)\|_2 \leq O(\Delta t)^2.$$

Let  $u_N(x, t_n) = c^T \phi(x)$  be the most accurate approximate solution of  $u(x, t_n)$  using Dickson polynomials for a particular time step  $t = t_n$ . The error bound is as follows:

$$\begin{aligned} \|u(x, t) - u_N(x, t_n)\|_2 &= \|u(x, t) - u(x, t_n) + u(x, t_n) - u_N(x, t_n)\|_2 \\ &\leq \|u(x, t) - u(x, t_n)\|_2 + \|u(x, t_n) - u_N(x, t_n)\|_2 \\ &\leq O(\Delta t)^2 + \left( \frac{G(u, D_0, D_1, \dots, D_n)}{G(D_0, D_1, \dots, D_n)} \right)^{\frac{1}{2}}. \end{aligned}$$

□



#### 4 Numerical method for Kawahara equation

Consider the nonlinear Kawahara equation defined in (1)–(4) and the differentiate equation (1) with respect to  $t$ , we get

$$u_{tt}^n = (u_{5s})_t - (u_{3s})_t - (uu_s)_t. \quad (12)$$

By replacing first-order derivatives  $u_t^n \approx \frac{(u^{n+1}-u^n)}{\Delta t}$  in all occurrences, we may write the equation as

$$u_{tt}^n = \left(\frac{u^{n+1}-u^n}{\Delta t}\right)_{5s} - \left(\frac{u^{n+1}-u^n}{\Delta t}\right)_{3s} - \left(\frac{u^{n+1}-u^n}{\Delta t}\right)u_s - u\left(\frac{u_s^{n+1}-u_s^n}{\Delta t}\right),$$

$$\Delta t u_{tt}^n = (u_{5s}^{n+1} - u_{5s}^n) - (u_{3s}^{n+1} - u_{3s}^n) - (u^{n+1} - u^n)u_s - u(u_s^{n+1} - u_s^n). \quad (13)$$

Next, using the time discretized form, (13) is inserted into the right-hand side of (6) to get

$$u_t^n = \left(\frac{u^{n+1}-u^n}{\Delta t}\right) - \frac{1}{2} \left[ (u_{5s}^{n+1} - u_{5s}^n) - (u_{3s}^{n+1} - u_{3s}^n) - (u^{n+1} - u^n)u_s - u(u_s^{n+1} - u_s^n) \right]. \quad (14)$$

Now, discretizing the time-dependent times of (1), we get

$$u_t^n = u_{5s}^n - u_{3s}^n - u^n u_s^n. \quad (15)$$

Again, replace (15) in the left side of (14); that is,

$$u_{5s}^n - u_{3s}^n - u^n u_s^n = \left(\frac{u^{n+1}-u^n}{\Delta t}\right) - \frac{1}{2} \left[ \Delta t u_{tt}^n = (u_{5s}^{n+1} - u_{5s}^n) - (u_{3s}^{n+1} - u_{3s}^n) - (u^{n+1} - u^n)u_s - u(u_s^{n+1} - u_s^n) \right]. \quad (16)$$

Equation (16) can be simplified as

$$\begin{aligned} \left[\frac{1}{\Delta t} + \frac{1}{2}u_s^n\right]u^{n+1} + \left[\frac{1}{2}u^n\right]u_s^{n+1} + \left[\frac{1}{2}\right]u_{3s}^{n+1} + \left[\frac{-1}{2}\right]u_{5s}^{n+1} \\ = \frac{1}{2}u_{5s}^n - \frac{1}{2}u_{3s}^n + \frac{u^n}{\Delta t}. \end{aligned} \quad (17)$$

The unknown functions  $u^{n+1}$ ,  $u_s^{n+1}$ ,  $u_{3s}^{n+1}$ , and  $u_{5s}^{n+1}$  can be approximated by Dickson polynomials defined in (17) as

$$u^{n+1}(s) = C_N^T D(s) = C_N^T \Psi S_n(s). \quad (18)$$

Similarly,

$$\begin{aligned} u_s^{n+1} &= C_N^T \Psi \lambda S_n(s), \\ u_{2s}^{n+1} &= C_N^T \Psi \lambda^2 S_n(s), \\ u_{3s}^{n+1} &= C_N^T \Psi \lambda^3 S_n(s), \\ u_{5s}^{n+1} &= C_N^T \Psi \lambda^5 S_n(s). \end{aligned}$$

Now, (17) can be described as follows:

$$\begin{aligned} A_3(s)C_N^T \Psi \lambda^5 S_n(s) + A_2(s)C_N^T \Psi \lambda^3 S_n(s) \\ + A_1(s)C_N^T \Psi \lambda S_n(s) + A_0(s)C_N^T \Psi S_n(s) = F_n(s), \end{aligned} \quad (19)$$

where

$$\begin{aligned} A_0(s) &= \frac{1}{\Delta t} + \frac{1}{2}u_s^n, & A_1(s) &= \frac{1}{2}u^n, & A_2(s) &= \frac{1}{2}, \\ A_3(s) &= \frac{-1}{2}, & F_n(s) &= \frac{1}{2}u_{5s}^n - \frac{1}{2}u_{3s}^n + \frac{u^n}{\Delta t}, \end{aligned}$$

and (19) can be simplified as

$$C_N^T \Psi [A_3(s)\lambda^5 + A_2(s)\lambda^3 + A_1(s)\lambda + A_0(s)]S_n(s) = F_n(s). \quad (20)$$

Putting the collocation points  $s_i = (\frac{i}{n})$  in (20), we get a system of the linear equation as follows:

$$C_N^T \Psi [A_3(s_i)\lambda^5 + A_2(s_i)\lambda^3 + A_1(s_i)\lambda + A_0(s_i)]S_n(s_i) = F_n(s_i) \quad (21)$$

for  $i = 0, 1, 2, \dots, N$ . Equation (21) is clearly a set of  $(N+1)$  linear equations in terms of  $(N+1)$  unknown coefficients  $c_0, c_1, c_2, \dots, c_N$  that need to be found.

Again, (21) can be expressed as a matrix equation as

$$D_N C_N = F_n, \quad (22)$$

where

$$D_n := (\Phi A_3(\lambda^T)^5 + \Phi A_2(\lambda^T)^3 + \Phi A_1(\lambda^T) + \Phi A_0)\Psi^T$$

and

$$\Phi = [S_n(s_0) \ S_n(s_1) \ S_n(s_2) \ \dots \ S_n(s_n)],$$

$$A_m = \begin{bmatrix} A_m(s_0) & 0 & \dots & 0 \\ 0 & A_m(s_1) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & A_m(s_N) \end{bmatrix}_{(N+1) \times (N+1)},$$

$$F_n = \begin{bmatrix} f_n(s_0) \\ f_n(s_1) \\ \vdots \\ f_n(s_N) \end{bmatrix}_{(N+1) \times 1}.$$

From the given boundary condition, we have  $u_{n+1}(0) = h_0^{n+1} = -\frac{72}{169} + \frac{105}{169} \operatorname{sech}^4 kct$  and  $u_{n+1}(1) = h_1^{n+1} = -\frac{72}{169} + \frac{105}{169} \operatorname{sech}^4 k(1+ct)$ , using the matrix notation, it may be represented as

$$D_n(0)C_N = h_0^{n+1}, \quad D_n(0) = S_n(s_0)\Psi^T,$$

$$D_n(1)C_N = h_1^{n+1}, \quad D_n(1) = S_n(s_1)\Psi^T.$$

Next, we substitute the first two rows of (22) by the boundary conditions  $[D_n(0); h_0^{n+1}]$  and  $[D_n(1); h_1^{n+1}]$  for the convenience. From the above system, we can obtain the approximate value of the unknown coefficients  $c_0, c_1, c_2, \dots, c_N$  and hence obtain the solution of (18).

## 5 Illustrative examples

To demonstrate the performance of the Combined Taylor–Dickson Approximation (CTDA) technique, numerical simulations based on the nonlinear Kawahara equation for specific initial and boundary values are provided. We utilize an ASUS laptop with Processor (CPU) Ryzen 5 (3500U) and RAM(8GB) processing capabilities for conducting the computation. The MATHEMATICA software is used for the problem implementation and 3D

plotting of exact solution and approximate solution graphs with various values of  $s$ ,  $t$ , and  $\Delta t$ .

**Example 1.** Consider the nonlinear Kawahara equation, (1)–(4). The exact solution of this problem is given by Kaya and Al-Khaled [16] as  $u(s, t) = -\frac{72}{169} + \frac{105}{169} \operatorname{sech}^4 k(s + ct)$ , where  $k = \frac{1}{2\sqrt{13}}$  and  $c = \frac{36}{169}$ . Using time steps  $\Delta t = 0.1$ ,  $\Delta t = 0.01$ ,  $\Delta t = 0.001$ , and  $N = 6$ , the absolute errors obtained by the present method for  $0 \leq s \leq L = 1$  is given in Table 1. Despite the fact that there are few approximation methods for solving the Kawahara problem, the authors have compared their findings to the exact answer that has been verified.

The maximum absolute errors solved by the RBF method [13] with the nonlinear Kawahara equation are compared in Table 2. Table 3 displays the absolute error for Example 1 determined using the CTDA technique for various values of the Dickson parameter  $\alpha = 1, 2$ , and the solution is not affected for different values of  $\alpha$ . Figures 1 and 2 show the behavior of the exact and approximation solutions for  $N = 6$  and  $\Delta t = 0.1$ , respectively. Figure 3 shows a two-dimensional graphical representation of the exact and approximate solutions for  $t = 1$  and  $\Delta t = 0.1$ . The graph represents how these possibilities compare visually. Figure 4 displays an “Absolute error graph” that shows the differences between the exact and approximate solutions for  $N = 6$  and  $\Delta t = 0.1$ . In the literature, the computation time of the RBF method has not been provided. However, the proposed method provides more accurate results than the RBF method, and the computational time of the present method is much less than 0.267 seconds for  $N = 6$ .

From Theorem 2, the maximum absolute error for  $N = 6$  and  $\Delta t = 0.1$  is computed as

$$\begin{aligned} O(\Delta t)^2 + \left( \frac{G(u, D_0, D_1, \dots, D_n)}{G(D_0, D_1, \dots, D_n)} \right)^{\frac{1}{2}} &= (0.1)^2 + 2.01111 \times 10^{-7} \\ &= 1.00002 \times 10^{-2}. \end{aligned}$$

Similarly, for  $\Delta t = 0.01$  and  $\Delta t = 0.001$ , the maximum error can be calculated as  $1.00201 \times 10^{-4}$  and  $1.20111 \times 10^{-6}$ , respectively. So, this present method satisfies the error-bound condition for different time steps.

Table 1: Absolute error for Example 1 for  $N = 6$ .

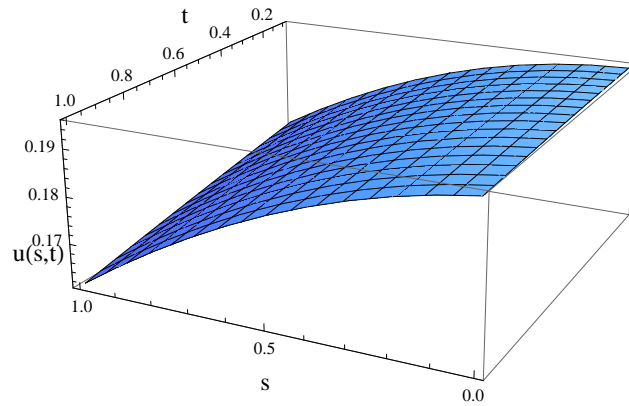
$t$	$s$	$\Delta t = 0.1$	$\Delta t = 0.01$	$\Delta t = 0.001$
0.2	0.2	$2.02289 \times 10^{-6}$	$2.35173 \times 10^{-8}$	$2.66443 \times 10^{-11}$
	0.4	$2.2706 \times 10^{-6}$	$1.01181 \times 10^{-7}$	$2.66443 \times 10^{-10}$
	0.6	$1.33716 \times 10^{-6}$	$2.24034 \times 10^{-8}$	$2.38497 \times 10^{-10}$
	0.8	$9.9872 \times 10^{-6}$	$4.80536 \times 10^{-8}$	$2.66443 \times 10^{-10}$
0.4	0.2	$3.82575 \times 10^{-6}$	$4.60788 \times 10^{-7}$	$3.20035 \times 10^{-10}$
	0.4	$5.57816 \times 10^{-6}$	$4.54266 \times 10^{-7}$	$2.24848 \times 10^{-9}$
	0.6	$1.9494 \times 10^{-6}$	$1.72914 \times 10^{-8}$	$1.91345 \times 10^{-9}$
	0.8	$8.80955 \times 10^{-6}$	$4.29541 \times 10^{-8}$	$1.91345 \times 10^{-9}$
0.6	0.2	$4.51829 \times 10^{-6}$	$5.54659 \times 10^{-7}$	$1.83046 \times 10^{-9}$
	0.4	$7.4988 \times 10^{-6}$	$7.01482 \times 10^{-7}$	$1.6928 \times 10^{-9}$
	0.6	$5.48562 \times 10^{-6}$	$7.81906 \times 10^{-7}$	$1.29511 \times 10^{-9}$
	0.8	$3.23781 \times 10^{-6}$	$2.91407 \times 10^{-8}$	$1.91345 \times 10^{-9}$
0.8	0.2	$3.41209 \times 10^{-6}$	$4.23962 \times 10^{-8}$	$8.60722 \times 10^{-9}$
	0.4	$6.12709 \times 10^{-6}$	$6.07458 \times 10^{-7}$	$1.10209 \times 10^{-10}$
	0.6	$5.48562 \times 10^{-6}$	$1.03009 \times 10^{-7}$	$8.43569 \times 10^{-10}$
	0.8	$1.20464 \times 10^{-6}$	$1.47048 \times 10^{-8}$	$1.91345 \times 10^{-9}$

Table 2: Comparison of  $L_\infty$  errors of Example 1 for  $\Delta t = 0.1$  and  $N = 6, 8, 10$ .

Present method( $N = 6$ )	$9.9872 \times 10^{-4}$
Present method( $N = 8$ )	$2.64883 \times 10^{-6}$
Present method( $N = 10$ )	$2.223943 \times 10^{-7}$
RBF[13] method( $N = 51$ )	$9.38 \times 10^{-6}$

Table 3: Absolute error of Example 1 for Dickson parameter  $\alpha = 1, 2$ .

<b>t</b>	<b>x</b>	$\alpha = 1$		$\alpha = 2$	
		$\Delta t = 0.1$	$\Delta t = 0.01$	$\Delta t = 0.1$	$\Delta t = 0.01$
0.1	0.1	$3.967 \times 10^{-6}$	$7.253 \times 10^{-8}$	$3.967 \times 10^{-6}$	$7.253 \times 10^{-8}$
	0.2	$9.746 \times 10^{-6}$	$1.098 \times 10^{-8}$	$9.746 \times 10^{-6}$	$1.098 \times 10^{-8}$
	0.3	$1.114 \times 10^{-6}$	$9.507 \times 10^{-8}$	$1.114 \times 10^{-6}$	$9.507 \times 10^{-8}$
0.2	0.1	$7.927 \times 10^{-6}$	$1.463 \times 10^{-8}$	$7.927 \times 10^{-6}$	$1.463 \times 10^{-8}$
	0.2	$2.022 \times 10^{-6}$	$2.351 \times 10^{-8}$	$2.022 \times 10^{-6}$	$2.351 \times 10^{-8}$
	0.3	$2.559 \times 10^{-6}$	$2.382 \times 10^{-8}$	$2.559 \times 10^{-6}$	$2.382 \times 10^{-8}$
0.3	0.1	$1.114 \times 10^{-6}$	$2.134 \times 10^{-8}$	$1.114 \times 10^{-6}$	$2.134 \times 10^{-8}$
	0.2	$3.012 \times 10^{-6}$	$3.575 \times 10^{-8}$	$3.012 \times 10^{-6}$	$3.575 \times 10^{-8}$
	0.3	$4.006 \times 10^{-6}$	$3.966 \times 10^{-8}$	$4.006 \times 10^{-6}$	$3.966 \times 10^{-8}$

Figure 1: Exact solution of Example 1 for  $\Delta t = 0.1$  and  $N = 6$ .

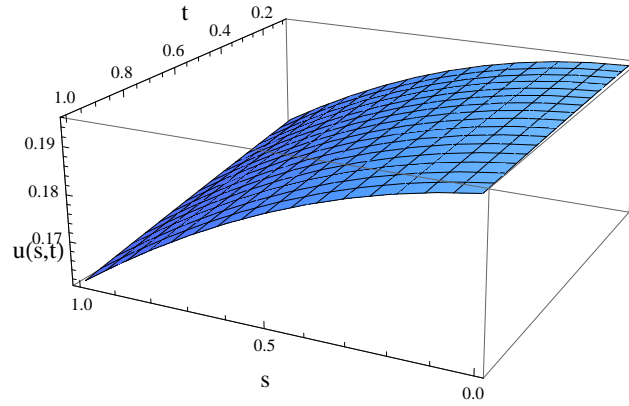


Figure 2: Approximate solution of Example 1 for  $\Delta t = 0.1$  and  $N = 6$ .

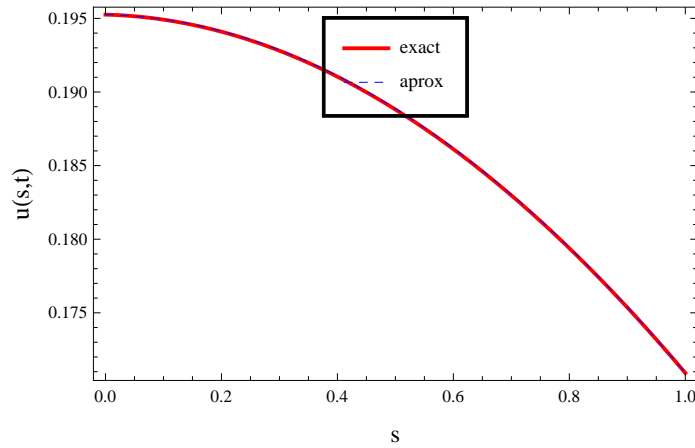


Figure 3: Two-dimensional graph of Example 1 for  $\Delta t = 0.1$  and  $N = 6$  and  $t = 1$ .

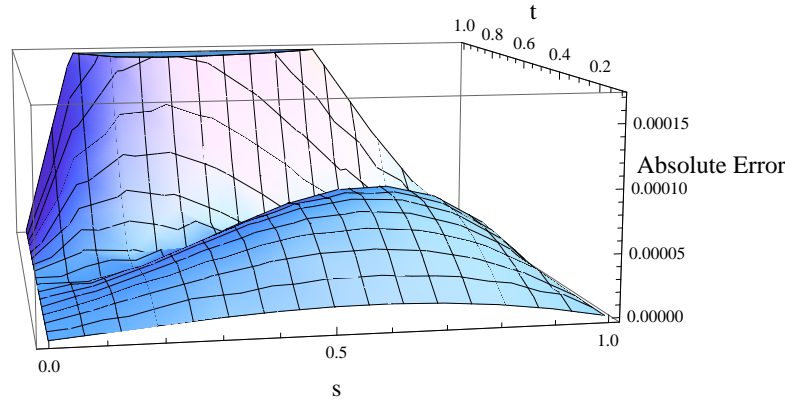


Figure 4: Absolute error of Example 1 for  $\Delta t = 0.1$  and  $N = 6$ .

**Example 2.** Here, we consider a Kawahara-type equation given as

$$u_t + uu_s + u_{3s} + u_s - u_{5s} = 0.$$

Whose exact solution is given as

$u(s, t) = -\frac{105}{169} \operatorname{sech}^4 k(s - 2 - ct)$ , where  $k = \frac{1}{2\sqrt{13}}$  and  $c = \frac{205}{169}$ . Using time steps  $\Delta t = 0.1$ ,  $\Delta t = 0.01$ ,  $\Delta t = 0.001$  and the Kawahara equation with  $N = 8$ , the approximation for  $0 \leq s \leq L = 1$  is given in Table 4. Despite the fact that there are few approximation methods for solving the Kawahara problem, the authors compare their findings to the exact answer that has been verified.

The maximum absolute errors solved by the RBF method [13] with the nonlinear Kawahara equation are compared in Table 5. Table 6 displays the absolute error for Example 2 determined using the CTDA technique for various values of the Dickson parameter  $\alpha = 1, 2$ , and the solution is not affected for different values of  $\alpha$ . Figures 5 and 6 show the behavior of the exact and approximation solutions for  $N = 8$  and  $\Delta t = 0.1$ , respectively. Figure 7 shows a two-dimensional graphical representation of the exact and approximate solutions for  $t = 1$  and  $\Delta t = 0.1$ . The graph represents how these possibilities compare visually. Figure 8 displays an “Absolute error graph” that shows the differences between the exact and approximate solutions for  $N = 6$ ,  $\Delta t = 0.1$ . In the literature, the computation time of the RBF method has not been provided. However, the proposed method provides more accu-



Table 4: Absolute error for Example 2 for  $N = 8$ .

$t$	$s$	$\Delta t = 0.1$	$\Delta t = 0.01$	$\Delta t = 0.001$
0.2	0.2	$1.894 \times 10^{-6}$	$1.73465 \times 10^{-8}$	$3.13242 \times 10^{-11}$
	0.4	$9.10089 \times 10^{-6}$	$1.21102 \times 10^{-7}$	$1.23812 \times 10^{-10}$
	0.6	$1.71853 \times 10^{-6}$	$2.04784 \times 10^{-8}$	$2.06468 \times 10^{-10}$
	0.8	$2.99047 \times 10^{-6}$	$2.01218 \times 10^{-8}$	$2.01002 \times 10^{-10}$
0.4	0.2	$2.64883 \times 10^{-6}$	$4.13838 \times 10^{-7}$	$4.36423 \times 10^{-10}$
	0.4	$1.32887 \times 10^{-6}$	$1.77252 \times 10^{-7}$	$1.81323 \times 10^{-9}$
	0.6	$2.59029 \times 10^{-6}$	$3.0947 \times 10^{-8}$	$3.12158 \times 10^{-9}$
	0.8	$2.89771 \times 10^{-6}$	$3.16247 \times 10^{-8}$	$3.161 \times 10^{-9}$
0.6	0.2	$2.46179 \times 10^{-6}$	$3.81904 \times 10^{-7}$	$4.03194 \times 10^{-9}$
	0.4	$1.28578 \times 10^{-6}$	$1.71872 \times 10^{-7}$	$1.75922 \times 10^{-9}$
	0.6	$2.5832 \times 10^{-6}$	$3.09405 \times 10^{-6}$	$3.12232 \times 10^{-9}$
	0.8	$2.99047 \times 10^{-6}$	$3.27557 \times 10^{-8}$	$3.27585 \times 10^{-9}$
0.8	0.2	$1.5832 \times 10^{-6}$	$2.32987 \times 10^{-6}$	$2.46238 \times 10^{-9}$
	0.4	$8.23146 \times 10^{-6}$	$1.110252 \times 10^{-7}$	$1.112919 \times 10^{-10}$
	0.6	$1.70397 \times 10^{-6}$	$2.04625 \times 10^{-7}$	$2.0659 \times 10^{-10}$
	0.8	$2.03731 \times 10^{-6}$	$2.223943 \times 10^{-7}$	$2.224081 \times 10^{-10}$

rate results than the RBF method and the computational time of the present method is much less that is 0.326 seconds for  $N = 8$ .

## 6 Conclusion

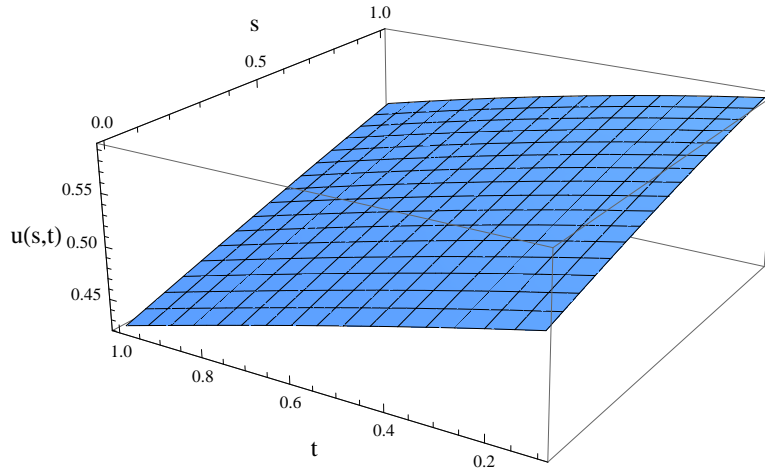
The Kawahara equation was solved using a combination of Taylor's and Dickson polynomial approximation. The authors employed derivative operational matrices as a tool to implement their chosen approximation methods. These matrices help in the numerical computation of function derivatives, which is necessary when dealing with differential equations. For small values of  $N$ , the current method yields very precise results with exact solutions. We get a more accurate solution if we increase the value of  $N$ . Also, taking small values of  $\Delta t$  yields a more accurate result. The validity and convenience of the existing techniques were demonstrated using numerical examples. These ex-

Table 5: Comparison of  $L_\infty$  errors of Example 2 for  $\Delta t = 0.1$  for  $N = 6, 8, 10$ .

Present method( $N = 6$ )	$1.3288 \times 10^{-4}$
Present method( $N = 8$ )	$1.28578 \times 10^{-6}$
Present method( $N = 10$ )	$4.36423 \times 10^{-8}$
RBF[13] method( $N = 51$ )	$6.159 \times 10^{-5}$

Table 6: Absolute error of Example 2 for Dickson parameter  $\alpha = 1, 2$ .

t	x	$\alpha = 1$		$\alpha = 2$	
		$\Delta t = 0.1$	$\Delta t = 0.01$	$\Delta t = 0.1$	$\Delta t = 0.01$
0.1	0.1	$1.118 \times 10^{-6}$	$2.285 \times 10^{-8}$	$1.118 \times 10^{-6}$	$2.285 \times 10^{-8}$
	0.2	$3.220 \times 10^{-6}$	$4.155 \times 10^{-8}$	$3.220 \times 10^{-6}$	$4.155 \times 10^{-8}$
	0.3	$5.039 \times 10^{-6}$	$4.523 \times 10^{-8}$	$5.039 \times 10^{-6}$	$4.523 \times 10^{-8}$
0.2	0.1	$2.138 \times 10^{-6}$	$4.117 \times 10^{-8}$	$2.138 \times 10^{-6}$	$4.117 \times 10^{-8}$
	0.2	$5.860 \times 10^{-6}$	$7.590 \times 10^{-8}$	$5.860 \times 10^{-6}$	$7.590 \times 10^{-8}$
	0.3	$8.355 \times 10^{-6}$	$9.451 \times 10^{-8}$	$8.355 \times 10^{-6}$	$9.451 \times 10^{-8}$
0.3	0.1	$2.835 \times 10^{-6}$	$5.463 \times 10^{-8}$	$2.835 \times 10^{-6}$	$5.463 \times 10^{-8}$
	0.2	$7.847 \times 10^{-6}$	$1.019 \times 10^{-7}$	$7.847 \times 10^{-6}$	$1.019 \times 10^{-7}$
	0.3	$1.133 \times 10^{-5}$	$1.296 \times 10^{-7}$	$1.133 \times 10^{-5}$	$1.296 \times 10^{-7}$

Figure 5: Exact solution of Example 2 for  $\Delta t = 0.1$  and  $N = 8$ .

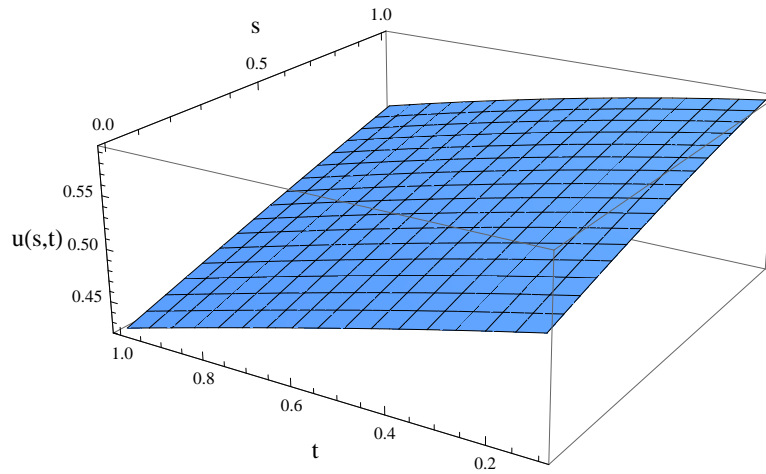


Figure 6: Approximate solution of Example 2 for  $\Delta t = 0.1$  and  $N = 8$ .

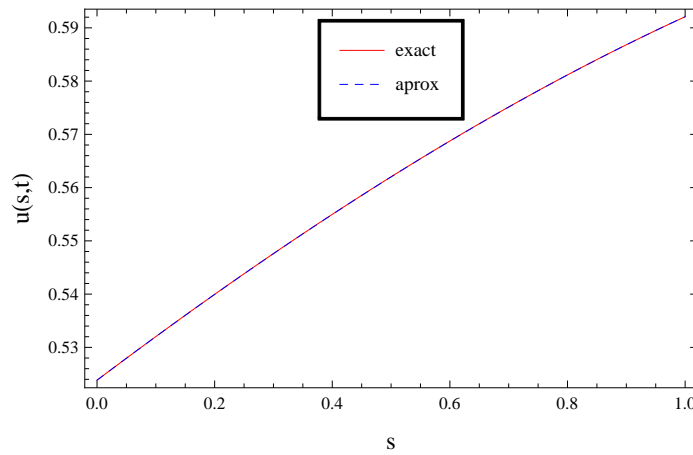


Figure 7: Two-dimensional graph of Example 2 for  $\Delta t = 0.1$ ,  $N = 8$ , and  $t = 1$ .

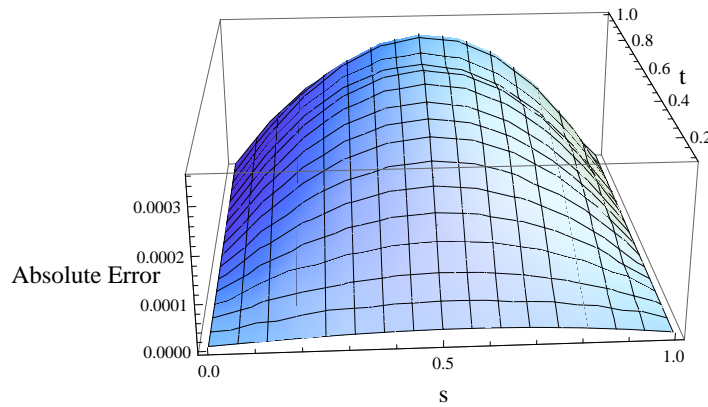


Figure 8: Absolute error of Example 2 for  $\Delta t = 0.1$  and  $N = 6$ .

amples also proved the accuracy and efficiency of the proposed method. This method established a balance between accuracy and computational efficiency, which is important when dealing with challenging mathematical problems. This method can also be used to solve various NPDEs derived from physical models, which are subject to specified initial and boundary conditions. The method's future direction is that it can be used for NPDE with higher-order time derivatives.

## Acknowledgements

Authors are grateful to the anonymous referees and editor for their constructive comments to strengthen this manuscript.

## References

- [1] Abbasbandy, S. *Homotopy analysis method for the Kawahara equation*, Nonlinear Anal. Real World Appl. 11(1) (2010), 307–312.
- [2] Ak, T. and Karakoc, S.B.G. *A numerical technique based on collocation*

- method for solving modified Kawahara equation*, J. Ocean Eng. Sci. 3(1) (2018), 67–75.
- [3] Alipour, M. and Soradi Zeid, S. *Optimal control of Volterra integro-differential equations based on interpolation polynomials and collocation method*, Comput. Methods Differ. Equ. 11(1) (2023), 52–64.
- [4] Aljahdaly, N.H. and Alweldi, A.M. *On the modified Laplace homotopy perturbation method for solving damped modified kawahara equation and its application in a fluid*, Symmetry, 15(2) (2023), 394.
- [5] Başhan, A., Uçar, Y., Yağmurlu, N.M. and Esen, A., 2016, October. *Numerical solution of the complex modified Korteweg-de Vries equation by DQM*, In Journal of Physics: Conference Series (Vol. 766, No. 1, p. 012028). IOP Publishing.
- [6] Başhan, A., Yağmurlu, N.M., Ucar, Y. and Esen, A., 2021. *Finite difference method combined with differential quadrature method for numerical computation of the modified equal width wave equation*, Numer. Methods Partial Differential Equations 37(1) (2021), 690–706.
- [7] Bhargava, M. and Zieve, M.E. *Factoring Dickson polynomials over finite fields*, Finite Fields and Their Appl. 5(2) (1999), 103–111.
- [8] Bhattar, S., Mathur, A., Kumar, D., Nisar, K.S. and Singh, J. *Fractional modified Kawahara equation with Mittag-Leffler law*, Chaos, Solitons Fractals, 131 (2020), 109508.
- [9] Biswas, A. *Solitary wave solution for the generalized Kawahara equation*, Appl. Math. Lett. 22(2) (2009), 208–210.
- [10] Brewer, B.W., *On certain character sums*, Trans. Amer. Math. Soc. 99(2) (1961), 241–245.
- [11] Dickson, L.E., *The analytic representation of substitutions on a power of a prime number of letters with a discussion of the linear group*, Ann. of Math. 11(1-6) (1896/97), 65–120.

- [12] Fernando, N., *A study of permutation polynomials over finite fields*, Thesis (Ph.D.)—University of South Florida. ProQuest LLC, Ann Arbor, MI, 2013. 109 pp.
- [13] Haq, S. and Uddin, M. *RBFs approximation method for Kawahara equation*, Eng. Anal. Bound. Elem. 35(3) (2011), 575–580.
- [14] He, D., *New solitary solutions and a conservative numerical method for the Rosenau–Kawahara equation with power law nonlinearity*, Nonlinear Dynam. 82 (2015), 1177–1190.
- [15] Karakoç, S.B.G., Zeybek, H. and Ak, T. *Numerical solutions of the Kawahara equation by the septic B spline collocation method*, Stat. Optim. Inf. Comput. 2(3) (2014), 211–221.
- [16] Kaya, D. and Al-Khaled, K. *A numerical comparison of a Kawahara equation*, Phys. Lett. A, 363 (5-6) (2007), 433–439.
- [17] Kreyszig, E. *Introductory functional analysis with applications*, John Wiley & Sons, New York-London-Sydney, 1978.
- [18] Kürkçü, Ö.K., Aslan, E. and Sezer, M. *A numerical approach with error estimation to solve general integro-differential-difference equations using Dickson polynomials*, Appl. Math. Comput 276 (2016), 324–339.
- [19] Kürkçü, Ö.K., Aslan, E. and Sezer, M. *A novel collocation method based on residual error analysis for solving integro-differential equations using hybrid Dickson and Taylor polynomials*, Sains Malays, 46 (2017), 335–347.
- [20] Kürkçü, Ö.K., Aslan, E., Sezer, M. and İlhan, Ö. *A numerical approach technique for solving generalized delay integro-differential equations with functional bounds by means of Dickson polynomials*, Int. J. Comput. Methods, 15(05) (2018), 1850039.
- [21] Lahiji, M.A. and Aziz, Z.A. *Numerical solution for Kawahara equation by using spectral methods*, IERI Procedia, 10 (2014), 259–265.

- [22] Nikan, O. and Avazzadeh, Z. *A locally stabilized radial basis function partition of unity technique for the sine-Gordon system in nonlinear optics*, Math. Comput. Simul. 199 (2022), 394–413.
- [23] Nikan, O., Avazzadeh, Z., Machado, J.T. and Rasoulizadeh, M.N., *An accurate localized meshfree collocation technique for the telegraph equation in propagation of electrical signals*, Eng. Comput. 39(3) (2023), 2327–2344.
- [24] Nikan, O., Avazzadeh, Z. and Rasoulizadeh, M.N., *Soliton solutions of the nonlinear sine-Gordon model with Neumann boundary conditions arising in crystal dislocation theory*, Nonlinear Dynam. 106(1) (2021), 783–813.
- [25] Nikan, O., Avazzadeh, Z. and Rasoulizadeh, M.N., *Soliton wave solutions of nonlinear mathematical models in elastic rods and bistable surfaces*, Eng. Anal. Bound. Elem. 143 (2022), 14–27.
- [26] Özer, S. *Numerical solution by quintic B-spline collocation finite element method of generalized Rosenau–Kawahara equation*, Math. Sci. 16(3) (2022), 213–224.
- [27] Polat, N., Kaya, D. and Tutalar, H.I. *A analytic and numerical solution to a modified Kawahara equation and a convergence analysis of the method*, Appl. Math. Comput. 179(2) (2006), 466–472.
- [28] Rasoulizadeh, M.N. and Rashidinia, J. *Numerical solution for the Kawahara equation using local RBF-FD meshless method*, J. King Saud Univ. Sci. 32(4) (2020), 2277–2283.
- [29] Sahu, P.K., Ray, S.S. *Numerical solutions for Volterra integro-differential forms of Lane-Emden equations of first and second kind using Legendre multiwavelets*, Electron. J. Differ. Equ. 2015 (28) (2015), 1–11.
- [30] Sahu, P.K. and Ray, S.S., *Legendre wavelets operational method for the numerical solutions of nonlinear Volterra integro-differential equations system*, Appl. Math. Comput. 256 (2015), 715–723.

- [31] Saldır, O., Sakar, M.G. and Erdogan, F., *Numerical solution of time-fractional Kawahara equation using reproducing kernel method with error estimate*, *Comput. Appl. Math.* 38 (2019), 1–23.
- [32] Soltanalizadeh, B., *Application of differential transformation method for numerical analysis of Kawahara equation*, *Aust. J. Basic Appl. Sci.* 5(12) (2011), 490–495.
- [33] Wang, X. and Cheng, H. *Solitary wave solution and a linear mass-conservative difference scheme for the generalized Korteweg–de Vries–Kawahara equation*, *Comput. Appl. Math.* 40 (2021), 1–26.
- [34] Zara, A., Rehman, S.U., Ahmad, F., Kouser, S. and Pervaiz, A. *Numerical approximation of modified Kawahara equation using Kernel smoothing method*, *Math. Comput. Simul.* 194 (2022), 169–184.