



# Single Solitary Wave and Wave Generation Solutions of the Regularised Long Wave (RLW) Equation

Ali BASHAN\* *Zonguldak Bulent Ecevit University, Department of Maths, 67100, Zonguldak, Turkey*

## Highlights

- This paper focuses on single solitary wave and wave generation solutions of the RLW equation.
- An effective algorithm is proposed for numerical solution of the RLW equation.
- High accurate solutions of the RLW equation were obtained.

## Article Info

*Received: 06 Mar 2021**Accepted: 20 Oct 2021*

## Keywords

*Differential quadrature method,  
Finite difference method,  
RLW equation  
Convergence*

## Abstract

In this paper, high accurate numerical solutions of the regularised long-wave (RLW) equation is going to be obtained by using effective algorithm including finite difference method, differential quadrature and Rubin-Graves type linearization technique. Solitary wave solutions and Maxwellian initial condition based wave generation solutions are obtained successfully. To observe the development of the present algorithm, the present numerical results are compared with many earlier works. The present results are seen as superior among the given ones. The rates of the convergence are also given.

## 1. INTRODUCTION

Benjamin et al. [1] developed the mathematical theory of the RLW equation (or also BBM equation) as

$$U_t + U_x + \varepsilon U U_x - \mu U_{xxt} = 0. \quad (1)$$

The RLW equation is an important model for long waves. The RLW equation can not completely integrable. So, RLW equation doesn't have N-soliton solutions but it has stable solitary wave solution [2]. The RLW equation was used to describe surface water waves, acoustic-gravity waves, acoustic waves in anharmonic crystals and hydromagnetic waves in cold plasma [2]. The RLW equation has been solved analytically [3–8]. Numerical solutions of the RLW equation are obtained by many scientist using various methods. For instance, finite difference method, collocation method, Galerkin method, least squares method, Fourier pseudospectral method, variational iteration method and Adomian decomposition method have been used [9–35].

There are many numerical methods in the literature [36–38]. The main motivation of the present approach is to gather two effective numerical methods and obtain the powerful algorithm. One of the classical numerical method is finite difference method (FDM) and the other numerical method is differential quadrature method (DQM). Bellman et al. [39] suggested DQM for approximating of the differential equations. In the past decades, many different base functions have been used for numerical solutions of the differential equations [40–47].

## 2. METHODOLOGY

DQM can be described as derivative value of any smooth function at grid points by linear combination of all functional values of grid points inside of the solution interval. The simple presentation of the DQM may be given as follows

$$\frac{d^{(r)}f(x_i)}{dx^{(r)}} = \sum_{j=1}^N m_{ij}^{(r)} f(x_j).$$

By using quartic B-splines [48] for obtaining the first order weighting coefficients

$$\frac{dQ_k(x_i)}{dx} = \sum_{j=k-1}^{k+2} m_{ij}^{(1)} Q_k(x_j), \quad k = -1, 0, \dots, N+1, \quad i = 1, 2, \dots, N \quad (2)$$

is obtained. After the quartic B-splines are used in Equation (2), an equation system including 3 more unknowns than equations is obtained. In order to create a solvable equation system, the derivative of Equation (2) for the three B-splines  $\{Q_{-1}, Q_N, Q_{N+1}\}$  is obtained and used to get solvable equation system. Weighting coefficients of the first grid point  $x_1$  are determined by solution of equation system

$$[K][Y_1^1] = [C_1^1]$$

where

$$K = \begin{bmatrix} 8 & 14 & 2 & & & & & & \\ 1 & 11 & 11 & 1 & & & & & \\ \ddots & \ddots & \ddots & \ddots & & & & & \\ & & & & 1 & 11 & 11 & 1 & \\ & & & & & 2 & 14 & 8 & \\ & & & & & & 30 & 42 & \end{bmatrix}_{(N+3) \times (N+3)},$$

$$[Y_1^1] = \left[ m_{1,-1}^{(1)}, m_{1,0}^{(1)}, m_{1,1}^{(1)}, m_{1,2}^{(1)}, \dots, m_{1,N}^{(1)}, m_{1,N+1}^{(1)} \right]_{(N+3) \times 1}^T,$$

and

$$[C_1^1] = \left[ -\frac{7}{h}, -\frac{12}{h}, \frac{12}{h}, \frac{4}{h}, 0 \dots 0 \ 0 \right]_{(N+3) \times 1}^T.$$

Weighting coefficients for  $m_{pj}^{(1)}$ ,  $j = -1, 0, \dots, N+1$  at mesh points  $x_p$ ,  $2 \leq p \leq N-1$  are obtained by solution of the equation system:

$$[K][Y_p^1] = [C_p^1]$$

where

$$[Y_p^1] = \left[ m_{p,-1}^{(1)}, \dots, m_{p,p-3}^{(1)}, m_{p,p-2}^{(1)}, m_{p,p-1}^{(1)}, m_{p,p}^{(1)}, m_{p,p+1}^{(1)}, m_{p,p+2}^{(1)}, \dots, m_{p,N+1}^{(1)} \right]_{(N+3) \times 1}^T,$$

and

$$[C_p^1] = \left[ 0, \dots, 0, -\frac{4}{h}, -\frac{12}{h}, \frac{12}{h}, \frac{4}{h}, 0 \dots 0 \right]_{(N+3) \times 1}^T.$$

For the last mesh point  $x_N$  weighting coefficients  $m_{Nj}^{(1)}$ ,  $j = -1, 0, \dots, N + 1$  are obtained by the solution of the equation system:

$$[K][Y_N^1] = [C_N^1]$$

where

$$[Y_N^1] = \left[ m_{N,-1}^{(1)}, m_{N,0}^{(1)}, \dots, m_{N,N-3}^{(1)}, m_{N,N-2}^{(1)}, m_{N,N-1}^{(1)}, m_{N,N}^{(1)}, m_{N,N+1}^{(1)} \right]_{(N+3) \times 1}^T,$$

and

$$[C_N^1] = \left[ 0, 0, \dots, 0, -\frac{4}{h}, -\frac{12}{h}, \frac{9}{h}, \frac{53}{h} \right]_{(N+3) \times 1}^T.$$

The same methodology is used to obtain the second order weighting coefficients.

### 3. APPLICATION OF METHOD

One of the significant type of the FDM namely the Crank-Nicolson approximation [49] is applied to the Equation (1)

$$\frac{U^{n+1} - U^n}{\Delta t} + \frac{U_x^{n+1} + U_x^n}{2} + \varepsilon \frac{(UU_x)^{n+1} + (UU_x)^n}{2} - \mu \frac{U_{xx}^{n+1} - U_{xx}^n}{\Delta t} = 0, \quad (3)$$

obtained and Equation (3) is arranged as follow,

$$2U^{n+1} + \Delta t[U_x^{n+1} + \varepsilon(UU_x)^{n+1}] - 2\mu U_{xx}^{n+1} = 2U^n - \Delta t[U_x^n + \varepsilon(UU_x)^n] - 2\mu U_{xx}^n. \quad (4)$$

Rubin and Graves type linearization process [50] is applied to Equation (4) to deal with the nonlinear terms.

$$(VV_x)^{n+1} = V^{n+1}V_x^n + V^nV_x^{n+1} - V^nV_x^n,$$

$$(VV_x)^n = V^nV_x^n.$$

Accordingly, nonlinear terms are dealt with and

$$2U^{n+1} + \Delta t[U_x^{n+1} + \varepsilon(U^nU_x^{n+1} + U^{n+1}U_x^n)] - 2\mu U_{xx}^{n+1} = 2U^n - \Delta tU_x^n - 2\mu U_{xx}^n, \quad (5)$$

obtained. Some notations are used to simplify Equation (5) as follows

$$A_i^n = \sum_{j=1}^N m_{ij}^{(1)} U_j^n = U_{x_i}^n, \quad B_i^n = \sum_{j=1}^N m_{ij}^{(2)} U_j^n = U_{xx_i}^n, \quad (6)$$

where  $A_i^n$  is the first-order derivative approach of  $U$  function related to the  $n$ -th time level on grid points  $x_i$  and  $B_i^n$  is the second-order derivative approach of  $U$  function at the  $n$ -th time level on grid points  $x_i$ . We have used those notations (6) in Equation (5) and found out that

$$2U_i^{n+1} + \Delta t \left[ \sum_{j=1}^N m_{ij}^{(1)} U_j^{n+1} + \varepsilon \left( U_i^n \sum_{j=1}^N m_{ij}^{(1)} U_j^{n+1} + A_i^n U_i^{n+1} \right) \right] - 2\mu \sum_{j=1}^N m_{ij}^{(2)} U_j^{n+1} = S_i^n, \quad (7)$$

where

$$S_i^n = 2U_i^n - \Delta t A_i^n - 2\mu B_i^n \quad \text{for } i = 1, 2, \dots, N.$$

Then Equation (7) is classified as below

$$\begin{aligned} & \left[ 2 + \Delta t \left( m_{ii}^{(1)} + \varepsilon \left( U_i^n m_{ii}^{(1)} + A_i^n \right) \right) - 2m_{ii}^{(2)} \right] U_i^{n+1} \\ & + \left[ \sum_{j=1, i \neq j}^N \left[ \Delta t \left( m_{ij}^{(1)} + \varepsilon U_i^n m_{ij}^{(1)} \right) - 2\mu m_{ij}^{(2)} \right] U_j^{n+1} \right] = S_i^n. \end{aligned} \quad (8)$$

The matrix form of the Equation System (8) has been given as below

$$\begin{bmatrix} K_{1,1} & K_{1,2} & \dots & K_{1,N} \\ K_{2,1} & K_{2,2} & \dots & K_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ K_{N-1,1} & K_{N-1,2} & \dots & K_{N-1,N} \\ K_{N,1} & K_{N,2} & \dots & K_{N,N} \end{bmatrix} \begin{bmatrix} U_1^{n+1} \\ U_2^{n+1} \\ \vdots \\ U_{N-1}^{n+1} \\ U_N^{n+1} \end{bmatrix} = \begin{bmatrix} S_1^n \\ S_2^n \\ \vdots \\ S_{N-1}^n \\ S_N^n \end{bmatrix}. \quad (9)$$

Lastly, the system of Equations (9) is arranged by using the boundary conditions. To get the square solvable matrix, the first and last equations are eliminated. So,

$$\begin{bmatrix} K_{2,2} & K_{2,3} & \dots & K_{2,N-1} \\ K_{3,2} & K_{3,3} & \dots & K_{3,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ K_{N-1,2} & K_{N-1,3} & \dots & K_{N-1,N-1} \end{bmatrix} \begin{bmatrix} U_2^{n+1} \\ U_3^{n+1} \\ \vdots \\ U_{N-1}^{n+1} \end{bmatrix} = \begin{bmatrix} S_2^n - K_{2,1} U_1^{n+1} - K_{2,N} U_N^{n+1} \\ S_3^n - K_{3,1} U_1^{n+1} - K_{3,N} U_N^{n+1} \\ \vdots \\ S_{N-1}^n - K_{N-1,1} U_1^{n+1} - K_{N-1,N} U_N^{n+1} \end{bmatrix},$$

solvable matrix system is obtained.

### 3.1. Local Truncation Error (LTE)

A new notation is used to display Equation (1) as follows [51]

$$\frac{\partial}{\partial t} \left( \frac{\partial^2 u(x, t)}{\partial x^2} - \frac{1}{\mu} u(x, t) \right) = \frac{1}{\mu} \left( \frac{\partial u(x, t)}{\partial x} + \varepsilon u(x, t) \frac{\partial u(x, t)}{\partial x} \right)$$

and Crank-Nicolson scheme is used to derive

$$\begin{aligned} & \left( \frac{\partial^2 u(x_i, t_{j+1})}{\partial x^2} - \frac{1}{\mu} u(x_i, t_{j+1}) \right) - \left( \frac{\partial^2 u(x_i, t_j)}{\partial x^2} - \frac{1}{\mu} u(x_i, t_j) \right) = \frac{k}{4\mu h} (1 + \varepsilon u(x_i, t_{j+1})) \\ & \times (u(x_{i+1}, t_{j+1}) - u(x_{i-1}, t_{j+1})) + \frac{k}{4\mu h} (1 + \varepsilon u(x_i, t_j)) (u(x_{i+1}, t_j) - u(x_{i-1}, t_j)) + O(k^3 + kh^2). \end{aligned}$$

#### 4. RESULTS AND DISCUSSIONS

The error norms  $L_2$  and  $L_\infty$  are calculated to observe the accuracy of the present method, respectively:

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

Another measurement of the accuracy of the numerical methods are invariants. The three invariants formulae are given by Olver [52] and calculated for RLW equation in the present paper

$$I_1 = \int_a^b U dx, \quad I_2 = \int_a^b [U^2 + \mu(U_x)^2] dx, \quad I_3 = \int_a^b [U^3 + 3U^2] dx.$$

To control the effectiveness of the suggested method numerical values are compared with analytic values that given by Zaki [29]

$$I_1 = \frac{6C}{k}, \quad I_2 = \frac{12C^2}{k} + \frac{48k\mu C^2}{5}, \quad I_3 = \frac{36C^2(4C+5)}{5k}.$$

To be able to calculate the rate of convergence, following formulae is used

$$ROC = \frac{\ln(\text{error}(N_2)/\text{error}(N_1))}{\ln((N_1)/(N_2))}.$$

##### 4.1. Different Applications of the Single Solitary Wave

The single solitary wave for the RLW equation has analytical solution as below

$$U(x, t) = 3C \operatorname{sech}^2(kx - kvt - kx_0), \quad (10)$$

where

$$v = 1 + \varepsilon C,$$

$$k = \sqrt{\frac{\varepsilon C}{4\mu v}}$$

The velocity of the single solitary wave is represented by  $v$ , amplitude of the wave is represented by  $3C$  and the width of the wave is represented by  $k$ . The position of the wave is initially is  $x_0$  and by the simulation is run, wave moves to the right direction. The initial condition is obtained from analytical solution (10) by using initial time  $t = 0$

$$U(x, 0) = 3C \operatorname{sech}^2(kx - kx_0).$$

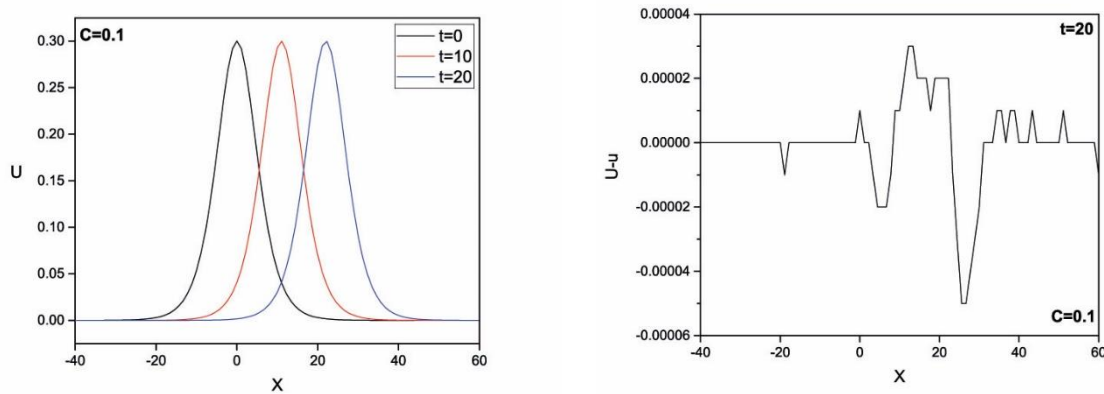
The boundary conditions of Equation (1) are the  $U(\pm\infty, t) = 0$ . The present method is checked and compared with earlier works and analytical results by solving the RLW equation for various amplitude values and solution domain values.

**1.A.** Firstly, the constant parameters  $\varepsilon = \mu = 1$ ,  $C = 0.1$  and  $x_0 = 0$  are chosen over the solution domain  $-40 \leq x \leq 60$  to be able to compare with earlier works. A simulation of the single solitary wave has been run until  $t = 20$ . The solitary wave preserved its amplitude, velocity and shape during movement towards right. The peak location and the amplitude of the wave is given in Table 1. The numerical illustration of the single solitary wave is given in Figure 1 with  $\Delta t = 0.1$  and  $N = 91$ . As it seen from Table 1 and Figure 1 that the shape of the wave is preserved and the amplitude of the wave changed for miserable values. Error values for the end of the simulation  $t = 20$  are given in Figure 1. The error norms and the three invariants

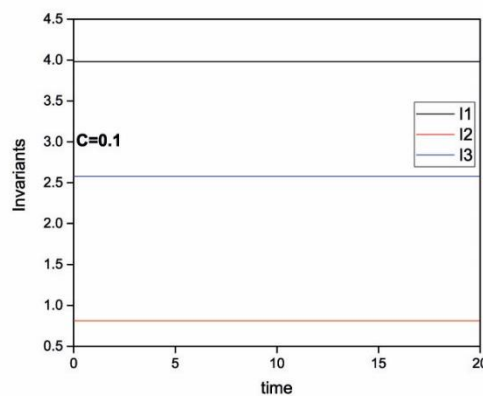
are calculated and they are compared with those of 34 various applications [9, 11–27, 29–31] in Table 2. By using the same time increment  $\Delta t = 0.1$ , the present algorithm has the smallest error norms  $L_2$  and  $L_\infty$  as  $1.3030 \times 10^{-4}$  and  $4.730 \times 10^{-5}$ , respectively at time  $t = 20$ . The three invariants are calculated analytically [29] as  $I_1 = 3.9799497484265$ ,  $I_2 = 0.810462494225$ ,  $I_3 = 2.5790074369804$ , respectively. The present numerical invariants are in good agreement with those of analytic values. The development of the three invariants during the simulation is plotted in Figure 2. Table 3 shows the calculated values of the rate of convergence and the error norms.

**Table 1.** 1.A. The peak locations and the amplitude values for  $\Delta t = 0.1$  on  $-40 \leq x \leq 60$

t	C = 0.1		C = 0.03	
	Peak Position(x)	Amplitude (U)	Peak Position(x)	Amplitude (U)
0	0.00000	0.30000	0.00000	0.09000
5	5.55556	0.29999	5.00000	0.08998
10	11.11111	0.29993	10.00000	0.08994
15	16.66667	0.29983	15.62500	0.08998
20	22.22222	0.29968	20.62500	0.09000



**Figure 1.** 1.A. Single solitary wave movement and maximum error for  $C = 0.1$  at time  $t = 20$



**Figure 2.** 1.A. Behaviour of the three invariants  $C = 0.1$

**Table 2.** 1.A. The error norms and the three invariants:  $C = 0.1, \Delta t = 0.1$  on  $-40 \leq x \leq 60$

Method	N	t	$L_2 \times 10^3$	$L_\infty \times 10^3$	$I_1$	$I_2$	$I_3$
Present	91	0	-	-	3.97993	0.81046	2.57901
		5	0.04569	0.01839	3.97993	0.81046	2.57901
		10	0.07981	0.02967	3.97992	0.81046	2.57901
		15	0.10623	0.03852	3.97992	0.81046	2.57901
		20	0.13030	0.04730	3.97987	0.81046	2.57901
FDM[9]	1001	20	0.55	0.21	3.97997	0.81045	2.57901
Coll.[12]	1001	20	-	0.308	3.97988	0.81046	2.57900
Coll.[13]	801	20	0.301	0.114	3.97996	0.81027	2.57839
Coll. [14]MQ	801	20	0.20691	0.07802	3.97988	0.81046	2.57900
Coll. [14]IMQ	801	20	0.20691	0.07802	3.97987	0.81046	2.57900
Coll. [14]IQ	801	20	0.20691	0.07802	3.97988	0.81046	2.57900
Coll. [14]GA	801	20	0.20691	0.07802	3.97988	0.81046	2.57900
Coll. [14]TPS	801	20	0.20714	0.07815	3.97988	0.81046	2.57900
Coll. [15]QBCM1	801	20	0.21519	0.08295	3.97987	0.81046	2.57900
Coll. [15]QBCM2	801	20	0.35667	0.12968	3.97988	0.81046	2.57900
Coll. [15]QBCM3	801	20	0.21519	0.08295	3.97987	0.81046	2.57900
Coll. [16]	129	20	0.17974	0.06799	3.97991	0.81046	2.57900
Coll. [17]	1001	20	0.53223	0.22722	3.97803	0.80972	2.57657
Coll. [18]	801	20	0.30	0.116	3.97988	0.81027	2.57839
Coll. [19]QBCM1	801	20	0.215	0.083	3.97995	0.81046	2.57901
Coll. [19]QBCM2	801	20	0.357	0.129	3.97995	0.81046	2.57901
Coll. [19]QBCM1	1001	20	0.215	0.083	3.97996	0.81046	2.57901
Coll. [19]QBCM2	1001	20	0.357	0.129	3.97996	0.81046	2.57901
Galerkin [20]	801	20	0.219	0.086	3.97988	0.81046	2.57901
Galerkin [21]	501	20	1.7569	0.68432	3.9800	0.8104	2.5792
Galerkin [22]	801	20	0.26685	0.09146	3.97972	0.81026	2.57873
Galerkin [23]	1001	20	0.217	0.084	3.97989	0.81046	2.57901
Galerkin [24]QBGM1	801	20	0.192	0.073	3.97988	0.81046	2.57900
Galerkin [24]QBGM2	801	20	0.355	0.128	3.97988	0.81046	2.57900
Galerkin [24]QBGM1	1001	20	0.192	0.073	3.97988	0.81046	2.57900
Galerkin [24]QBGM2	1001	20	0.355	0.129	3.97988	0.81046	2.57900
Galerkin [25]	801	20	0.192	0.073	3.97989	0.81046	2.57901
Galerkin [25]	1001	20	0.192	0.074	3.97989	0.81047	2.57901
Galerkin [26]	801	20	0.511	0.198	3.98206	0.81116	2.58133
Galerkin [27]	1001	20	0.220	0.086	3.97989	0.81046	2.57902
FDM [11, 27]	1001	20	196.1	67.35	4.41219	0.89734	2.85361
Bub-Gal. [29]	801	20	0.71913	0.25398	3.97989	0.80925	2.57501
L-S [30]	801	20	4.688	1.755	3.98203	0.80865	2.57302
L-S [31]	801	20	2.157	-	3.99046	0.82345	2.67399

**Table 3.** 1.A. The error norms and the rate of convergence at  $t = 20$ :  $C = 0.1, \Delta t = 0.1$  on  $-40 \leq x \leq 60$

N	$L_2 \times 10^3$	ROC	$L_\infty \times 10^3$	ROC
21	101.28714	-	26.72919	-
31	30.02020	3.12246	8.58606	2.91583
51	1.85904	5.58777	0.72326	4.96973
71	0.31324	5.38256	0.09773	6.04967
91	0.13030	3.53425	0.04730	2.92408

**1.B.** Secondly, the same parameters with in Application 1.A. are chosen except for  $C = 0.03$ . The numerical solutions for this situation are obtained by using  $\Delta t = 0.1$  and  $N = 161$ . The numerical movement of the solitary wave which has smaller amplitude is given in Figure 3. The solitary wave preserved its amplitude, velocity and shape during movement towards right. The error values for the time  $t = 20$  are given in Figure 3. The error norms  $L_2$  and  $L_\infty$  are computed at time  $t = 20$  as  $5.8878 \times 10^{-4}$  and  $2.1794 \times 10^{-4}$ , respectively. The peak position and the amplitude of the wave are given in Table 1. The amplitudes are seen almost constant. Table 4 shows a detailed report about the error norms. Some of the earlier results are added to the Table 4 for clear comparison. Undoubtedly it is seen that the present algorithm produced better results than those of the given works. Besides those, the three invariants are calculated. The analytic values of the three invariants are calculated with formulae given by Zaki [29] as  $I_1 = 2.1094074997496$ ,  $I_2 = 0.1273017186257$ ,  $I_3 = 0.3888059903539$ , respectively. The present numerical invariants are in good agreement with those of analytical values. The development of the three invariants during the simulation are plotted in Figure 4.

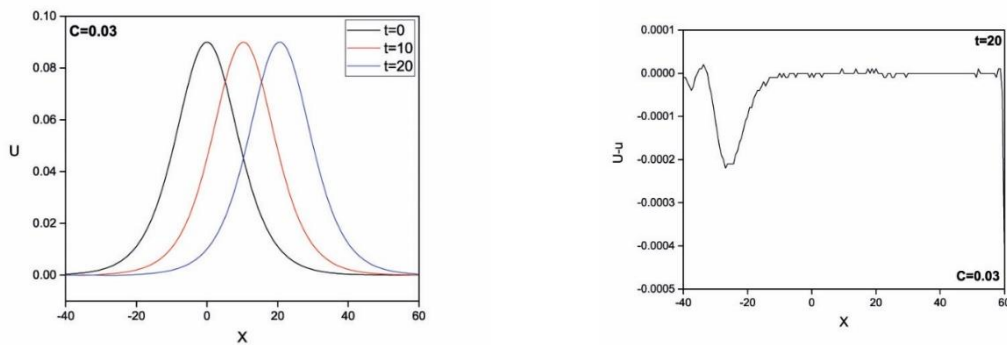


Figure 3. 1.B. Single solitary wave movement and maximum error for  $C = 0.03$  at time  $t = 20$

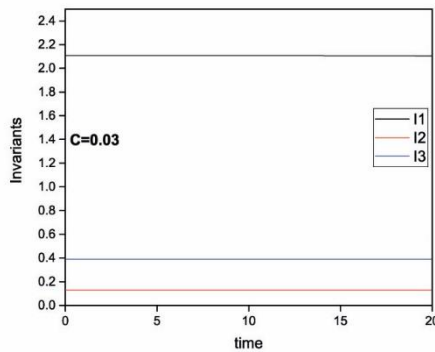


Figure 4. 1.B. Behaviour of the three invariants  $C = 0.03$

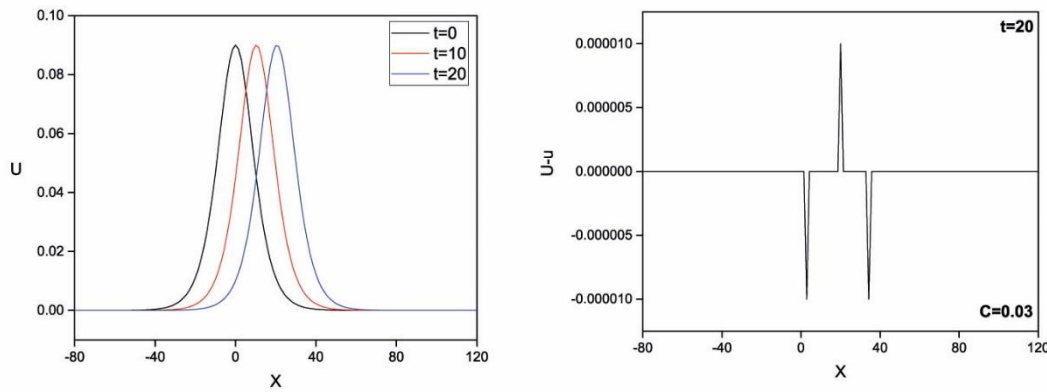
Table 4. 1.B. The error norms and the three invariants:  $C = 0.03$ ,  $\Delta t = 0.1$  on  $-40 \leq x \leq 60$

Method	N	t	$L_2 \times 10^3$	$L_\infty \times 10^3$	$I_1$	$I_2$	$I_3$
Present	161	0	-	-	2.10716	0.12734	0.38880
		5	0.50178	0.25334	2.10694	0.12730	0.38880
		10	0.58002	0.23788	2.10655	0.12730	0.38880
		15	0.58626	0.22906	2.10594	0.12730	0.38880
		20	0.58878	0.21794	2.10436	0.12730	0.38880
FDM[9]	1001	20	0.638	0.233	2.109	0.12730	0.38880
FDM[10] $\Delta t = 0.2$	1001	20	0.642	0.233	2.10949	0.12730	0.38880
Galerkin [28]	801	20	-	0.4315	2.10458	0.12730	0.38880
FDM [11, 27]	1001	20	14.45	3.996	2.333	0.14081	0.43005
Bub-Gal. [29]	2000	20	0.70663	0.70663	2.10238	0.12691	0.38759

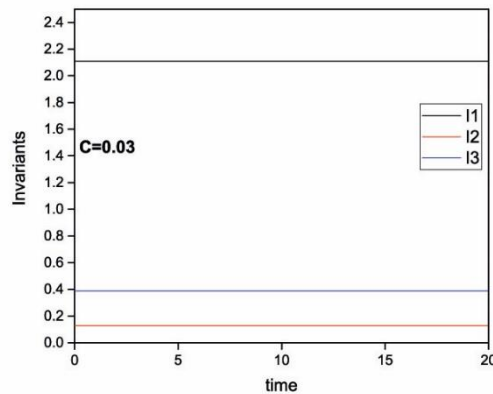


Cos-DQM [32]	34	20	0.662	0.221	2.10352	0.12730	0.38880
--------------	----	----	-------	-------	---------	---------	---------

**1.C.** For the third application, the same parameters with in Application 1.B. are chosen except for the solution domain  $-80 \leq x \leq 120$ . The numerical solutions are obtained by using  $\Delta t = 0.1$  and  $N = 141$ . Figure 5 shows the movement of the single solitary wave and error value at time  $t = 20$ . The error norms  $L_2$  and  $L_\infty$  are computed at time  $t = 20$  as  $9.04 \times 10^{-6}$  and  $2.65 \times 10^{-6}$ , respectively. Table 5 displays a detailed information about the error norms and the three invariants. Some of the earlier works are given with the present results in Table 5. Undoubtedly it is seen that the present algorithm produced better results than the given works. The analytic values of the three invariants are calculated with formulae given by Zaki [29] as  $I_1 = 2.1094074997496$ ,  $I_2 = 0.1273017186257$ ,  $I_3 = 0.3888059903539$ , respectively. The present numerical invariants are in good agreement with those of analytic values. The development of the three invariants during the simulation is plotted in Figure 6.



**Figure 5.** 1.C. Single solitary wave movement and maximum error for  $C = 0.03$  at time  $t = 20$



**Figure 6.** 1.C. Behaviour of the three invariants  $C = 0.03$

**Table 5.** 1.C. The error norms and the three invariants:  $C = 0.03, \Delta t = 0.1$  on  $-80 \leq x \leq 120$

Method	N	t	$L_2 \times 10^3$	$L_\infty \times 10^3$	$I_1$	$I_2$	$I_3$
Present	141	0	-	-	2.10941	0.12730	0.38881
		5	0.00239	0.00071	2.10941	0.12730	0.38881
		10	0.00465	0.00138	2.10941	0.12730	0.38881
		15	0.00688	0.00206	2.10940	0.12730	0.38881
		20	0.00904	0.00265	2.10940	0.12730	0.38881
FDM[9]	801	20	0.05460	0.01647	-	-	-
Coll. [16]	129	20	0.01367	0.00372	2.10940	0.12730	0.38880
Galerkin [26]	801	20	0.106	0.041	-	-	-
Bub-Gal. [29]	800	20	0.03097	0.00936	2.10972	0.12723	0.28887
Bub-Gal. [29]	1000	20	0.05222	0.01326	2.10970	0.12734	0.38892
Bub-Gal. [29]	2000	20	0.02746	0.01193	2.10938	0.12730	0.38879
Bub-Gal. [29]	4000	20	0.56067	0.13688	2.10600	0.12690	0.38750
L-S [30]	801	20	0.255	0.095	-	-	-
Cos-DQM [32]	34	20	0.662	0.221	2.10941	0.12730	0.38881

### 4.2. Wave Generation

The second test problem is wave generation which is based on Maxwellian initial condition problem of the RLW equation which has initial condition as below,

$$U(x, 0) = e^{-(x-7)^2}.$$

To be able to generate the waves with Maxwellian initial condition fixed value of the  $\varepsilon = 1$  and different values of the parameter  $\mu$  is chosen. All of the applications are studied over the fixed domain  $0 \leq x \leq 50$ .

**2.A.** For the first application of the wave generation for value of  $\mu = 0.04$  is chosen. The numerical solutions are obtained by using  $\Delta t = 0.05$  and  $N = 321$  up to time  $t = 25$ . The three invariants are given in Table 6. They are observed almost constant. The development of the three invariants during the simulation are plotted in Figure 7. The generation of the wave for  $\mu = 0.04$  is illustrated in Figure 8. At the final of the run, at time  $t = 25$  behind the leading wave, a tail is seen.

**Table 6.** 2. The three invariants of the wave generation for various values of  $\mu$

$\mu$	$\Delta t$	N	t	$I_1$	$I_2$	$I_3$
0.04	0.05	321	0	1.77245	1.30345	4.78327
			5	1.77245	1.30402	4.78612
			10	1.77245	1.30403	4.78617
			15	1.77245	1.30402	4.78616
			20	1.77245	1.30402	4.78616
			25	1.77245	1.30402	4.78616
0.01	0.05	321	0	1.77245	1.26585	4.78327
			5	1.77249	1.27674	4.82553
			10	1.77214	1.27492	4.81834
			15	1.77252	1.27190	4.80890
			20	1.77240	1.26889	4.79943
			25	1.77199	1.26609	4.79002
0.001	0.01	1201	0	1.77245	1.25457	4.78327
			5	1.77245	1.25971	4.80468
			10	1.77236	1.25827	4.79955
			15	1.77240	1.25649	4.79347
			20	1.77236	1.25482	4.78736
			25	1.77227	1.25331	4.78122

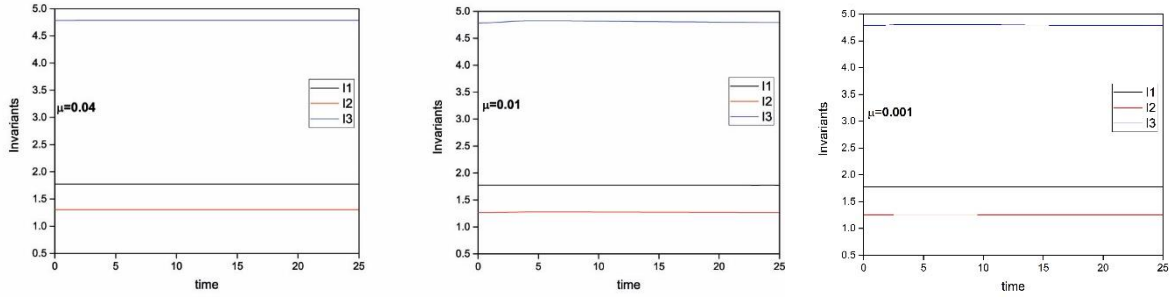


Figure 7. 2.A, 2.B, 2.C. The invariants of the wave generation

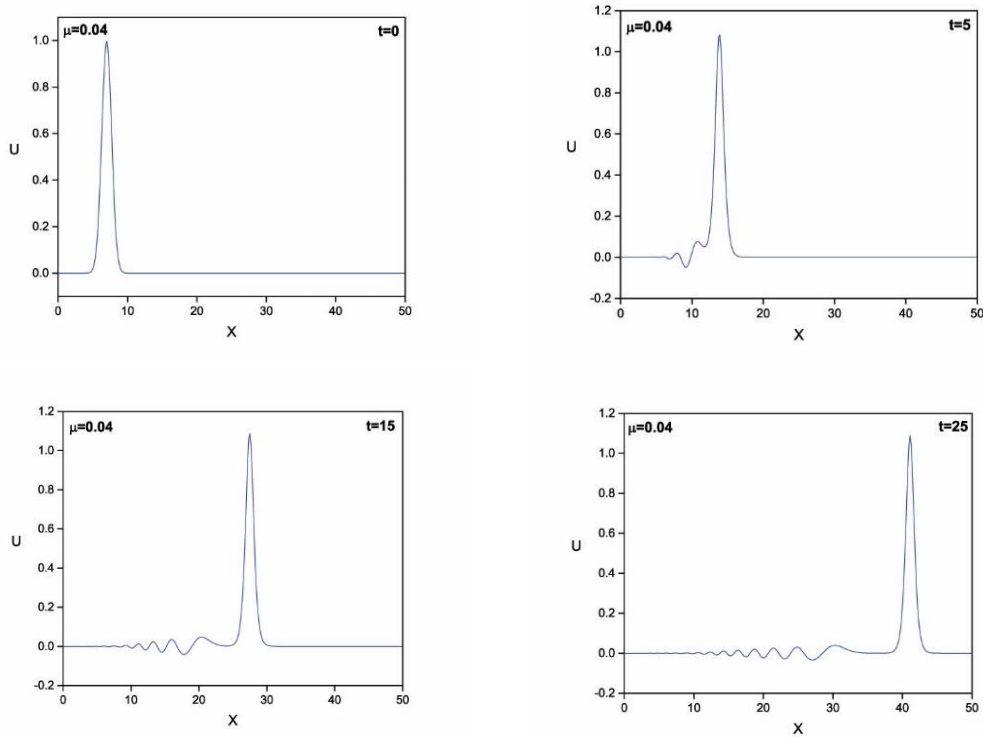


Figure 8. 2.A. The generation of the waves:  $\mu = 0.04$

**2.B.** For the second application of the wave generation for value of  $\mu = 0.01$  is chosen. The numerical solutions are obtained by using  $\Delta t = 0.05$  and  $N = 321$  up to time  $t = 25$ . The three invariants are given in Table 6. They are observed almost constant. The development of the three invariants during the simulation are plotted in Figure 7. The generation of the wave for  $\mu = 0.01$  is illustrated in Figure 9. At the final of the run, at time  $t = 25$  three waves are generated.

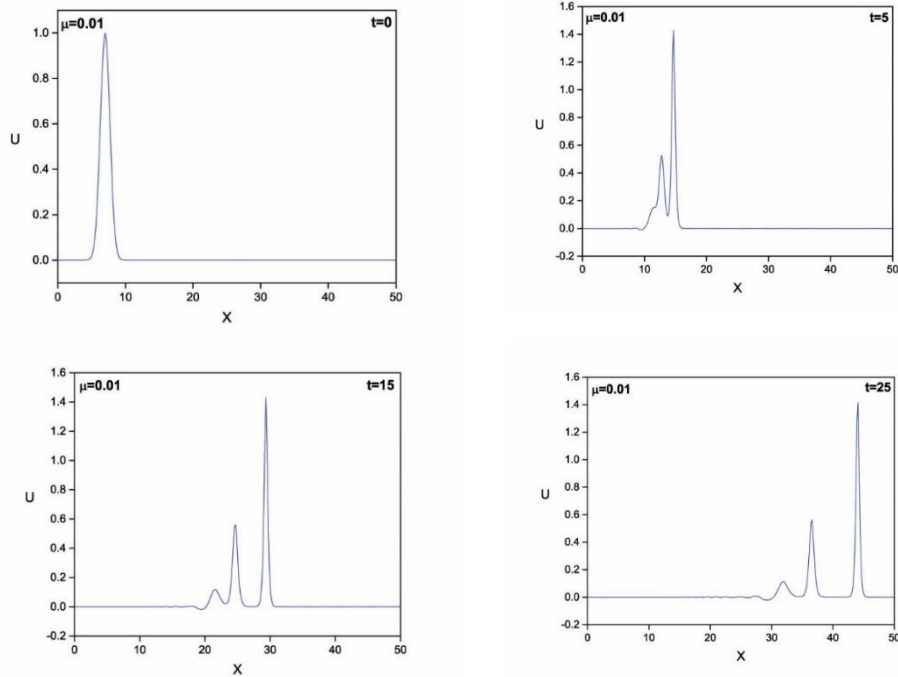


Figure 9. 2.B. The generation of the waves:  $\mu = 0.01$

2.C. For the second application of the wave generation for value of  $\mu = 0.001$  is chosen. The numerical solutions are obtained by using  $\Delta t = 0.01$  and  $N = 1201$  up to time  $t = 25$ . The three invariants are given in Table 6. They are observed almost constant. The development of the three invariants during the simulation is given in Figure 7. The generation of the wave for  $\mu = 0.001$  is illustrated in Figure 10. At the final of the run, at time  $t = 25$  nine waves are generated.

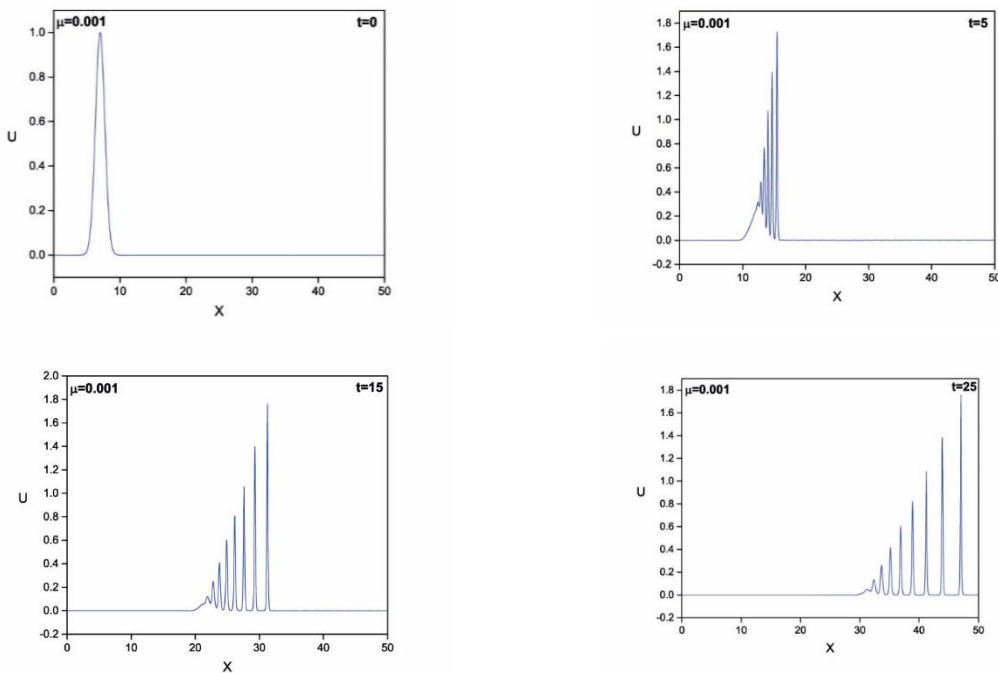


Figure 10. 2.C. The generation of the waves:  $\mu = 0.001$

## 5. CONCLUSION

In this work, an effective algorithm is used to obtain numerical solutions of the RLW equation. Three fundamental components of the algorithm are differential quadrature method, finite difference method and Rubin-Graves type linearizing technique. Solitary wave solution and wave generation solutions are obtained successfully. Two error norms and the three invariants are calculated and compared with other numerical results and analytical results. The rates of the convergence are calculated and reported. The present algorithm considerably improved the numerical solution of the RLW equation and may be useful for future investigations.

## CONFLICTS OF INTEREST

No conflict of interest was declared by the author.

## REFERENCES

- [1] Benjamin, T. B., Bona, J. L., Mahony, J. J., "Model Equations for Long Waves in Nonlinear Dispersive Systems", *Philosophical Transactions of the Royal Society A*, 272(1220): 47-78, (1972). DOI: 10.1098/rsta.1972.0032
- [2] Wazwaz, A., "Partial differential equations and solitary waves theory", Springer-Verlag Berlin Heidelberg, (2009).
- [3] Wazwaz, A., "Analytic study on nonlinear variants of the RLW and the PHI-four equations", *Communications in Nonlinear Science and Numerical Simulation*, 12(3): 314-327, (2007).
- [4] Lou, Y., "Bifurcation of travelling wave solutions in a nonlinear variant of the RLW equation", *Communications in Nonlinear Science and Numerical Simulation*, 12(8): 1488-1503, (2007).
- [5] Soliman, A.A., "Exact traveling wave solution of nonlinear variants of the RLW and the PHI-four equations", *Physics Letters A*, 368(5): 383-390, (2007).
- [6] Nuruddeen, R. I., Aboodh, K. S., Ali, K. K., "Investigating the tangent dispersive solitary wave solutions to the Equal Width and Regularized Long Wave equations", *Journal of King Saud University – Science*, 32(1): 677-681, (2020).
- [7] Feng, D., Li, J., Lu, J., He, T., "New explicit and exact solutions for a system of variant RLW equations", *Applied Mathematics and Computation*, 198(2): 715-720, (2008).
- [8] Roshid, H., Roshid, Md. M., Rahman, N., Pervin, Mst. R., "New solitary wave in shallow water, plasma and ion acoustic plasma via the GZK-BBM equation and the RLW equation", *Propulsion and Power Research*, 6(1): 49-57, (2017).
- [9] Kutluay, S., Esen, A., "A finite difference solution of the regularized long-wave equation", *Mathematical Problems in Engineering*, 2006: 1-14, (2006).
- [10] Inan, B., Bahadır, A. R., "A fully implicit finite difference scheme for the regularized long wave equation", *General Mathematics Notes*, 33(2):40-59, (2016).
- [11] Jain, P. C., Shankar, R., Singh, T.V., "Numerical solution of regularized long wave equation", *Communications in Numerical Methods in Engineering*, 9: 579-586, (1993).
- [12] Dag, I., Irk, D., Kaçmaz, O., Adar, N., "Trigonometric B-spline collocation algorithm for solving the RLW equation", *Applied and Computational Mathematics*, 15(1): 96-105, (2016).

- [13] Irk, D., Dag, I., Doğan, A., “Numerical integration of the RLW equation using cubic splines”, *The Australian & New Zealand Industrial and Applied Mathematics Journal*, 47(1): 131-142, (2005).
- [14] Islam, S., Haq, S., Ali, A., “A meshfree method for the numerical solution of the RLW equation”, *Journal of Computational and Applied Mathematics*, 223(2): 997–1012, (2009).
- [15] Saka, B., Dag, I., Irk, D., “Quintic B-spline collocation method for numerical solution of the RLW equation”, *The Australian & New Zealand Industrial and Applied Mathematics Journal*, 49(3): 389–410, (2008). DOI: 10.1017/S1446181108000072
- [16] Mokhtari, R., Mohammadi, M., “Numerical solution of GRLW equation using Sinc-collocation method”, *Computer Physics Communications*, 181(7): 1266–1274, (2010).
- [17] Raslan, K. R., “A computational method for the regularized long wave (RLW) equation”, *Applied Mathematics and Computation*, 167(2): 1101–1118, (2005).
- [18] Dag, I., Saka, B., Irk, D., “Application of cubic B-splines for numerical solution of the RLW equation”, *Applied Mathematics and Computation*, 159(2): 373–389, (2004).
- [19] Saka, B., Dag, I., “Quartic B-Spline Collocation Algorithms for Numerical Solution of the RLW Equation”, *Numerical Methods for Partial Differential Equations*, 23(3): 731–751, (2007).
- [20] Esen, A., Kutluay, S., “Application of a lumped Galerkin method to the regularized long wave equation”, *Applied Mathematics and Computation*, 174(2): 833–845, (2006).
- [21] Mei, L., Chen, Y., “Explicit multistep method for the numerical solution of RLW equation”, *Applied Mathematics and Computation*, 218(18): 9547–9554, (2012).
- [22] Mei, L., Chen, Y., “Numerical solutions of RLW equation using Galerkin method with extrapolation techniques”, *Computer Physics Communications*, 183(8): 1609–1616, (2012).
- [23] Gardner, L.R.T., Dag, I., “The boundary-forced regularised long-wave equation”, *Il Nuovo Cimento*, 110 B(12): 1487-1496, (1995).
- [24] Dag, I., Saka, B., Irk, D., “Galerkin method for the numerical solution of the RLW equation using quintic B-splines”, *Journal of Computational and Applied Mathematics*, 190(1-2): 532–547, (2006).
- [25] Saka, B., Dag, I., Doğan, A., “Galerkin method for the numerical solution of the RLW equation using quadratic B-splines”, *International Journal of Computer Mathematics*, 81(6): 727–739, (2004).
- [26] Doğan, A., “Numerical solution of RLW equation using linear finite elements within Galerkin’s method”, *Applied Mathematical Modelling*, 26(7): 771–783, (2002).
- [27] Gardner, L. R. T., Gardner, G. A., Dag, I., “A B-spline finite element method for the regularized long wave equation”, *Communications in Numerical Methods in Engineering*, 11: 59-68, (1995).
- [28] Görgülü, M. Z., Dag, I., Irk, D., “Simulations of solitary waves of RLW equation by exponential B-spline Galerkin method”, *Chinese Physics B*, 26(8): 080202, (2017).
- [29] Zaki, S. I., “Solitary waves of the splitted RLW equation”, *Computer Physics Communications*, 138(1): 80–91, (2001).
- [30] Gardner, L. R. T., Gardner, G. A., Doğan, A., “A least-squares finite element scheme for the RLW equation”, *Communications in Numerical Methods in Engineering*, 12: 795-804, (1996).

- [31] Dag, I., “Least-squares quadratic B-spline finite element method for the regularized long wave equation”, *Computer Methods in Applied Mechanics and Engineering*, 182(1-2): 205-215, (2000).
- [32] Dag, I., Korkmaz, A., Saka, B., “Cosine expansion-based differential quadrature algorithm for numerical solution of the RLW equation”, *Numerical Methods for Partial Differential Equations*, 26(3): 544–560, (2010).
- [33] Sloan, D. M., “Fourier pseudospectral solution of the regularised long wave equation”, *Journal of Computational and Applied Mathematics*, 36(2): 159-179, (1991).
- [34] Hosseini, M. M., Ghaneai, H., Mohyud-Din, S. T., Usman, M., “Tri-prong scheme for regularized long wave equation”, *Journal of the Association of Arab Universities for Basic and Applied Sciences*, 20(1): 68–77, (2016).
- [35] El-Danaf, T. S., Ramadan, M. A., Abd Alaal, F. E. I., “The use of adomian decomposition method for solving the regularized long-wave equation”, *Chaos, Solitons and Fractals*, 26(3): 747–757, (2005).
- [36] Çelik, E., Bayram, M., “On the numerical solution of differential-algebraic equations by Pade series”, *Applied Mathematics and Computation*, 137: 151–160, (2003).
- [37] Çelik, E., Karaduman, E., Bayram, M., “Numerical method to solve chemical differential-algebraic equations”, *International Journal of Quantum Chemistry*, 89: 447–451, (2002).
- [38] Çelik, E., Bayram, M., Yeloğlu, T., “Solution of differential-algebraic equations (DAEs) by adomian decomposition method”, *International Journal Pure & Applied Mathematical Sciences*, 3(1): 93-100, (2006).
- [39] Bellman, R., Kashef, B. G., Casti, J., “Differential quadrature: a technique for the rapid solution of nonlinear differential equations”, *Journal of Computational Physics*, 10(1): 40-52, (1972).
- [40] Başhan, A., “Highly efficient approach to numerical solutions of two different forms of the modified Kawahara equation via contribution of two effective methods”, *Mathematics and Computers in Simulation*, 179: 111-125, (2021).
- [41] Başhan, A., “An effective approximation to the dispersive soliton solutions of the coupled KdV equation via combination of two efficient methods”, *Computational and Applied Mathematics*, 39(2): 80, (2020).
- [42] Başhan, A., Yağmurlu, N. M., Uçar, Y., Esen, A., “A new perspective for the numerical solutions of the cmKdV equation via modified cubic B-spline differential quadrature method”, *International Journal of Modern Physics C*, 29(6): 1850043, (2018).
- [43] Başhan, A., Yağmurlu, N. M., Uçar, Y., Esen, A., “Finite difference method combined with differential quadrature method for numerical computation of the modified equal width wave equation”, *Numerical Methods for Partial Differential Equations*, 37(1): 690–706, (2021).
- [44] Shu, C., “Differential Quadrature and its application in engineering”, Springer-Verlag, London, (2000).
- [45] Uçar, Y., Yağmurlu, N. M., Başhan, A., “Numerical Solutions and Stability Analysis of Modified Burgers Equation via Modified Cubic B-spline Differential Quadrature Methods”, *Sigma Journal of Engineering and Natural Sciences*, 37 (1): 129-142, (2019).

- [46] Başhan, A., Uçar, Y., Yağmurlu, N. M., Esen, A., “Numerical solutions for the fourth order extended Fisher-Kolmogorov equation with high accuracy by differential quadrature method”, *Sigma Journal of Engineering and Natural Sciences*, 9(3): 273-284, (2018).
- [47] Başhan, A., “A numerical treatment of the coupled viscous Burgers’ equation in the presence of very large Reynolds number”, *Physica A*, 545: 123755, (2020).
- [48] Başhan, A., “Quartic B-spline Differential Quadrature Method for Solving the Extended Fisher-Kolmogorov Equation”, *Erzincan University, Journal of Science and Technology*, 12(1): 56-62, (2019).
- [49] Crank, J., Nicolson, P., “A practical method for numerical evaluation of solutions of partial differential equations of the heat conduction type”, *Mathematical Proceedings of the Cambridge Philosophical Society*, 43(1): 50-67, (1947). DOI: 10.1017/S0305004100023197
- [50] Rubin, S.G., Graves, R.A., “A cubic spline approximation for problems in fluid mechanics”, *National Aeronautics and Space Administration, Technical Report*, Washington, (1975).
- [51] Chegini, N.G., Salaripناه, A., Mokhtari, R., Isvand, D., “Numerical solution of the regularized long wave equation using nonpolynomial splines”, *Nonlinear Dynamics*, 69: 459–471, (2012).
- [52] Olver, P.J., “Euler operators and conservation laws of the BBM equation”, *Mathematical Proceedings of the Cambridge Philosophical Society*, 85(01): 143-160, (1979).