Computational Methods for Differential Equations
http://cmde.tabrizu.ac.ir
Vol. 6, No. 2, 2018, pp. 238-247

An improved collocation method based on deviation of the error for solving BBMB equation

Reza Parvaz
Department of Mathematics, University of Mohaghegh Ardabili,
56199-11367 Ardabil, Iran.
E-mail: rparvaz@uma.ac.ir

Mohammad Zarebnia*
Department of Mathematics, University of Mohaghegh Ardabili,
56199-11367 Ardabil, Iran.
E-mail: Zarebnia@uma.ac.ir

Abstract In this paper, we improve B-spline collocation method for Benjamin-Bona-Mahony-Burgers (BBMB) by using defect correction principle. The exact finite difference scheme is used to find defect and the defect correction principle is used to improve collocation method. The method is tested on some model problems and the numerical results have been obtained and compared.

Keywords. Exact finite difference, Defect principle, Collocation method, Deviation of the error.

2010 Mathematics Subject Classification. 65M70, 41A15, 65M06.

1. Introduction

We consider Benjamin-Bona-Mahony-Burgers (BBMB) equation as follows

$$u_t - u_{xxt} - \alpha u_{xx} + \beta u_x + u u_x = 0, \qquad x \in [a, b], \;\; t \in [0, T], \tag{1.1}$$

where α and β are positive constants. The initial condition and boundary conditions are given as

$$u(x, 0) = h(x), \qquad x \in [a, b], \tag{1.2}$$

$$u(a, t) = u(b, t) = 0. \tag{1.3}$$

This equation can be found in many branches of engineering, for example see [15]. For $\alpha = 0$, Eq. (1.1) is called the Benjamin-Bona-Mahony (BBM) equation or regularized long-wave (RLW) equation. BBM equation was introduced in an article titled "Model equations for long waves in nonlinear dispersive systems " [4]. In this paper we use the defect correction principle for improve collocation method based on cubic B-spline. The B-spline collocation method for BBMB equation was introduced in [20]. Also defect correction principle and its application can be found in [3, 5, 17, 14]. The error estimation based on defect that will be used in this paper, has been introduced in [14]. In the previous work, we study the B-spline collocation method for

Received: 4 July 2017 ; Accepted: 11 March 2018.
∗ Corresponding author.

BBMB equation. In recent years, various type of methods for solving BBMB equation has been introduced in many papers. For example, Lucas polynomials has been used in [11] and non-polynomial spline method has been introduced in [13]. Also other methods can be found in [1, 2, 6, 12, 18, 21].

In the first step of the proposed algorithm, exact finite difference is used to find defect for collocation method, and in the second step, we improve collocation method by using the defect correction method.

The paper is organized as follows. In Section 2, collocation method is explained. In Section 3, we develop an algorithm for improved collocation method. In Section 4, examples are presented. Also in Section 5, a conclusion of the paper is given.

2. Collocation method

In this section, we give a brief introduction to the cubic B-spline collocation method, more details about this method can be found in [20]. In the first step of the method the domain $[a, b]$ is partitioned into N equidistant intervals with length $h = \frac{b-a}{N}$, by the knots $x_i = x_0 + ih$ where $i = 0, \dots, N$ such that $a = x_0 < x_1 < \dots < x_{N-1} < x_N = b$. We define partition Δ as

$$\Delta := \{a = x_0 < \dots < x_n = b\}.$$

By using the forward Euler scheme for time integration (1.1) and δt as the time step and following formula

$$(uu_x)^{n+1} = 3(uu_x)^n - u^{n-1}u_x^n - u^n u_x^{n-1} + O(\delta t^2),$$

Eq. (1.1) can be written as a second-order differential equation (2.1)

$$u^{n+1} + su_x^{n+1} + qu_{xx}^{n+1} = \Psi^n, \tag{2.1}$$

where

$$\Psi^n = u^n - \frac{\beta\delta t}{2}u_x^n + \big(-1 + \frac{\alpha\delta t}{2}\big)u_{xx}^n - \frac{\delta t}{2}(4u^n u_x^n - u^{n-1}u_x^n - u^n u_x^{n-1}), \tag{2.2}$$

$s = \dfrac{\beta\Delta t}{2}$, $q = -1 - \dfrac{\alpha\Delta t}{2}$ and $u^i(x) := u(x, i\delta t)$.

Now we define the approximate solution for u^{n+1} as $u^{n+1}(x) \approx U^{n+1}(x) := \sum_{i=-1}^{N+1} c_i^{n+1} B_i(x)$ where $B_i(x)$ are the cubic B-spline basis functions. B-spline function is defined as [16]

$$B_i(x) = \frac{1}{6h^3}\begin{cases} (x - x_{i-2})^3, & x \in [x_{i-2}, x_{i-1}), \\ h^3 + 3h^2(x - x_{i-1}) + 3h(x - x_{i-1})^2 & \\ -3(x - x_{i-1})^3, & x \in [x_{i-1}, x_i), \\ h^3 + 3h^2(x_{i+1} - x) + 3h(x_{i+1} - x)^2 & \\ -3(x_{i+1} - x)^3, & x \in [x_i, x_{i+1}), \\ (x_{i+2} - x)^3, & x \in [x_{i+1}, x_{i+2}). \\ 0, & otherwise. \end{cases}$$

Substituting the approximate solution $U^{n+1}(x)$ for $u^{n+1}(x)$ in (2.1) and using Eqs. (1.2)-(1.3), we get

$$AC = h^2 Q, \tag{2.3}$$

where

$$A = \begin{pmatrix} -4\acute{a}+\acute{b} & -\acute{a}+\acute{c} & 0 & \dots & 0 \\ \acute{a} & \acute{b} & \acute{c} & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots & \\ 0 & \dots & \acute{a} & \acute{b} & \acute{c} \\ 0 & \dots & 0 & \acute{a}-\acute{c} & \acute{b}-4\acute{c} \end{pmatrix}, \tag{2.4}$$

$$C = (c_0^{n+1}, c_1^{n+1}, \dots, c_{N-1}^{n+1}, c_N^{n+1})^T,$$

$$Q = \big(\Psi^n(x_0), \dots, \Psi^n(x_N)\big)^T, \tag{2.5}$$

and $\acute{a} = \frac{h^2}{6} - \frac{sh}{2} + q$, $\acute{b} = \frac{4h^2}{6} - 2q$, $\acute{c} = \frac{h^2}{6} + \frac{sh}{2} + q$. The above system can be solved by using Thomas algorithm. According to the above method, before start any calculating, we must find $U^{n+1}(x)$. Let $U^1(x) = \sum_{i=-1}^{N+1} c_i^1 B_i(x)$ then by using the forward Euler scheme for time integration (1.1) and boundary condition we can find

$$\acute{a}c_{i-1}^1 + \acute{b}c_i^1 + \acute{c}c_{i+1}^1 + \frac{\delta t}{24h}(c_{i-1}^1 + 4c_i^1 + c_{i+1}^1)(-c_{i-1}^1 + c_{i+1}^1) = \Phi^i,$$

$$i = 0, \dots, n, \tag{2.6}$$

$$c_{-1}^1 + c_0^1 + c_1^1 = 0, \tag{2.7}$$

$$c_{n-1}^1 + c_n^1 + c_{n+1}^1 = 0, \tag{2.8}$$

where

$$\Phi^i = u(x_i, 0) + (-1 + \alpha\frac{\delta t}{2})u_{xx}(x, 0) - \beta\frac{\delta}{2}u_x(x_i, 0) - \frac{\delta}{2}u(x, 0)u_x(x, 0). \tag{2.9}$$

Based on above discussion, collocation method can be written as Algorithm 1.

3. Improvement of collocation method

In this section, a finite difference scheme and exact finite difference scheme are used to define defect and in the next step, by using defect principle and deviation of the error, we improve collocation method.

According to Eq. (2.1) in the collocation method in $t = (n+1)\delta t$ we must solve the following equation,

$$u_{xx}^{n+1}(x) = f(x), \ a \le x \le b, \tag{3.1}$$

$$u^{n+1}(a) = u^{n+1}(b) = 0, \tag{3.2}$$

Algorithm 1: Collocation method

```
Input  : a, b, α, β, N, n.
Output: U^n(x).
h := (b−a)/N.
x_i = a + ih.
á := h^2/6 − sh/2 + q,  b́ := 4h^2/6 − 2q,  ć := h^2/6 + sh/2 + q.
Find {c_i^1}_{i=0}^{N+1} by using (2.6),(2.7) and (2.8).
Define A as (2.4).
for j=2:n do
    Define Q by using (2.5).
    Find {c_i^j}_{i=0}^{N+1} by using (2.3).
end
Consider U^n(x) as Σ_{i=−1}^{N+1} c_i^n B_i(x).
```

where $f(x) = \frac{\Psi^n(x_i)}{q} - \frac{s}{q}u_x^{n+1}(x_i) - \frac{1}{q}u^{n+1}(x_i)$. By using Taylor expansion for $u^{n+1}(x_{i\pm1})$ $i = 1, \dots, N-1$, we have

$$u^{n+1}(x_i + h) = u^{n+1}(x_i) + hu_x^{n+1}(x_i) + h^2 \int_0^1 (1-\xi) f(x_i + \xi h) d\xi, \tag{3.3}$$

$$u^{n+1}(x_i - h) = u^{n+1}(x_i) - hu_x^{n+1}(x_i) + h^2 \int_0^1 (1-\xi) f(x_i - \xi h) d\xi, \tag{3.4}$$

from (3.3) and (3.4), the exact finite difference scheme for Eq.(3.1) can be written as

$$\begin{aligned}(L_\Delta^{(2)} u^{n+1})_i &:= \frac{u^{n+1}(x_{i+1}) - 2u^{n+1}(x_i) + u^{n+1}(x_{i-1})}{h^2} \\ &= \int_{-1}^{1} (1-|\xi|) f(x_i + \xi h) d\xi =: \mathcal{I}_\Delta\big(f(x_i)\big), i = 1, \dots, N-1.\end{aligned}$$

According to collocation method, since

$$U_{xx}^{n+1}(x_i) + \frac{s}{q} U_x^{n+1}(x_i) + \frac{1}{q} U^{n+1}(x_i) - \frac{\Psi^n(x_i)}{q} = 0, \ \ i = 1, \dots, N-1,$$

therefore, by using the exact difference scheme we define the defect at x_i as

$$D_i := (L_\Delta^{(2)} U^{n+1})_i - \mathcal{I}_\Delta\big(p(x_i)\big), \ \ i = 1, \dots, N-1, \tag{3.5}$$

where $p(x) = \frac{\Psi^n(x)}{q} - \frac{s}{q} U_x^{n+1}(x) - \frac{1}{q} U^{n+1}(x)$.
In order to compute integral term in Eq. (3.5), we use the three-point Gaussian quadrature method. This method is exact for polynomials up to degree 5. Then we can write

$$\mathcal{I}_\Delta\big(p(x_i)\big) \approx Q_\Delta\big(p(x_i)\big) = \sum_{j=0}^{2} \omega_j p_i^+(\alpha_j) + \sum_{j=0}^{2} \omega_j p_i^-(\alpha_j), i = 1, \dots, N-1,$$

where $\alpha_0 = 0,\ \alpha_1 = -\alpha_2 = \sqrt{\frac{3}{5}}$ and $\omega_0 = \frac{8}{9},\ \omega_1 = \omega_2 = \frac{5}{9}$,

$$p_i^+(x) = \frac{1-x}{4} p(x_i + h\frac{x+1}{2}),$$
$$p_i^-(x) = \frac{1-x}{4} p(x_i - h\frac{x+1}{2}).$$

Then we have

$$D_i \approx (L_\Delta^{(2)} U^{n+1})_i - Q_\Delta\big(p(x_i)\big),\ i = 1, \ldots, N-1. \tag{3.6}$$

In the next step we find the deviation of the error. For Eq.s (3.1)-(3.2) we can write a general finite difference scheme as

$$\frac{\widehat{y}_{i+1} - 2\widehat{y}_i + \widehat{y}_{i-1}}{h^2} + \frac{s}{q}\frac{\widehat{y}_{i+1} - \widehat{y}_{i-1}}{2h} + \frac{1}{q}\widehat{y}_i = \frac{\Psi^n(x_i)}{q}, i = 1, \ldots, N-1, \tag{3.7}$$

$$\widehat{y}_0 = \widehat{y}_N = 0. \tag{3.8}$$

Let $\widehat{\widehat{y}}$ be defined as the solution of the following finite difference scheme,

$$\frac{\widehat{\widehat{y}}_{i+1} - 2\widehat{\widehat{y}}_i + \widehat{\widehat{y}}_{i-1}}{h^2} + \frac{s}{q}\frac{\widehat{\widehat{y}}_{i+1} - \widehat{\widehat{y}}_{i-1}}{2h} + \frac{1}{q}\widehat{\widehat{y}}_i = \frac{\Psi^n(x_i)}{q} + D_i, i = 1, \ldots, N-1, \tag{3.9}$$

$$\widehat{\widehat{y}}_0 = \widehat{\widehat{y}}_N = 0. \tag{3.10}$$

Now we define D as $D := (D_1, \ldots, D_{N-1})^T$ and $\widehat{y} := \{\widehat{y}_i; i = 1, \ldots, N-1\}$, $\widehat{\widehat{y}} := \{\widehat{\widehat{y}}_i; i = 1, \ldots, N-1\}$. Also for any function z, we define $\mathcal{R}(z) := \{z(x_i)\ ; i = 1, \ldots, N-1\}$. For small D, we can say [3, 5, 17, 14]

$$\widehat{\widehat{y}} - \mathcal{R}(U^{n+1}) \approx \widehat{y} - \mathcal{R}(u^{n+1}).$$

Then we have

$$\varepsilon := \widehat{\widehat{y}} - \widehat{y} \approx \mathcal{R}(U^{n+1}) - \mathcal{R}(u^{n+1}) := e,$$

where e is the error and ε is the error estimated. We can write the deviation of the error as

$$deviation\ of\ the\ error = e - \varepsilon.$$

Using above discuss, improved collocation method can be written as follows

$$\mathcal{R}(u^{n+1}) \approx \mathcal{R}(U^{n+1}) - \varepsilon.$$

The improved collocation method is written in Algorithm 2.

4. Numerical Examples

In order to illustrate the theoretical results, we consider some test problems. Also to show the efficiency of the present method for our problem in comparison with the analytical solution, we report L_∞ and L_2 using formulae

$$L_\infty = \max_i \mid U(x_i,t) - u(x_i,t) \mid,\ L_2 = (h\sum_i \mid U(x_i,t) - u(x_i,t) \mid^2)^{\frac{1}{2}},$$

where U is numerical solution and u denotes analytical solution. In this section, we use Mathematica-9 software.

Algorithm 2: Improved collocation method

Input : $a, b, \alpha, \beta, N, n$.

Output: Improved collocation solution.

$h := \frac{b-a}{N}$

$x_i = a + ih$

$\acute{a} := \frac{h^2}{6} - \frac{sh}{2} + q,\ \acute{b} := \frac{4h^2}{6} - 2q,\ \acute{c} := \frac{h^2}{6} + \frac{sh}{2} + q.$

Find $\{c_i^1\}_{i=0}^{N+1}$ by using (2.6),(2.7) and (2.8).

Define A as (2.4).

for j=2:n **do**

  Define Q by using (2.5).

  Find $\{c_i^j\}_{i=0}^{N+1}$ by using (2.3).

  Define $U^j(x) = \sum_{i=-1}^{N+1} c_i^j B_i(x)$.

  Find defect by using (3.6).

  Solve finite difference scheme (3.7)-(3.8).

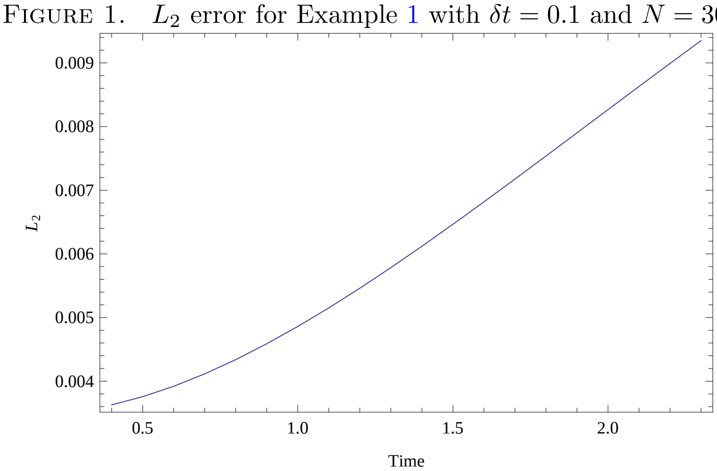
  Solve finite difference scheme (3.9)-(3.10).

  Define $\varepsilon := \widehat{\widehat{y}} - \widehat{y}$.

  Improve collocation solution by using $\mathcal{R}(U^{n+1}) - \varepsilon$.

end

FIGURE 1. L_2 error for Example 1 with $\delta t = 0.1$ and $N = 30$.



Example 1. We consider Eq. (1.1) with $\alpha = 0$ and $\beta = 1$ in the interval $[-40, 60]$, with the analytical solution $u(x,t) = 3c\,\text{sech}^2(k(x - vt - x_0))$, where $c = 0.1$, $v = 1 + c$, $x_0 = 0$, $k = \sqrt{\frac{c}{4v}}$. Table 1 and Table 2 give a comparisons between improved collocation method and collocation method for different partitions. Also Figure 3 shows that the solution obtained by our method is close to the analytical solution.

FIGURE 2. L_∞ error for Example 1 with $\delta t = 0.1$ and $N = 30$.

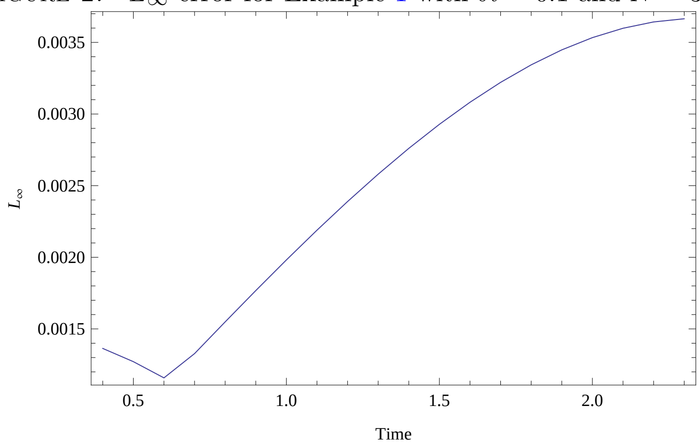
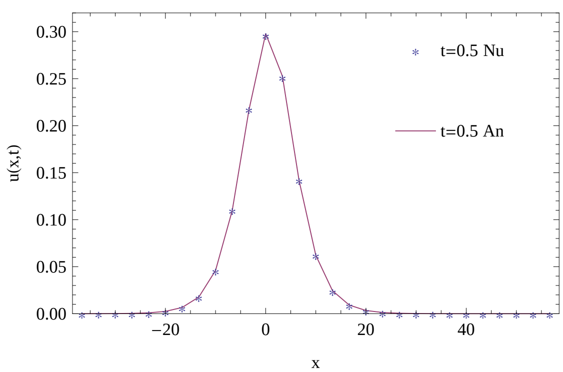


FIGURE 3. Analytical-estimated graph of Example 1 with $\delta t = 0.1$, $N = 30$.



Figures 1 and 2 display L_2 error and L_∞ error of the improved collocation method. In Table 3, the presented method has been compared with methods in [9, 8, 19].

Example 2. In this example BBMB equation is considered with $\alpha = 0$, $\beta = 1$ and $[-10, 30]$, with the initial condition $u(x, 0) = sech^2(x/4)$. The analytical solution is given by [7] as $u(x, t) = sech^2(x/4 - t/3)$.

Example 3. We consider Eq.(1.1) with $\alpha = 1$, $\beta = 1$, $[a, b] = [-10, 10]$ and the initial condition $u(x, 0) = \exp(-x^2)$. Figure 5 shows the approximated solution with $\delta t = 0.01$ and $N = 200$. This figure shows the same behavior as in [10, 20].

FIGURE 4. Analytical-estimated graph of Example 2 with $\delta t = 0.01$, $N = 50$.

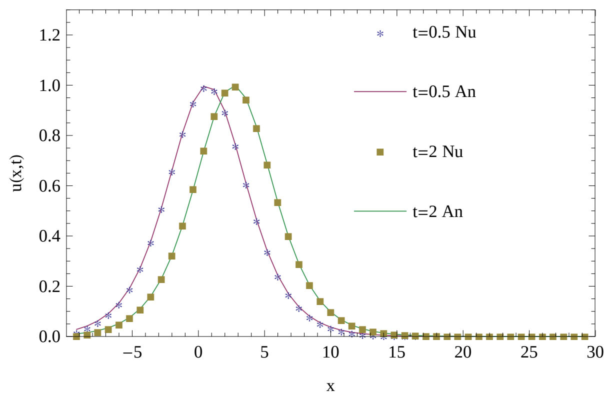


TABLE 1. Comparison of errors for Example 1 with $N = 40, \delta t = 0.1$.

Method	Error	1	2	3	4
Improved collocation method	$L_\infty \times 10^3$	1.16146	1.40295	2.16317	2.42435
	$L_2 \times 10^3$	2.50472	3.7120	5.12807	6.5856
Collocation method [15]	$L_\infty \times 10^3$	1.20730	1.41962	2.21249	2.46378
	$L_2 \times 10^3$	2.58751	3.82596	5.25022	6.70887

TABLE 2. Comparison of errors for Example 1 with $N = 20, \delta t = 0.01$.

Method	Error	0.5	1	1.5	2
Improved collocation method	$L_\infty \times 10^3$	1.61184	3.22385	5.99015	8.90136
	$L_2 \times 10^3$	6.06002	9.93017	1.72764	2.67476
Collocation method [15]	$L_\infty \times 10^3$	1.62821	3.2423	6.00809	8.91827
	$L_2 \times 10^3$	6.10323	9.98487	1.73256	2.67915

TABLE 3. Comparison of errors for Example 1 with $\delta t = 0.01$ in $t = 20$.

Method	N	$L_2 \times 10^3$	$L_\infty \times 10^3$
Our method	401	0.170212	0.0685521
	801	0.0371011	0.0126135
Method I in [9]	401	0.27	0.07
Method II in [9]	401	0.26	0.07
Method in [8]	801	0.19	0.07
Method in [19]	801	0.71	0.25

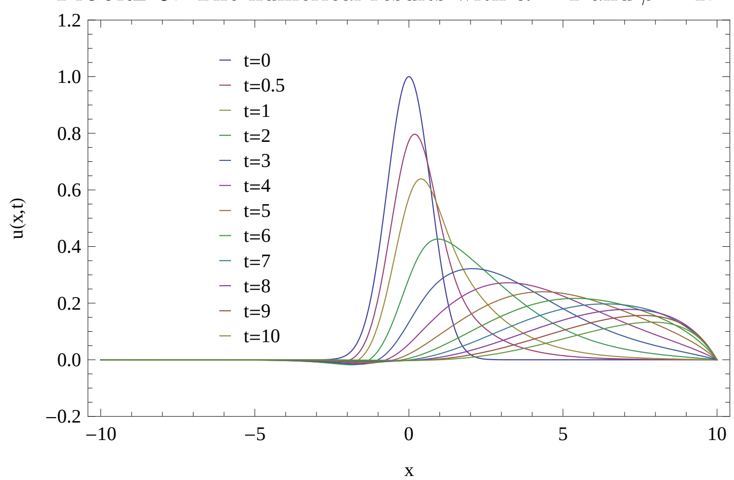
TABLE 4. Comparison of errors for Example 2 with $N = 40, \delta t = 0.1$.

Method	Error	0.4	0.5	0.7	1.1
Improved collocation method	$L_\infty \times 10^3$	8.85816	8.80431	8.68408	8.39897
	$L_2 \times 10^3$	10.9784	11.0908	11.3672	1.21008
Collocation method [20]	$L_\infty \times 10^3$	8.92138	8.85892	8.72373	8.41641
	$L_2 \times 10^3$	11.1237	11.2447	11.5382	12.3039

TABLE 5. Comparison of errors for Example 2 with $N = 50, \delta t = 0.01$.

Method	Error	0.88	1	1.28	1.45
Improved collocation method	$L_\infty \times 10^3$	9.82729	9.60314	9.09979	8.80559
	$L_2 \times 10^3$	11.8387	11.9084	12.141	12.3238
Collocation method [20]	$L_\infty \times 10^3$	9.82878	9.60428	9.10025	8.80573
	$L_2 \times 10^3$	11.8476	11.9177	12.1512	12.3346

FIGURE 5. The numerical results with $\alpha = 1$ and $\beta = 1$.



5. CONCLUSION

The deviation of the error based on defect principle has been used to improve the collocation method. The method is based on the forward Euler scheme formulation for time integration and improved collocation method for space integration. The numerical results given in the previous section demonstrate the accuracy of the proposed method in this research.

REFERENCES

[1] K. Al-Khaled, S. Momani, and A. Alawneh, *Approximate wave solutions for generalized Benjamin-Bona-Mahony-Burgers equations*, Appl. Math. Comput., *171* (2005), 281–292. DOI 10.1016/j.amc.2005.01.056.

[2] J. Avrin and J. A. Goldstein, *Global existence for the Benjamin-Bona-Mahony equation in arbitrary dimensions*, Nonlinear Anal., *9* (1985), 861–865. DOI 10.1016/0362-546X(85)90023-9.

[3] W. Auzinger, O. Koch, and A. Saboor Bagherzadeh, *Error estimation based on locally weighted defect for boundary value problems in second order ordinary differential equations*, BIT. Numer. Math., *54* (2014), 873–900. DOI 10.1007/s10543-014-0488-y.

[4] T. B. Benjamin, J. L. Bona, and J. J. Mahony, *Model equations for long waves in nonlinear dispersive systems*, Philos. T. Roy. Soc. A (1972) 47–78.

[5] K. Bohmer, P. Hemker, and H. J.Stetter, *The defect correction approach* , Computing, Suppl., *5* (1984), 1–32. DOI 10.1007/978-3-7091-7023-6.

[6] A. O. Celebi, V. K. Kalantarov, and M. Polat, *Attractors for the generalized Benjamin-Bona-Mahony equation*, J. Differ. Equ., *157* (1999), 439–451. DOI 10.1006/jdeq.1999.3634.

[7] Y. Chen, B. Li, and H. Zhang, *Exact solutions of two nonlinear wave equations with simulation terms of any order*, Commun. Nonlinear Sci. Numer. Simul., *10* (2005) 133–138. DOI 10.1016/S1007-5704(03)00121-7.

[8] I. Dağ, B. Saka, and D. Irk, *Galerkin method for the numerical solution of the RLW equation using quintic b-splines*, J. Comput. Appl. Math., *190* (2006), 532–547. DOI 10.1016/j.cam.2005.04.026.

[9] A. Korkmaz and I. Dağ, *Numerical simulations of boundary-forced RLW equation with cubic b-spline-based differential quadrature methods*, Arab. J. Sci. Eng., *38*, (2013) 1151–1160. DOI 10.1007/s13369-012-0353-8.

[10] K. Omrani and M. Ayadi, *Finite difference discretization of the Benjamin- Bona-Mahony-Burgers equation*, Numer. Methods Partial Differ. Equ., *24* (2008), 239–248. DOI 10.1002/num.

[11] Ö. Oruç, *A new algorithm based on Lucas polynomials for approximate solution of 1D and 2D nonlinear generalized Benjamin-Bona-Mahony-Burgers equation*, Comput. Math. Appl., *74*(12) (2017), 3042–3057. DOI 10.1016/j.camwa.2017.07.046.

[12] M. A. Raupp, *Galerkin methods applied to the Benjamin-Bona-Mahony equation*, Bol. Soc. Brasil. Mat., *6* (1975), 65–77.

[13] A. S. V. Ravi Kanth's and S. Deepika, *Non-Polynomial Spline Method for One Dimensional Nonlinear Benjamin-Bona-Mahony-Burgers Equation*, Int. J. Nonlinear. Sci., *3-4* (2017), 277–284. DOI 10.1515/ijnsns-2016-0136.

[14] A. Saboor Bagherzadeh, *Defect-based error estimation for higher order differential equations*, PhD thesis, Vienna University of Technology, (2011).

[15] G. Stephenson, *Partial Differential Equations for Scientists and Engineers*, Imperial College Press, 1996.

[16] J. Stoer and R. Bulirsch, *Introduction to numerical analysis*, third edition ,Springer-Verlg, 2002.

[17] H. J. Stetter, *The defect correction principle and discretization methods*, Numer. Math., *29* (1978), 425-443.

[18] B. Wang, *Random attractors for the stochastic Benjamin-Bona-Mahony equation on unbounded domains*, J. Differ. Equ., *246* (2009), 2506-2537. DOI 10.1016/j.jde.2008.10.012.

[19] S. I. Zaki, *Solitary waves of the split RLW equation*, Comput. Phys. Commun., *138* (2001), 80–91. DOI 10.1016/S0010-4655(01)00200-4.

[20] M. Zarebnia and R. Parvaz, *On the numerical treatment and analysis of Benjamin-Bona-Mahony-Burgers equation*, Appl. Math. Comput., *284* (2016), 79–88. DOI 10.1016/j.amc.2016.02.037.

[21] M. Zarebnia and R. Parvaz, *Numerical study of the Benjamin-Bona-Mahony-Burgers equation*, Bol. Soc. Paran. Mat., *35* (2017), 127-138. DOI 10.5269/bspm.v35i1.28804.