ISSN 1749-3889 (print), 1749-3897 (online)
International Journal of Nonlinear Science
Vol.13(2012) No.2,pp.142-152

Application of Quintic B-splines Collocation Method on Some Rosenau Type Nonlinear Higher Order Evolution Equations

R.C. Mittal *, R.K. Jain **
Department of Mathematics, Indian Institute of Technology Roorkee, Roorkee 247667, Uttarakhand,India
(Received 27 September 2011, accepted 19 January 2012)

Abstract: In this work, we discuss a collocation method for solving some Rosenau type non-linear higher order evolution equations with Dirichlet's boundary conditions. The approach used, is based on collocation of a quintic B-splines over finite elements so that we have continuity of the dependent variable and its first four derivatives throughout the solution range. We apply quintic. B-splines for spatial variable and derivatives which produce a system of first order ordinary differential equations. We solve this system by using SSP-RK3 scheme. This method needs less storage space that causes to less accumulation of numerical errors. The numerical approximate solutions to Rosenau type non-linear evolution equations have been computed without transforming the equations and without using the linearization. This method is tested on four test problems where one example is with the variable coefficients. Easy and economical implementation is the strength of this method.

Keywords: Rosenau type nonlinear higher order evolution equation; quintic B-splines basis functions; SSP-RK3 scheme; Thomas algorithm

1 Introduction

Non-linear higher order evolution equations are special classes of the category of partial differential equations, which have been studied intensively in past several decades. It is well known that seeking approximate numerical solutions for non-linear higher order evolution equations, by using different numerous methods, has long been a major concern for mathematicians, physicists, and engineers. In particular, the travelling wave solutions play an important role in the study of the models arising from various natural phenomena and scientific and engineering fields; for instance, the wave phenomena observed in fluid dynamics, elastic media, optical fibres, nuclear physics, high-energy physics, plasma physics, gravitation and in statistical and condensed matter physics, biology, solid-state physics, chemical kinematics, chemical physics and geochemistry, etc.[8, 12, 13, 15-17].

In this paper, we consider the mathematical model of Rosenau type non-linear higher order evolution equation of the form

$$u_t + u_{xxxxt} = \phi(u, u_x, u_{xx}), (x,t) \in \Omega \times (0,T] \tag{1.1}$$

where ϕ is some non-linear expression in terms of u, u_x, u_{xx} with boundary conditions

$$u(x,t) = g_0(t), u_x(x,t) = g_1(t), (x,t) \in \partial\Omega \times (0,T] \tag{1.2}$$

and initial coundition

$$u(x,0) = u_0(x), x \in \Omega \tag{1.3}$$

where $\Omega = (a,b)$ and $\mu > 0$.

In literature, we have found different Rosenau type of non-linear higher order evolution equations of the form (1.1), some of them are as follows:

1. Rosenau equation

$$u_t + u_{xxxxt} + u_x + uu_x = 0 \tag{1.4}$$

*__Corresponding author__. *E-mail address*: rcmmmfma@iitr.ernet.in (R.C. Mittal *), rkjain_2000@rediffmail.com (R.K.Jain **)

IJNS.2012.04.15/586

$$u_t + \mu u_{xxxxt} = f(u)_x \tag{1.5}$$

equation of type (1.4) is considered in [9, 15] and type (1.5) in [16].

2. Generalized Rosenau equation

$$u_t + u_{xxxxt} + u_x + (u^p)_x = 0 \tag{1.6}$$

$$u_t + [a(x,t)u_{xxt}]_{xx} = [\psi(u)]_x \tag{1.7}$$

equation of type (1.6) is considered in [7] and type (1.7) in [4, 5].

3. Rosenau-Burger equation

$$u_t + u_{xxxxt} - u_{xx} + u_x + uu_x = 0 \tag{1.8}$$

equation of type (1.8) is considered in [1, 2, 11, 13, 17].

4. Generalized Rosenau-Burgers equation

$$u_t + u_{xxxxt} - \alpha u_{xx} + \beta u_x + (\frac{u^{p+1}}{p+1})_x = 0 \tag{1.9}$$

equation of type (1.9) is considered in [8, 10, 12].

Many algorithms have been developed and simulations performed for (1.1) equation, for example, example finite difference scheme [9, 15] for equation (1.4), discontinuous Galerkin method [16] for equation (1.5), energy conservative finite difference schemes [7] for equation (1.6), finite element Galerkin approximation [4, 5] for (1.7), CrankNicolson finite difference scheme [1], finite difference scheme [2], Fourier transforms method with energy estimates [11] and three-level difference scheme [17] for equation (1.8), Crank-Nicolson difference scheme [10] and Average implicit linear difference scheme [8] for (1.9).

Solution of equation (1.1) is not available analytically in general. So, one has to obtain its numerical solutions to develop an understanding of the non-linear phenomena. There is the different type of equations, which are found in literature of the form (1.1), namely (1.4)-(1.9), in which each equation represents several different physical phenomena. In this paper, we design a collocation method based on a quintic B-splines basis functions for solving some Rosenau type non-linear higher order evolution equations of the form (1.1) with boundary conditions (1.2). We know that B-splines have some special features, which are useful in numerical work. One feature is that the continuity conditions are inherent, other special features of B-splines is that they have small local support, i.e. each B-spline function is only non-zero over a few mesh subintervals, so that the resulting matrix for the discretization equation is tightly banded. Due to their smoothness and capability to handle local phenomena, B-splines offer distinct advantages. In combination with collocation, this significantly simplifies the solution procedure of differential equations. There is a great reduction of the numerical effort, because there is no need to calculate the integrals (like in variational methods) in order to form the final set of algebraic equations, which substitutes the given set of non-linear differential equations. Unlike some previous techniques using various transformations to reduce the equation into more simple equation, the current method does not require extra effort to deal with the non-linear terms. Therefore, the equations are solved easily and elegantly using the present method. This method has also additional advantages over some rival techniques, such as ease in use and computational cost effectiveness in finding solutions of the given non-linear evolution equations. In the present method, the combination of the quintic B-spline collocation method in space with the low-storage third-order total variation diminishing SSP-RK3 scheme in time provides an efficient explicit solution with high accuracy and minimal computational efforts for the problems represented by (1.1)-(1.3).

This paper is organized as follows. In section 2, description of the quintic, B-splines collocation method is explained. In section 3, procedure for implementation of present method for equation (1.1) is described. In section 4, procedure to obtain an initial vector which is required to start our method is explained. We present four numerical examples to establish the adaptability of the proposed method computationally in section 5. Conclusion is given in section 6 that briefly summarizes the numerical outcomes.

2 Description of method

In quintic B-splines collocation method the approximate solution can be written as a linear combination of basis functions which constitute a basis for the approximation space under consideration.

We consider a mesh $a = x_0 < x_1,\ldots,x_{N-1} < x_N = b$ as a uniform partition of the solution domain $a \leq x \leq b$ by the knots x_j with $h = x_j - x_{j-1}, j = 1,\ldots,N$.

Our numerical treatment for solving equation (1.1) using the collocation method with quintic B-spline is to find an approximate solution $U^N(x,t)$ to the exact solution $u(x,t)$ in the form:

$$U^N(x,t) = \sum_{j=-2}^{N+2} c_j(t)B_j(x) \tag{2.1}$$

where $c_j(t)$ are time dependent quantities to be determined from the boundary conditions and collocation from the differential equation.

The quintic B-spline $B_j(x)$ at the knots is given by [14].

$$B_j(x) = \frac{1}{h^5}\begin{cases}(x-x_{j-3})^5, x \in [x_{j-3}, x_{j-2})\\ (x-x_{j-3})^5 - 6(x-x_{j-2})^5, x \in [x_{j-2}, x_{j-1})\\ (x-x_{j-3})^5 - 6(x-x_{j-2})^5 + 15(x-x_{j-1})^5, x \in [x_{j-1}, x_j)\\ (x_{j+3}-x)^5 - 6(x_{j+2}-x)^5 + 15(x_{j+1}-x)^5, x \in [x_j, x_{j+1})\\ (x_{j+3}-x)^5 - 6(x_{j+2}-x)^5, x \in [x_{j+1}, x_{j+2})\\ (x_{j+3}-x)^5, x \in [x_{j+2}, x_{j+3})\\ 0, otherwise\end{cases}$$

where $B_{-2}, B_{-1}, B_0, B_1,\ldots, B_{N-1}, B_N, B_{N+1}, B_{N+2}$ forms a basis over the region of solution domain axb. Each quintic B-spline cover six elements so that each element is covered by six quintic B-splines. The values of $B_j(x)$ and its derivative may be tabulated as in Table- 2.1.

TABLE- 2.1: Coefficient of quintic B-splines and derivatives at nodes x_j

x	x_{j-3}	x_{j-2}	x_{j-1}	x_j	x_{j+1}	x_{j+2}	x_{j+3}
$B_j(x)$	0	1	26	66	26	1	0
$B_j^{'}(x)$	0	$\frac{5}{h}$	$\frac{50}{h}$	0	$\frac{-50}{h}$	$\frac{-5}{h}$	0
$B_j^{''}(x)$	0	$\frac{20}{h^2}$	$\frac{40}{h^2}$	$\frac{-120}{h^2}$	$\frac{40}{h^2}$	$\frac{20}{h^2}$	0
$B_j^{'''}(x)$	0	$\frac{60}{h^3}$	$\frac{-120}{h^3}$	0	$\frac{120}{h^3}$	$\frac{-60}{h^3}$	0
$B_j^{iv}(x)$	0	$\frac{120}{h^4}$	$\frac{-480}{h^4}$	$\frac{720}{h^4}$	$\frac{-480}{h^4}$	$\frac{120}{h^4}$	0

Using approximate function (2.1) and quintic B-spline functions (2.2), the approximate values of $U^N(x,t)$ and its four derivatives at the knots are determined in terms of the time parameters c_j as follows:

$$\begin{aligned} U_j &= c_{j-2} + 26c_{j-1} + 66c_j + 26c_{j+1} + c_{j+2}\\ hU_j^{'} &= 5(c_{j+2} + 10c_{j+1} - 10c_{j-1} - c_{j-2})\\ h^2U_j^{''} &= 20(c_{j-2} + 2c_{j-1} - 6c_j + 2c_{j+1} + c_{j+2})\\ h^3U_j^{'''} &= 60(c_{j+2} - 2c_{j+1} + 2c_{j-1} - c_{j-2})\\ h^4U_j^{iv} &= 120(c_{j-2} - 4c_{j-1} + 6c_j - 4c_{j+1} + c_{j+2}) \end{aligned} \tag{2.2}$$

3 Implementation of method

Our numerical treatment for solving equation (1.1) using the collocation method with quintic B-splines is to find an approximate solution $U^N(x,t)$ to the exact solution $u(x,t)$ is given in (2.1), where $c_j(t)$ are times dependent quantities to be determined from the boundary conditions and collocation from the differential equation. From equation (1.1), we have

$$u_t + \mu u_{xxxxt} = \phi(u, u_x, u_{xx}), \tag{3.1}$$

Now, using (2.1) in (3.1), we get

$$\sum_{j=-2}^{N+2} \dot{c}_j B_j(x) + \mu \sum_{j=-2}^{N+2} \dot{c}_j B_j^{iv}(x) = \phi(\{\sum_{j=-2}^{N+2} c_j B_j(x)\}, \{\sum_{j=-2}^{N+2} c_j B_j^{'}(x)\}, \{\sum_{j=-2}^{N+2} c_j B_j^{''}(x)\}) \tag{3.2}$$

Using approximates function (2.1) and quintic B-splines functions (2.2), the approximate values of $U_t^N(x)$ and its four derivatives at the knots/nodes are determined in terms of the time parameters c_j as follows:

$$\begin{aligned}
(U_j)_t &= \dot{c}_{j-2} + 26\dot{c}_{j-1} + 66\dot{c}_j + 26\dot{c}_{j+1} + \dot{c}_{j+2} \\
h(U_j^{'})_t &= 5(\dot{c}_{j+2} + 10\dot{c}_{j+1} - 10\dot{c}_{j-1} - \dot{c}_{j-2}) \\
h^2(U_j^{''})_t &= 20(\dot{c}_{j-2} + 2\dot{c}_{j-1} - 6\dot{c}_j + 2\dot{c}_{j+1} + \dot{c}_{j+2}) \\
h^3(U_j^{'''})_t &= 60(\dot{c}_{j+2} - 2\dot{c}_{j+1} + 2\dot{c}_{j-1} - \dot{c}_{j-2}) \\
h^4(U_j^{iv})_t &= 120(\dot{c}_{j-2} - 4\dot{c}_{j-1} + 6\dot{c}_j - 4\dot{c}_{j+1} + \dot{c}_{j+2})
\end{aligned} \tag{3.3}$$

Using (2.1),(2.2) and (3.3) in (3.2) we get a system of ordinary differential equations as follows:

$$\begin{aligned}
&(\dot{c}_{j-2} + 26\dot{c}_{j-1} + 66\dot{c}_j + 26\dot{c}_{j+1} + \dot{c}_{j+2}) + \mu\frac{120}{h^4}(\dot{c}_{j-2} - 4\dot{c}_{j-1} + 6\dot{c}_j - 4\dot{c}_{j+1} + \dot{c}_{j+2}) = \\
&\phi(c_{j-2} + 26c_{j-1} + 66c_j + 26c_{j+1} + c_{j+2}, (\frac{5}{h})(c_{j+2} + 10c_{j+1} - 10c_{j-1} - c_{j-2}), \\
&\frac{120}{h^2}(c_{j-2} - 4c_{j-1} + 6c_j - 4c_{j+1} + c_{j+2})), 0 \le j \le N
\end{aligned} \tag{3.4}$$

For simplicity, taking homogeneous boundary conditions in (1.2) and using (2.2) and (3.3) in it, we get

$$\begin{aligned}
\dot{c}_{-2} &= 41.25\dot{c}_0 + 32.5\dot{c}_1 + 2.25\dot{c}_2, \\
\dot{c}_{-1} &= -4.125\dot{c}_0 - 2.25\dot{c}_1 - 0.125\dot{c}_2, \\
\dot{c}_{N+1} &= -4.125\dot{c}_N - 2.25\dot{c}_{N-1} - 0.125\dot{c}_{N-2}, \\
\dot{c}_{N+2} &= 41.25\dot{c}_N + 32.5\dot{c}_{N-1} + 2.25\dot{c}_{N-2}, \\
c_{-2} &= 41.25c_0 + 32.5c_1 + 2.25c_2, \\
c_{-1} &= -4.125c_0 - 2.25c_1 - 0.125c_2, \\
c_{N+1} &= -4.125c_N - 2.25c_{N-1} - 0.125c_{N-2}, \\
c_{N+2} &= 41.25c_N + 32.5c_{N-1} + 2.25c_{N-2}.
\end{aligned} \tag{3.5}$$

We eliminate $\dot{c}_{-2}, \dot{c}_{-1}, \dot{c}_{N+1}, \dot{c}_{N+2}, c_{-2}, c_{-1}, c_{N+1}$ and c_{N+2} in (3.4) by using (3.5) then the system of first order differential equations can be written in the compact form as

$$\mathbf{A}\dot{\mathbf{C}} = \varphi \tag{3.6}$$

where

$$\mathbf{A} = \begin{pmatrix}
a_0 & b_0 & c_0 & & & & & \\
d_1 & a_1 & b_1 & x & & & & \\
x & y & z & y & x & & & \\
 & . & . & . & . & . & & \\
 & & . & . & . & . & . & . \\
 & & & x & y & z & y & x \\
 & & & & x & d_{N-1} & a_{N-1} & b_{N-1} \\
 & & & & & e_N & d_N & a_N
\end{pmatrix}, \dot{\mathbf{C}} = \begin{pmatrix} \dot{c}_0 \\ \dot{c}_1 \\ \dot{c}_2 \\ . \\ . \\ \dot{c}_{N-2} \\ \dot{c}_{N-1} \\ \dot{c}_N \end{pmatrix}, \varphi = \begin{pmatrix} \varphi_0 \\ \varphi_1 \\ \varphi_2 \\ . \\ . \\ \varphi_{N-2} \\ \varphi_{N-1} \\ \varphi_N \end{pmatrix}$$

$$x = 1 + \mu\frac{120}{h^4}, y = 26 - \mu\frac{480}{h^4}, z = 66 + \mu\frac{720}{h^4},$$

$$a_0 = 41.25x - 4.125y + z, b_0 = 32.5x - 1.125y, c_0 = 3.25x - 0.125y,$$

$$d_1 = -4.125x + y, a_1 = -2.25x + z, b_1 = -0.125x + y,$$

$$d_{N-1} = -4.125x + y, a_{N-1} = -2.25x + z, b_{N-1} = -0.125x + y,$$

$$a_N = 41.25x - 4.125y + z, d_N = 32.5x - 1.125y, e_N = 3.25x - 0.125y,$$

$$\varphi_0 = \varphi((\frac{20}{h^2})(27c_0 + 30c_1 + 3c_2)), for j = 0$$

$$\begin{aligned}\varphi_1 = \varphi((&21.875c_0 + 63.75c_1 + 25.875c_2 + c_3),\\ &(\frac{5}{h})(-5.875c_0 + 2.25c_1 + 10.125c_2 + c_3),\\ &(\frac{20}{h^2})(-2.125c_0 - 8.25c_1 + 1.875c_2 + c_3)), for j = 1\end{aligned}$$

$$\begin{aligned}\varphi_j = \varphi((&c_{j-2} + 26c_{j-1} + 66c_j + 26c_{j+1} + c_{j+2}),\\ &(\frac{5}{h})(c_{j+2} + 10c_{j+1} - 10c_{j-1} - c_{j-2}),\\ &(\frac{20}{h^2})(c_{j-2} + 2c_{j-1} - 6c_j + 2c_{j+1} + c_{j+2})), for j = 2 to N - 2\end{aligned}$$

$$\begin{aligned}\varphi_{N-1} = \varphi((&21.875c_N + 63.75c_{N-1} + 25.875c_{N-2} + c_{N-3}),\\ &(\frac{5}{h})(-5.875c_N + 2.25c_{N-1} + 10.125c_{N-2} + c_{N-3}),\\ &(\frac{20}{h^2})(-2.125c_N - 8.25c_{N-1} + 1.875c_{N-2} + c_{N-3})), for j = N - 1\end{aligned}$$

$$\varphi_N = \varphi((\frac{20}{h^2})(27c_N + 30c_{N-1} + 3c_{N-2})), for j = 0$$

Here $\mathbf{A}$ is $(N+1)\times(N+1)$ penta-diagonal matrix, $\dot{\mathbf{C}}$ and φ are $(N+1)$ order vectors, which depend on the boundary conditions. Now, we solve the first order ordinary differential equations system (3.6) by using SSP-RK3 scheme [3]. For computing the RHS of equation (3.6) we need an initial vector $\mathbf{C^0}$ which can be obtained to follow the procedure of section-4.

4 The initial vector $\mathbf{C^0}$

The initial vector $\mathbf{C^0}$ can be obtained from the initial condition and boundary values of the derivatives of the initial condition as the following expressions:

$$\begin{aligned}U_x(x_j, 0) &= U_{xx}(x_j, 0), for j = 0, 1\\ U(x_j, 0) &= u_0(x_j), for j = 2, \cdots N - 2\\ U_x(x_j, 0) &= U_{xx}(x_j, 0), for j = N - 1, N\end{aligned}$$

This yields a $(N+1)\times(N+1)$ system of equations, of the form

$$\mathbf{AC^0} = \mathbf{b} \tag{4.1}$$

where

$$\mathbf{A} = \begin{pmatrix} 54 & 60 & 6 & & & & & \\ 25.75 & 67.5 & 26.25 & 1 & & & & \\ 1 & 26 & 66 & 26 & 1 & & & \\ & \cdot & \cdot & \cdot & \cdot & \cdot & & \\ & & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ & & & 1 & 26 & 66 & 26 & 1 \\ & & & & 1 & 26.25 & 67.5 & 25.25 \\ & & & & & 6 & 60 & 54 \end{pmatrix}, \mathbf{C^0} = \begin{pmatrix} c_0^0 \\ c_1^0 \\ c_2^0 \\ \cdot \\ \cdot \\ c_{N-2}^0 \\ c_{N-1}^0 \\ c_N^0 \end{pmatrix}, \mathbf{b} = \begin{pmatrix} u_0(x_0) \\ u_0(x_1) \\ u_0(x_2) \\ \cdot \\ \cdot \\ u_0(x_{N-2}) \\ u_0(x_{N-1}) \\ u_0(x_N) \end{pmatrix}$$

The solution of (4.1) can be found by Thomas algorithm.

5 Numerical experiments and discussion

In order to show the utility and adaptability of the method, it is tested on the following four test problems. To gain insight into the performance of the present method, the numerical approximation errors L_∞ and L_2 are obtained by following formulae:

$$L_\infty = \max |u_j^{exact} - U_j^{num}| \tag{5.1}$$

$$L_2 = \sqrt{h(\sum_0^N |u_j^{exact} - U_j^{num}|^2)} \tag{5.2}$$

The order of convergence for the numerical method has been computed by the formula

$$order of convergence = \frac{\log\left[\frac{\|u_h - u_{h_i}\|_{L_j}}{\|u_h - u_{h_{i+1}}\|_{L_j}}\right]}{\log\left[\frac{h_i}{h_{i+1}}\right]}, i = 1, 2 and j = 2, \infty \tag{5.3}$$

Example 5.1:

In equation (1.9), we take $\alpha = 1, \beta = 1$ and $p = 1$, then it becomes [17]

$$u_t + u_{xxxxt} - u_{xx} + u_x + uu_x = 0, x \in [0, 1], t \in [0, T] \tag{5.1.1}$$

with the boundary conditions

$$u(0, t) = u(1, t) = 0 and u_x(0, t) = u_x(1, t) = 0, t \in [0, T] \tag{5.1.2}$$

and intial condition

$$u(x, 0) = x^4(1 - x)^4, x \in [0, 1]. \tag{5.1.3}$$

Since the exact solution of $(5.1.1) - (5.1.3)$ is not known, we consider the solution U_{ref}^N which is computed by the present method on fine mesh $h = \frac{1}{160}$ and $k = 0.05$ as a reference solution at $T = 1$. Error estimates and order of convergence for different step size at $T = 1.0$ are computed using $(5.1) - (5.3)$ and reported in Table-5.1.1. The graph depicted in Figure-1, shows the same characteristics as shown by [17].

TABLE-5.1.1: Error Estimates and order of convergence at $T = 1.0$

N	L_∞	$order$	L_2	$order$
10	$7.03E-04$	$-$	$4.58E-04$	$-$
20	$2.04E-04$	1.7865	$1.34E-04$	1.7787
40	$5.23E-05$	1.9613	$3.42E-05$	1.9652
80	$1.08E-05$	2.2746	$7.08E-06$	2.2739

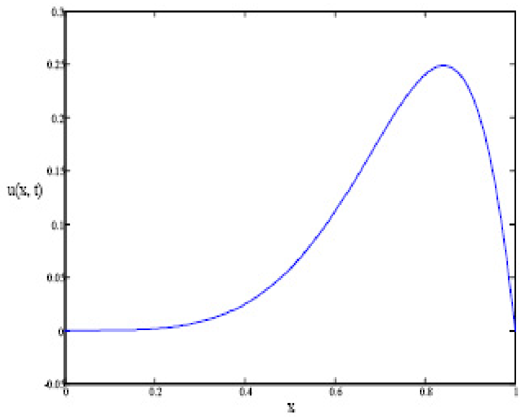

Figure 1: Approximate solution at T = 1.0.

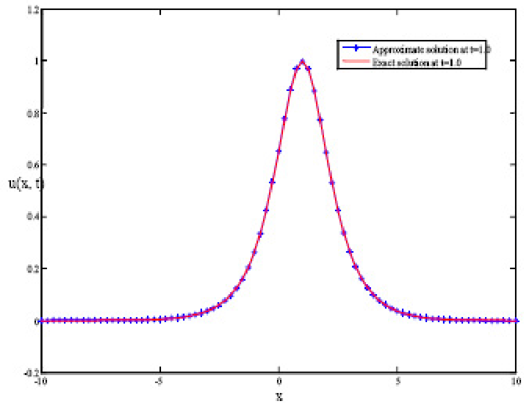

Figure 2: Approximate and exact solution at $T = 1.0 (h = 0.25, k = 0.1)$.

Example 5.2:

We consider the following generalized Rosenau equation given in [16]

$$2u_t + u_{xxxxt} + 3u_x - 60u^2u_x + 120u^4u_x = 0, x \in [-10, 10], t \in [0, T] \tag{5.2.1}$$

with the boundary conditions

$$\begin{aligned} u(-10, t) &= sech(-10 - t), \\ u(10, t) &= sech(10 - t), \\ u_x(-10, t) &= -sech(-10 - t)tanh(-10 - t), \\ u_x(10, t) &= sech(10 - t)tanh(10 - t), t \in [0, t], \end{aligned} \tag{5.2.2}$$

and intial condition

$$u(x, 0) = sech(x), x \in [-10, 10]. \tag{5.2.3}$$

The exact solution for the Rosenau equation $(5.2.1)$ is known to be $u(x, t) = sech(x - t)$.
The equation $(5.2.1)$ can be rewritten as

$$u_t + 0.5u_{xxxxt} = f(u)_x \tag{5.2.4}$$

where $f(u) = 10u^3 - 12u^5 - 1.5u$. For numerical computation, we take $h = \frac{1}{4}, \frac{1}{5}, \frac{1}{6}$ and $\frac{1}{7}$ with $k = \frac{1}{20}$. The error estimates and order of convergence are computed using $(5.1) - (5.3)$ at $T = 0.2$ and reported in Table- $5.2.1$. Graph of exact and approximate solution depicted in Fig.- 2 at $T = 1.0$, which shows the same characteristics as shown by S.M. Choo et. al [17].

TABLE-5.2.1 : Error Estimates and order of convergence at $T = 0.2$

N	L_∞	$order$	L_2	$order$
80	$1.49E-03$	$-$	$1.53E-03$	$-$
100	$9.84E-04$	1.864625	$1.01E-03$	1.891216
120	$6.91E-04$	1.938629	$7.38E-04$	1.697033
140	$5.00E-04$	2.105758	$6.04E-04$	1.306621

Example 5.3:

We consider the following generalized Rosenau equation given in [5]

$$u_t + \{a(x, t)u_{xxt}\}_{xx} = \{\psi(u)\}_x \tag{5.3.1}$$

Case 1: We take $a(x, t) = 1.0$ and $\psi(u) = u$, with initial condition $u_0(x) = x^2(1 - x)^2$and boundary conditions (1.2), where $\Omega = (0, 1)$.

Equation (5.3.1) takes form

$$u_t + u_{xxxxt} = u_x \tag{5.3.2}$$

Since we dont have the exact solution to $(5.3.2)$, we consider the solution on mesh $h = \frac{1}{160}$ as the reference solution and obtain the error estimates by comparing the numerical solutions on mesh $h = 0.1, h = 0.05, h = 0.025$ and $h = 0.0125$ with those on mesh $h = \frac{1}{160}$ with $\Delta t = 0.05$ respectively. The error estimates and order of convergence are computed from $(5.1) - (5.3)$ and reported in Table- 5.3.1 at $T = 1.0$. The graph of approximate solution at $T = 0.0$ and 1.0 is depicted in Fig.- 3.

TABLE-5.3.1 : Error Estimates and order of convergence at $T = 1.0$

N	L_∞	$order$	L_2	$order$
10	$2.26E-05$	$-$	$1.44E-05$	$-$
20	$2.13E-06$	3.4086	$1.28E-06$	3.4986
40	$2.13E-07$	3.3255	$1.17E-07$	3.4458
80	$2.53E-08$	3.0698	$1.44E-08$	3.0271

Case 2: We take $a(x,t) = 1.0$ and $\psi(u) = u^3$, with initial condition $u_0(x) = x^2(1-x)^2$and boundary conditions (1.2), where $\Omega = (0,1)$.

Equation (5.3.1) takes form

$$u_t + u_{xxxxt} = 3u^2 u_x \tag{5.3.3}$$

Since we dont have the exact solution to $(5.3.3)$, we consider the solution on mesh $h = \frac{1}{160}$ as the reference solution. We obtain the error estimates and order of convergence using $(5.1) - (5.3)$. We compare the numerical solutions on mesh $h = 0.1, h = 0.05, h = 0.025$ and $h = 0.0125$ with those on mesh $h = \frac{1}{160}$ with $\Delta t = 0.05$ respectively. The error estimates and order of convergence are reported in Table- 5.3.2 at $T = 1.0$.

TABLE-5.3.2 : Error Estimates and order of convergence at $T = 1.0$

N	L_∞	$order$	L_2	$order$
10	$1.45E-08$	$-$	$7.91E-09$	$-$
20	$2.63E-09$	2.4616	$1.68E-09$	2.2330
40	$6.34E-10$	2.0533	$4.01E-10$	2.0679
80	$1.27E-10$	2.3204	$8.04E-11$	2.3203

Case 3: We take $a(x,t) = (x^2+5)e^{-t}$ and $\psi(u) = u$, with initial condition $u_0(x) = x^2(1-x)^2$and boundary conditions (1.2), where $\Omega = (0,1)$.

Equation (5.3.1) takes form

$$u_t + 2e^{-t}u_{xxt} + 4xe^{-t}u_{xxxt} + (x^2+5)e^{-t}u_{xxxxt} = u_x \tag{5.3.4}$$

Since we dont have the exact solution to $(5.3.4)$, we consider the solution on mesh $h = \frac{1}{160}$ as the reference solution. We obtain the error estimates and order of convergence using $(5.1) - (5.3)$. We compare the numerical solutions on mesh $h = 0.1, h = 0.05, h = 0.025$ and $h = 0.0125$ with those on mesh $h = \frac{1}{160}$ with $\Delta t = 0.05$ respectively. The error estimates and order of convergence are reported in Table- 5.3.3 at $T = 1.0$.

TABLE-5.3.3 : Error Estimates and order of convergence at $T = 1.0$

N	L_∞	$order$	L_2	$order$
10	$6.87E-06$	$-$	$4.35E-06$	$-$
20	$6.73E-07$	3.3509	$4.05E-07$	3.4283
40	$7.35E-08$	3.1959	$4.11E-08$	3.3002
80	$9.63E-09$	2.9315	$5.29E-09$	2.9568

Case 4: We take $a(x,t) = (x^2+5)e^{-t}$ and $\psi(u) = u^3$, with initial condition $u_0(x) = x^2(1-x)^2$and boundary conditions (1.2), where $\Omega = (0,1)$.

Equation (5.3.1) takes form

$$u_t + 2e^{-t}u_{xxt} + 4xe^{-t}u_{xxxt} + (x^2+5)e^{-t}u_{xxxxt} = 3u^2 u_x \tag{5.3.5}$$

Since we dont have the exact solution to $(5.3.4)$, we consider the solution on mesh $h = \frac{1}{160}$ as the reference solution. We obtain the error estimates and order of convergence using $(5.1) - (5.3)$. We compare the numerical solutions on mesh $h = 0.1, h = 0.05, h = 0.025$ and $h = 0.0125$ with those on mesh $h = \frac{1}{160}$ with $\Delta t = 0.05$ respectively. The error estimates and order of convergence are reported in Table- $5.3.4$ at $T = 1.0$. The graph of approximate solution at $T = 0$ and $T = 1$ are depicted in Fig.- 4.

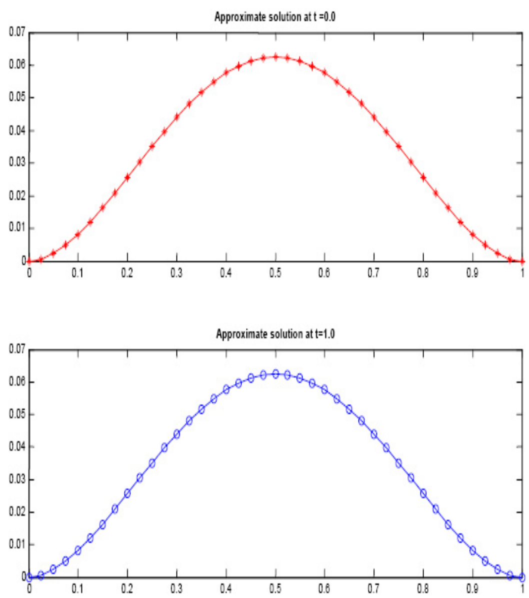

Figure 3: Approximate solution at $T = 0.0$ and $1.0(h = 0.025, k = 0.05)$.

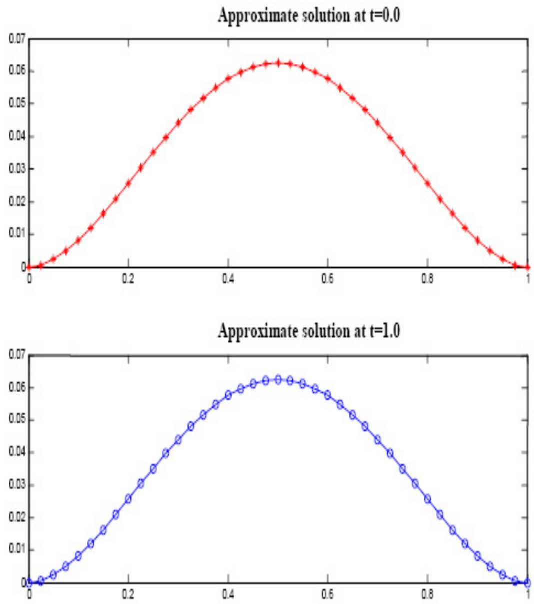

Figure 4: Approximate solution at $T = 0.0$ and $1.0(h = 0.025, k = 0.05)$.

TABLE-5.3.4 : Error Estimates and order of convergence at $T = 1.0$

N	L_∞	$order$	L_2	$order$
10	$4.83E-09$	$-$	$2.61E-09$	$-$
20	$9.53E-10$	2.3412	$5.45E-10$	2.2619
40	$2.27E-10$	2.0723	$1.30E-10$	2.0694
80	$4.56E-11$	2.3143	$2.60E-11$	2.3203

Example 5.4:

(a) In equation (1.9), we take $\alpha = 0.1, \beta = 1$ and $p = 2$, then it becomes

$$u_t + u_{xxxxt} - 0.1u_{xx} + u_x + u^2u_x = 0, x \in [0, 1], t \in [0, T] \tag{5.4.1}$$

with the boundary conditions

$$u(0, t) = u(1, t) = 0, u_x(0, t) = u_x(1, t) = 0, t \in [0, T] \tag{5.4.2}$$

and initial condition

$$u(x, 0) = x^4(1 - x)^4, x \in [0, 1] \tag{5.4.3}$$

Since we dont have the exact solution to $(5.4.1)$, we consider the solution on mesh $h = \frac{1}{160}$ as the reference solution. We obtain the error estimates and order of convergence using $(5.1) - (5.3)$. We compare the numerical solutions on mesh $h = 0.1, h = 0.05, h = 0.025$ and $h = 0.0125$ with those on mesh $h = \frac{1}{160}$ with $\Delta t = 0.1$ respectively. The error estimates and order of convergence are reported in Table- $5.4.1$ at $T = 1.0$.

TABLE-5.4.1 : Error Estimates and order of convergence at $T = 1.0$

N	L_∞	$order$	L_2	$order$
10	$2.18E-05$	$-$	$1.48E-05$	$-$
20	$1.03E-05$	1.0827	$6.98E-06$	1.0900
40	$3.06E-06$	1.7513	$2.06E-06$	1.7573
80	$6.73E-07$	2.1826	$4.54E-07$	2.1847

(b) In equation (1.9), we take $\alpha = 0.5, \beta = 1$ and $p = 5$, then it becomes

$$u_t + u_{xxxxt} - 0.5u_{xx} + u_x + u^5 u_x = 0, x \in [0, 1], t \in [0, T] \quad (5.4.4)$$

with the boundary conditions $(5.5.2)$ and initial condition $(5.4.3)$. Since we dont have the exact solution to $(5.4.4)$, we consider the solution on mesh $h = \frac{1}{160}$ as the reference solution. We obtain the error estimates and order of convergence using $(5.1) - (5.3)$. We compare the numerical solutions on mesh $h = 0.1, h = 0.05, h = 0.025$ and $h = 0.0125$ with those on mesh $h = \frac{1}{160}$ with $\Delta t = 0.1$ respectively. The error estimates and order of convergence are reported in Table-5.4.2 at $T = 1.0$.

TABLE-5.4.2 : Error Estimates and order of convergence at $T = 1.0$

N	L_∞	$order$	L_2	$order$
10	$3.30E-04$	$-$	$2.16E-04$	$-$
20	$9.72E-05$	1.7644	$6.38E-05$	1.7567
40	$2.52E-05$	1.9485	$1.65E-05$	1.9521
80	$5.23E-06$	2.2682	$3.43E-06$	2.2675

(c) In equation (1.9), we take $\alpha = 1, \beta = 1$ and $p = 8$, then it becomes

$$u_t + u_{xxxxt} - u_{xx} + u_x + u^8 u_x = 0, x \in [0, 1], t \in [0, T] \quad (5.4.5)$$

with the boundary conditions $(5.5.2)$ and initial condition $(5.4.3)$. Since we dont have the exact solution to $(5.4.4)$, we consider the solution on mesh $h = \frac{1}{160}$ as the reference solution. We obtain the error estimates and order of convergence using $(5.1) - (5.3)$. We compare the numerical solutions on mesh $h = 0.1, h = 0.05, h = 0.025$ and $h = 0.0125$ with those on mesh $h = \frac{1}{160}$ with $\Delta t = 0.1$ respectively. The error estimates and order of convergence are reported in Table-5.4.3 at $T = 1.0$.

TABLE-5.4.3 : Error Estimates and order of convergence at $T = 1.0$

N	L_∞	$order$	L_2	$order$
10	$7.11E-04$	$-$	$4.64E-04$	$-$
20	$2.05E-04$	1.7959	$1.34E-04$	1.7886
40	$5.26E-05$	1.9622	$3.44E-05$	1.9662
80	$1.09E-05$	2.2747	$7.10E-06$	2.2739

6 Conclusions

In this work, we have developed a collocation method for solving some Roseneu type non-linear higher order evolution equation with Dirichlet's boundary conditions using quintic B-splines basis functions. In the present method, we apply quintic B-splines for spatial variable and derivatives, which produce a system of first order ordinary differential equations. The resulting systems of ordinary differential equations are solved by using SSP-RK3 scheme. The numerical approximate solutions to Roseneu type non-linear equations have been computed without transforming the equation and without using the linearization. This method is tested on four problems, and the numerical results obtained are quite satisfactory and comparable with the existing solutions found in literature. Easy and economical implementation is the strength of this method. The method is capable of solving problems with variable coefficients. The computed results justify the advantage of this method.

Acknowledgments

One of the authors R.K. Jain thankfully acknowledges the sponsorship under QIP, provided by Technical Education and Training Department, Bhopal (M.P.), India.

References

[1] Bing Hu, Youcai Xu and Jinsong Hu. CrankNicolson finite difference scheme for the RosenauBurgers equation. *Applied Mathematics and Computation*, 204 (2008): 311-316.

[2] Zhuangzhuang Wang, Yongbing Wu and Yao Lu. A finite difference simulation for Rosenau-Burgers Equation. 978-1-4244-4994-1/09/25.00 2009 IEEE.

[3] Graslan G. and Sari M. Numerical solutions of linear and nonlinear diffusion equations by a differential quadrature method (DQM). *Int. J. Numer. Meth. in Biomed. Engng*, 27 (2011): 69-77.

[4] H. Y. Lee, M. R. Ohm and J. Y. Shin. Finite Element Galerkin Approximations of the Rosenau Equation. *Proceedings of Nonlinear Functional Analysis and Applications*, 2 (1997): 153-160.

[5] H. Y. Lee, M. R. Ohm and J. Y. Shin. Convergence of Fully Discrete Galerkin Approximations of the Rosenau Equation. *Korean J. Comput. and Appl. Math.*,6(1)(1999): 1-13.

[6] Huabing Jia and Wei Xu. Solitons solutions for some nonlinear evolution equations. *Applied Mathematics and Computation*, 217 (2010): 1678-1687.

[7] Jinsong Hu and Kelong Zheng. Two Conservative Difference Schemes for the Generalized Rosenau Equation. Hindawi Publishing Corporation, Boundary Value Problems Volume 2010, Article ID 543503, doi:10.1155/2010/543503.

[8] Jinsong Hu, Bing Hu and Youcai Xu. Average implicit linear difference scheme for generalized RosenauBurgers equation. *Applied Mathematics and Computation*, 217 (2011): 7557-7563.

[9] Khaled Omrani, Faycal Abidi, Talha Achouri and Noomen Khiari: A new conservative finite difference scheme for the Rosenau equation, Applied Mathematics and Computation 201 (2008): 35-43.

[10] Kelong Zheng and Jinsong Hu. Crank-Nicolson difference scheme for the generalized Rosenau-Burgers equation. *International Journal of Mathematical and Computer Sciences*, 5:4 (2009).

[11] Liping Liu and Ming Mei. A better asymptotic profile of RosenauBurgers equation. *Applied Mathematics and Computation* 131 (2002): 147-170.

[12] MI AI PARK. Point wise Decay Estimates of Solutions of the Generalized Rosenau Equation. *J. Korean Math. Soc.*, 29 (1992)(2): 261-280.

[13] Ming Mei. Long-Time Behavior of Solution for Rosenau-Burgers Equation (I). *Applicable Analysis*, 63(3) 1996: 315-330.

[14] R.C. Mittal and Geeta Arora. Quintic B-Spline Collocation Method for Numerical Solution of the Extended Fisher-Kolmogrov Equation. *Int. J. of Appl. Math and Mech.*, 6 2010(1): 74-85 .

[15] S. K. Chung. Finite Difference Approximate Solutions for the Rosenau Equation. *Applicable Analysis*, 69(1-2): 149-156.

[16] S.M. Choo, S.K. Chung, and K.I. Kim. A discontinuous Galerkin method for the Rosenau equation. *Applied Numerical Mathematics*, 58 (2008): 783-799.

[17] Weiyuan Ma, Aili Yang and Yang Wang. A Second-Order Accurate Linearized Difference Scheme for the Rosenau-Burgers Equation. *Journal of Information and Computational Science*, 7(8) (2010): 1793-1800.