University Of AL-Qadisiyah

Available online at www.qu.edu.iq/journalcm

JOURNAL OF AL-QADISIYAH FOR COMPUTER SCIENCE AND MATHEMATICS

ISSN:2521-3504(online) ISSN:2074-0204(print)

Combining Cubic B-Spline Galerkin Method with Quadratic Weight Function for Solving Partial Integro-Differential Equations

Hameeda O.Al-Humedi [a] **,** ***Zahraa Adnan jameel*** [b]

[a] *Department of Mathematics, College of Education for Pure Sciences, Basrah University, Basrah, Iraq. Email: ahameeda722@yahoo.com*

[b] *Department of Mathematics, College of Education for Pure Sciences, Basrah University, Basrah, Iraq. Email: zahraad788@yahoo.com*

ARTICLE INFO

Article history:
Received: 18/12 /2019
Rrevised form: 07/1/2020
Accepted : 28/1/2020
Available online: 17/02 / 2020

Keywords:

B-spline method, Galerkin method, integro-differential equation, Von-Neuman.

ABSTRACT

In this article, a numerical scheme was implemented for solving the partial integro-differential equations (PIDEs) with weakly singular kernel by using the cubic B-spline Galerkin method with quadratic B-spline as a weight function. backward Euler scheme was used for time direction and the cubic B-spline Galerkin method with quadratic weight function was used for spatial derivative. We observed from the numerical examples that the proposed method possesses a high degree of efficiency and accuracy. In addition, the numerical results are in suitable agreement with the exact solutions via calculating L_2 and L_∞ norms errors. Theoretically, we discussed the stable evaluation of the current method using the Von-Neumann method, which explained that the present technique is unconditionally stable.

MSC :

DOI : 10.29304/jqcm.2020.12.1.660

1 . Introduction

Many mathematical formulations of physical phenomena contain PIDEs, which can describe some physical situations such as viscoelasticity, convection- diffusion problems, heat flow in materials with memory, nuclear reactor dynamics, geophysics and plasma physics etc.

Consider the following PIDE with a weakly singular kernel is

$$u_t(x,t) + m\,u_x(x,t) - b\,u_{xx}(x,t) = \int_0^t K(t-s)\,u(x,s)ds + f(x,t) \qquad x \in [a,b], t > 0 \qquad (1)$$

where, $K(t-s) = (t-s)^{-\alpha}$, $\quad 0 < \alpha < 1$

subject to the initial condition are :

Corresponding author Zahraa Adnan jameel

Email addresses: zahraad788@yahoo.com

Communicated by Alaa H. H. Al-Ka'bi

$$u(x,0) = g_0(x), \qquad a \le x \le b \tag{2}$$

and the boundary conditions are :

$$u(a,t) = f_0(t), \ \ u(b,t) = f_1(t), \quad t \ge 0 \tag{3}$$

where $g_0(x), f_0(t), f_1(t)$ are known functions and $f(x,t)$ is a smooth function.

The integral-differential equations (1) - (3) are primary importance in many physical systems especially those involving fluid flow [4, 16].

T. Tang [18] and A.F. Soliman et al. [15] used finite difference scheme for PIDEs with a weakly singular kernel and PIDEs respectively. A.F. Soliman et al [14] applied the numerical solution for solving PIDEs with sixth-degree B-spline functions. H. Zhang et al [20] used quintic B-spline collocation method for finding a solution to fourth order PIDEs with a weakly singular kernel. S.S. Siddiqi and S. Arshed [12] utilized cubic B-spline collocation method to solve a convection-diffusion integro-differential equation with a weakly singular kernel. M. Gholamian and J. Saberi-Nadjafi [5] and M. Gholamian et al. [6] used cubic B-splines collocation method for solving class of PIDEs and the second order PIDEs with a weakly singular kernel respectively. R.C. Mittal and R.K. Jain [9] studied a collocation method based on redefined cubic B-splines basis functions for solving convection–diffusion equation with Dirichlets type boundary conditions. A. Ali et al [2] solved PIDEs with a weakly singular kernel by using a quartic B-spline collocation method. S. S. Siddiqi and S. Arshed [13] solved the PIDEs by applying cubic B-spline collocation method. H. Zhang and X. Han [21] used a quasi-wavelet method to find a solution of time dependent fractional partial differential equation. F.I. Haq et al [7] got the numerical solution of modified regularized long wave (MRLW) equation using quartic B-splines. H.O. Al-Humedi and A. Abd Al-wahed [3] resolved modified equal width (MEW) by stratifying B-spline Galerkin methods with a change weight function. A. Abdul Wahid [1] employed sextic B-spline Galerkin scheme with quintic weight function for solving Burgers equation.

In this paper, we will solve PIDEs by using a new numerical solution depending on the combination of the cubic B-spline Galerkin method and a quadratic weight function.

2. Cubic B-Spline with a Quadratic Weight Function

The cubic B-spline $C_m(x), (m = -1(1)N+1)$, at the knots x_m which form a basis over the solution domain $[a,b]$, is defined as [11]

$$C_m(x) = \frac{1}{h^3}\begin{cases} (x - x_{m-2})^3, & \text{if } x \in [x_{m-2}, x_{m-1}], \\ h^3 + 3h^2(x - x_{m-1}) + 3h(x - x_{m-1})^2 - 3(x - x_{m-1})^3, & \text{if } x \in [x_{m-1}, x_m], \\ h^3 + 3h^2(x_{m+1} - x) + 3h(x_{m+1} - x)^2 - 3(x_{m+1} - x)^3, & \text{if } x \in [x_m, x_{m+1}], \\ (x_{m+2} - x)^3, & \text{if } x \in [x_{m+1}, x_{m+2}], \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

The set of splines $(C_{-1}(x), C_0(x), \dots, C_N(x),\ C_{N+1}(x))$ forms a basis for functions defined over $[a,b]$. Consider the approximate solution $U_N(x,t)$ to the exact solution $U(x,t)$ is given by

$$U_N(x,t) = \sum_{i=-1}^{N+1} C_i(x)\sigma_i(t), \tag{5}$$

where σ_i are unknown time-dependent parameters to be determined from the boundary and weighted residual conditions. We will use the following local coordinate transformation

$$h\eta = x - x_m, \quad 0 \le \eta \le 1, \tag{6}$$

a cubic B-spline shape functions in terms of η over the element $[x_m, x_{m+1}]$ that can be defined as

$$\begin{cases} C_{m-1} = (1-\eta)^3, \\ C_m = 1 + 3(1-\eta) + 3(1-\eta)^2 - 3(1-\eta)^3, \\ C_{m+1} = 1 + 3\eta + 3\eta^2 - 3\eta^3, \\ C_{m+2} = \eta^3, \end{cases} \tag{7}$$

All splines apart from C_{m-1}, C_m, C_{m+1} and C_{m+2} are zero over the element $[x_m, x_{m+1}]$.The Variation of the function $U(\eta, t)$ over element $[x_m, x_{m+1}]$ is approximated by

$$U_N(\eta, t) = \sum_{i=m-1}^{m+2} C_i(\eta)\sigma_i(t), \tag{8}$$

where $\sigma_{m-1}(t), \sigma_m(t), \sigma_{m+1}$ and $\sigma_{m+2}(t)$ act as element parameters and B-splines $C_{m-1}(\eta), C_m(\eta), C_{m+1}(\eta)$ and $C_{m+2}(\eta)$ as element shape functions. The spline $C_m(x)$ vanishes outside the interval $[x_{m-2}, x_{m+2}]$. So, the value of U with its first and second derivatives U', U'' respectively at the knots, x_m which is determined in terms of element parameters σ_m by

$$\left.\begin{aligned} U_m = U(x_m) = \sigma_{m-1} + 4\sigma_m + \sigma_{m+1} \\ U'_m = U'(x_m) = \tfrac{3}{h}(\sigma_{m-1} - \sigma_{m+1}) \\ U''_m = U''(x_m) = \tfrac{6}{h^2}(\sigma_{m-1} - 2\sigma_m + \sigma_{m+1}) \end{aligned}\right\} \tag{9}$$

take the weight function, W quadratic B-spline that is defined as

$$B_m(x) = \frac{1}{h^2}\begin{cases} (x_{m+2} - x)^2 - 3(x_{m+1} - x)^2 + 3(x_m - x)^2, & \text{if } x \in [x_{m-1}, x_m], \\ (x_{m+2} - x)^2 - 3(x_{m+1} - x)^2, & \text{if } x \in [x_m, x_{m+1}], \\ (x_{m+2} - x)^2, & \text{if } x \in [x_{m+1}, x_{m+2}], \\ 0, & \text{otherwise}. \end{cases} \tag{10}$$

by applying Galerkin method on equation (1) with a weight function W, we get

$$\int_a^b W\,[\,u_t + m\,u_x - b\,u_{xx} - \int_0^t K(t-s)\,u(x,s)ds - f(x,t)\,]\;dx = 0 \tag{11}$$

$$\int_0^1 W\,u_t\,d\eta + \frac{m}{h}\int_0^1 W\,u_\eta\,d\eta - \frac{b}{h^2}\int_0^1 W\,u_{\eta\eta}\,d\eta - \int_0^1[\int_0^t W\,(t-s)^{-\alpha}u(\eta,s)ds - Wf(\eta,t)]\,d\eta = 0, \tag{12}$$

where , $\frac{m}{h} = \lambda$ and $\frac{b}{h^2} = \beta$.

By taking weight function quadratic B-spline and applying weak form we get:

$$\sum_{i=m-1}^{m+2} \int_0^1 C_i B_j d\eta \ \sigma_i' + \sum_{i=m-1}^{m+2} [\int_0^1 (\lambda\, C_i' B_j \ + \beta C_i' \ B_j')\, d\eta - \ \beta \ C_i' \ B_j(\eta) \ |_0^1 \ \ ]\, \sigma_i - \sum_{i=m-1}^{m+2} \int_0^t (t-s)^{-\propto} \int_0^1 C_i B_j d\eta \ \ \sigma_i(s) \ ds - \int_0^1 f(\eta,t)\, B_j(\eta) d\eta = 0\,, \qquad (13)$$

write the equation (13) as in following matrix form:

$$X_{ij}^e (\sigma')^e + [\ \lambda\, Q_{ij}^e + \ \beta\ (Y_{ij}^e - \ Z_{ij}^e)]\ \sigma^e + \ \int_0^t (t-s)^{-\propto} X_{ij}^e \ \sigma^e(s) ds - \ F_j = 0, \qquad (14)$$

where $\sigma^e = (\sigma_{m-1}\,, \sigma_m\,, \sigma_{m+1}\,, \sigma_{m+2})$ are the element parameters which can be gotten with the element matrices $X_{ij}^e,\ Y_{ij}^e,\ Z_{ij}^e$ and Q_{ij}^e are rectangular 3×4 write by the following integrals:

$$X_{ij}^e = \int_0^1 C_i\ B_j\ d\eta = \frac{1}{60}\begin{bmatrix} 10 & 71 & 38 & 1 \\ 19 & 221 & 221 & 19 \\ 1 & 38 & 71 & 10 \end{bmatrix}, \qquad Y_{ij}^e = \int_0^1 C_i'\ B_j'\ d\eta = \frac{1}{2}\begin{bmatrix} 3 & 5 & -7 & -1 \\ -2 & 2 & 2 & -2 \\ -1 & -7 & 5 & 3 \end{bmatrix},$$

$$Q_{ij}^e = \int_0^1 C_i' B_j\ d\eta = \frac{1}{10}\begin{bmatrix} -6 & -7 & 12 & 1 \\ -13 & -41 & 41 & 13 \\ -1 & -12 & 7 & 6 \end{bmatrix}, \qquad Z_{ij}^e = C_i'\ B_j\ |_0^1 \ = 3\begin{bmatrix} 1 & 0 & -1 & 0 \\ 1 & -1 & -1 & 1 \\ 0 & -1 & 0 & 1 \end{bmatrix}.$$

Where i and j take the value m-1, m, m+1, m+2 from element $[\mathrm{x_m}, \mathrm{x_{m+1}}]$.
Assembling all contribution from all element yields the global system of equations.
Compensate the time point $t = t_{n+1}$, get

$$X_{ij}\sigma'(t_{n+1}) + [\ \lambda\, Q_{ij} + \ \beta\ (Y_{ij} - \ Z_{ij})]\ \sigma(t_{n+1}) + \ \int_0^{t_{n+1}} (t_{n+1}-s)^{-\propto} X_{ij}\ \sigma(s) ds - \ F_j = 0, \qquad (15)$$

where $\sigma = (\sigma_0, \sigma_1, \dots, \sigma_N)^{\mathrm{T}}$ is a global element.
Substituting the time derivatives σ' by backward finite difference (15) get:

$$\sigma'(t_{n+1}) = \frac{\sigma(t_{n+1}) - \sigma(t_n)}{\Delta t}$$

The integral in above equation can be calculated as [12]

$$\begin{aligned} X_{ij} \int_0^{t_{n+1}} (t_{n+1}-s)^{-\propto} \sigma(s) ds \ &= \ X_{ij} \int_0^{t_{n+1}} s^{-\propto} \ \sigma(t_{n+1}-s) ds \\ &= X_{ij} \sum_{k=0}^{n} \ \sigma(t_{n-k+1}) \int_{t_k}^{t_{k+1}} s^{-\propto}\ ds \\ &= \frac{\Delta t^{1-\propto}}{1-\propto} X_{ij} \sum_{k=0}^{n} \ \sigma(t_{n-k+1})\ [(k+1)^{1-\propto} - (\ k)^{1-\propto}], \end{aligned} \qquad (16)$$

the matrix form (15) can be written as:

$$X_{ij} \frac{\sigma(t_{n+1}) - \sigma(t_n)}{\Delta t} + [\ \lambda\, Q_{ij} + \ \beta\big(\ Y_{ij} - \ Z_{ij}\big)]\ \sigma(t_{n+1}) + \ \frac{\Delta t^{1-\propto}}{1-\propto} X_{ij} \sum_{k=0}^{n} \ \sigma(t_{n-k+1})\ [(k+1)^{1-\propto} - (k)^{1-\propto}] - F_j = 0 \qquad (17)$$

Write $\sigma(t_{n+1}) = \ \sigma^{n+1}\,, \sigma\ (t_n) = \ \sigma^n$ and $\sigma(t_{n-k+1}) = \sigma^{n-k+1}$, then

$$\big[X_{ij} + \Delta \mathrm{t}\ \{\ \lambda\, Q_{ij} + \ \beta\big(\ Y_{ij} - \ Z_{ij}\big)\}\big]\ \sigma^{n+1} = X_{ij}\ \sigma^n + \ \frac{\Delta \mathrm{t}^{2-\propto}}{1-\propto} X_{ij} \sum_{k=0}^{n} \ \sigma^{n-k+1}\ [(k+1)^{1-\propto} - (\ k)^{1-\propto}] + \Delta \mathrm{t}\ F_j \qquad (18)$$

or,

$$\left[X_{ij} + \Delta t\left\{\lambda\, Q_{ij} + \beta\left(Y_{ij} - Z_{ij}\right)\right\} - \frac{\Delta t^{2-\propto}}{1-\propto}\, X_{ij}\right]\ \sigma^{n+1} = X_{ij}\,\sigma^{n} + \frac{\Delta t^{2-\propto}}{1-\propto} X_{ij} \sum_{k=1}^{n} \sigma^{n-k+1}\, b_k + \Delta t\, F_j\ , \qquad (19)$$

where,

$$b_k = [(k+1)^{1-\propto} - (\,k)^{1-\propto}\,] \qquad k = 1,2,3, \dots ..$$

$$F_j = \int_0^1 f(\eta, t_{n+1})\, B_j(\eta) d\eta$$

The system (19) is of (N+1) linear equation with (N+3) unknowns.

Applying the initial condition $u(x,0) = g_0(x)$ to the equation (15) makes the matrix equation square, computing the initial vector $\sigma^0 = [\sigma_0^0, \sigma_1^0, \sigma_2^0, \dots., \sigma_N^0\,]^T$ from the initial condition $u(x,0) = g_0(x)$ given (N+1) equation in (N+3) unknowns, to determine these unknown, the following relations at the knots are used

$U_x(0{,}0) = u_x(x_0, 0)$,

$U(x_i, 0) = g_0(x_i), \qquad i = 1(1)(N-1)$

$U_x(L, 0) = u_x(x_N, 0)$.

We have the tridiagonal system of equation that can be solved by: $\quad R\,\sigma^0 = E$

where,

$$R = \begin{bmatrix} 4 & 2 & 0 & & & & & & \\ 1 & 4 & 1 & & & & & & \\ & 1 & 4 & 1 & & & & & \\ & & & \ddots & \ddots & \ddots & & & \\ & & & & & & 1 & 4 & 1 \\ & & & & & & 0 & 2 & 4 \end{bmatrix},$$

at first reducing it to tridiagonal matrix by eliminating the first equation and then applying Thomas algorithm [10].

3. stability

To study the stability of the proposed method, we will apply the Von-Neumann method [8], of equation (19) and set $f(\eta, t) = 0$ [19] for any, $0 \le \eta \le 1$ to have:

$$(q_1 - \varrho_1)\sigma_{m-2}^{n+1} + (q_2 - \varrho_2)\sigma_{m-1}^{n+1} + (q_3 - \varrho_3)\,\sigma_m^{n+1} + (q_4 - \varrho_3)\,\sigma_{m+1}^{n+1} + (q_5 - \varrho_2)\sigma_{m+2}^{n+1} + (q_6 - \varrho_1)\sigma_{m+3}^{n+1} = q_7\sigma_{m-2}^{n} + q_8\,\sigma_{m-1}^{n} + q_9\,\sigma_m^{n} + q_9\,\sigma_{m+1}^{n} + q_8\sigma_{m+2}^{n} + q_7\sigma_{m+3}^{n} + \varrho_1 \sum_{j=1}^{n} \sigma_{m-2}^{n-j+1}\, b_k + \varrho_2 \sum_{k=1}^{n} \sigma_{m-1}^{n-k+1}\, b_k + \varrho_3 \sum_{k=1}^{n} \sigma_m^{n-k+1}\, b_k + \varrho_3 \sum_{j=k}^{n} \sigma_{m+1}^{n-k+1}\, b_k + \varrho_2 \sum_{k=1}^{n} \sigma_{m+2}^{n-k+1}\, b_k + \varrho_1 \sum_{k=1}^{n} \sigma_{m+3}^{n-k+1}\, b_k\ , \qquad (20)$$

suppose the following form that can present the solution of (20): $\quad \sigma_m^n = \gamma^n e^{i\beta m h}$,

where, γ represents the time dependence of the solution, and the exponential function shows that the spatial dependence such that βh represents the position along the grid and i is $\sqrt{-1}$. By substituting, σ_m^n , into (20), we get:

$$G\,\gamma^{(n+1)} = H\,\gamma^{n} + R \sum_{k=1}^{n} \gamma^{(n-k+1)}\ b_k \qquad (21)$$

where,

$G = (q_1 - \varrho_1)e^{-2i\beta h} + (q_2 - \varrho_2)e^{-i\beta h} + (q_3 - \varrho_3) + (q_4 - \varrho_3)e^{i\beta h} + (q_5 - \varrho_2)e^{2i\beta h} + (q_6 - \varrho_1)e^{3i\beta h}$

$H = \left(q_7 e^{-2i\beta h} + q_8\, e^{-i\beta h} + q_9 + q_9\, e^{i\beta h} + q_8 e^{2i\beta h} + q_7 e^{3i\beta h}\right)$

$R = \left(\varrho_1 e^{-2i\beta h} + \varrho_2\, e^{-i\beta h} b_j + \varrho_3 + \varrho_3\, e^{i\beta h} + \varrho_2 e^{2i\beta h} + \varrho_1\, e^{3i\beta h}\right)$

and,

$q_1 = \left(\frac{1}{60} + \frac{1}{10}\Delta t\lambda - \frac{1}{2}\Delta t\beta\right)$, $q_2 = \left(\frac{57}{60} + \frac{25}{10}\Delta t\lambda - \frac{9}{2}\Delta t\beta\right)$, $q_3 = \left(\frac{302}{60} + \frac{40}{10}\Delta t\lambda + \frac{10}{2}\Delta t\beta\right)$,

$q_4 = \left(\frac{302}{60} - \frac{40}{10}\Delta t\lambda + \frac{10}{2}\Delta t\beta\right)$, $q_5 = \left(\frac{57}{60} - \frac{25}{10}\Delta t\lambda - \frac{9}{2}\Delta t\beta\right)$, $q_6 = \left(\frac{1}{60} - \frac{1}{10}\Delta t\lambda - \frac{1}{2}\Delta t\beta\right)$,

$q_7 = \left(\frac{1}{60}\right)$, $q_8 = \left(\frac{57}{60}\right)$, $q_9 = \left(\frac{302}{60}\right)$, $\varrho_1 = \left(\frac{1}{60}\ \frac{\Delta t^{2-\alpha}}{1-\alpha}\right)$, $\varrho_2 = \left(\frac{57}{60}\ \frac{\Delta t^{2-\alpha}}{1-\alpha}\right)$, $\varrho_3 = \left(\frac{302}{60}\ \frac{\Delta t^{2-\alpha}}{1-\alpha}\right)$.

From equation (21) we get

$$\gamma^n - \left(\frac{H}{G} + \frac{R}{G}\right)\gamma^{n-1} - \frac{R}{G}\sum_{k=2}^{n}\gamma^{n-k}\ b_k = 0 \tag{22}$$

we apply

$$B_1 = -\frac{H}{G} - \frac{R}{G}$$

$$B_k = -\frac{R}{G} b_k\ , \qquad k = 2, \dots, n \tag{23}$$

using (22) with (23) and applying them in equation (21) we have

$$\gamma^n - B_1\gamma^{n-1} - B_2\gamma^{n-2} + \cdots + B_{n-1}\gamma + B_n = 0 \tag{24}$$

It is easy to see that $G > 0$, $H > 0$ and $R > 0$. Hence all $B_1, B_2, \dots, B_n$ are positive in the equation (24).

Theorem 1:[17] from all values of the root x_i of the arbitrary polynomial as

$p(x) = a_0 x^n - a_1 x^{n-1} - a_2 x^{n-2} + \cdots + a_{n-1}x + a_n$,

we have

$$|\zeta_i| \le \max\left\{1, \sum_{j=1}^{n}\left|\frac{a_j}{a_0}\right|\right\}. \tag{25}$$

The stability must prove that $|\gamma_i| \ge 1$ in equation (24) from Theorem 1, $B_0 = 1$ and $B_k > 0, j = 1, \dots, n$ we have,

$$\sum_{k=1}^{n}\left|\frac{B_k}{B_0}\right| = \frac{-\left(H + R\ \sum_{k=1}^{n} b_k\right)}{G}$$

$$= \frac{-\left(H + R\ [(n+1)^{1-\alpha} - 1]\right)}{G} \tag{26}$$

where,

$\sum_{k=1}^{n}\ b_k = \sum_{k=1}^{n}\ [(k+1)^{1-\alpha} - k^{1-\alpha}]$

$$= (n+1)^{1-\propto} - 1 \tag{27}$$

let $\Lambda_\propto = (n+1)^{1-\propto} - 1$ from (26) we get

$$\sum_{k=1}^{n} B_k = \frac{-(H+R\,\Lambda_\propto)}{G} \tag{28}$$

if $R < 0$ and from (28) we get: $\frac{-(H+R\,\Lambda_\propto)}{G} < 1$

$$\frac{L_1\cos(2\beta h)+L_2 isin(2\beta h)+L_3\cos(\beta h)+L_4 isin(\beta h)+L_5\cos(3\beta h)+L_5 isin(3\beta h)+L_6}{L_7\cos(2\beta h)+L_8 isin(2\beta h))\ +L_9\cos(\beta h)+L_{10} isin(\beta h)+L_{11}\cos(3\beta h)+L_{11} isin(3\beta h)+L_{12}} < 1 \tag{29}$$

where,

$$L_1 = -\frac{58}{60} - \frac{58}{60}\frac{\Delta t^{2-\propto}}{1-\propto}\Lambda_\propto \quad , \quad L_2 = -\frac{56}{60} - \frac{56}{60}\frac{\Delta t^{2-\propto}}{1-\propto}\Lambda_\propto \quad , \quad L_3 = -\frac{359}{60} - \frac{359}{60}\frac{\Delta t^{2-\propto}}{1-\propto}\Lambda_\propto ,$$

$$L_4 = -\frac{245}{60} - \frac{245}{60}\frac{\Delta t^{2-\propto}}{1-\propto}\Lambda_\propto \ , \ L_5 = -\frac{1}{60} - \frac{1}{60}\frac{\Delta t^{2-\propto}}{1-\propto}\Lambda_\propto \quad , \quad L_6 = -\frac{302}{60} - \frac{302}{60}\frac{\Delta t^{2-\propto}}{1-\propto}\Lambda_\propto ,$$

$$L_7 = \frac{58}{60} - \frac{24}{10}\Delta t\lambda - \frac{10}{2}\Delta t\beta - \frac{58}{60}\frac{\Delta t^{2-\propto}}{1-\propto} \quad , \quad L_8 = \frac{56}{60} - \frac{26}{10}\Delta t\lambda - \frac{8}{2}\Delta t\beta - \frac{56}{60}\frac{\Delta t^{2-\propto}}{1-\propto},$$

$$L_9 = \frac{359}{60} - \frac{15}{10}\Delta t\lambda + \frac{1}{2}\Delta t\beta - \frac{359}{60}\frac{\Delta t^{2-\propto}}{1-\propto} \quad , \quad L_{10} = \frac{245}{60} - \frac{65}{10}\Delta t\lambda + \frac{19}{2}\Delta t\beta - \frac{245}{60}\frac{\Delta t^{2-\propto}}{1-\propto},$$

$$L_{11} = \frac{1}{60} - \frac{1}{10}\Delta t\lambda - \frac{1}{2}\Delta t\beta - \frac{1}{60}\frac{\Delta t^{2-\propto}}{1-\propto} \quad , \quad L_{12} = \frac{302}{60} + \frac{40}{10}\Delta t\lambda + \frac{10}{2}\Delta t\beta - \frac{302}{60}\frac{\Delta t^{2-\propto}}{1-\propto}.$$

The equation (29) is the necessary and sufficient condition for the stability of the proposed mothed.

From the above stability analysis, the present method is unconditionally stable.

4. Numerical Examples

In this section, we will apply the scheme described in section 3 to three test examples to demonstrate the efficiency, accuracy, and applicability of the present scheme. Results obtained by this scheme are compared with the analytical solution of each example and with [5], [12] by computing the maximum norm error L_∞ and norm error L_2.

Let, $t_n = nk$, $n = 0(1)M$, where M denoted the final time level t_M and $N+1$ is the number of the nodes to check the accuracy of the proposed method, where,

$$L_\infty = \max_{0\le i\le N}|u(x_i,t_M) - U_i^M|$$

$$L_2 = \frac{1}{N}\left(\sum_{i=0}^{N}|u(x_i,t_M) - U_i^M|^2\right)^{\frac{1}{2}}$$

Example 1:[12]

$u_t(x,t) + m\,u_x(x,t) - b\,u_{xx}(x,t) = \int_0^t (t-s)^{-\propto} u(x,s)ds + f(x,t) \quad x \in [0,1], \ \propto = \frac{1}{2}, \ t > 0$
$m = 0.05$, $b = 0.4$

The initial and boundary conditions are

$u(x,0) = sin\,\pi x \qquad 0 \le x \le 1$

$u(0,t) = u(1,t) = 0, \qquad 0 \le t \le T$

The exact solution is: $u(x,t) = (t+1)^2 \sin \pi x$.

Example 2:[12]

$u_t(x,t) + m\,u_x(x,t) - b\,u_{xx}(x,t) = \int_0^t (t-s)^{-\alpha}\,u(x,s)ds + f(x,t) \quad x \in [0,1],\ \alpha = \frac{1}{3},\ \ t > 0$

$m = 0.5\,, b = 0.005.$

The initial and boundary conditions are

$u(x,0) = \cos \pi x, \quad 0 \le x \le 1$

$u(0,t) = (t+1), \quad u(1,t) = -(t+1), \quad t \ge 0$

The exact solution is: $u(x,t) = (t+1)\cos \pi x.$

Example 3:[5]

$u_t(x,t) + m\,u_x(x,t) - b\,u_{xx}(x,t) = \int_0^t (t-s)^{-\alpha}\,u(x,s)ds + f(x,t) \quad x \in [0,1],\ \alpha = \frac{1}{2},\ \ t > 0$

$m = 0.5\,, b = 0.4$

The initial condition and boundary condition are

$u(x,0) = (x - x^2)\,, \quad 0 \le x \le 1$

$u(0,t) = u(1,t) = 0, \quad 0 \le t \le 1$

The exact solution is: $u(x,t) = (x - x^2)(t^2 + 1)$

Table 1: L_∞ and L_2 at $\Delta t = 0.0001$ and $\Delta t = 0.00001$ of Example1

h	M	L_2 (Δt =0.00001)	L_∞ (Δt =0.00001)	L_2 (Δt =0.0001)	L_∞ (Δt =0.0001)
0.1	10	5.6813e-08	2.4801e-07	5.6472e-07	2.4513e-06
0.02	50	8.4470e-10	9.1344e-09	8.1822e-09	8.1209e-08
0.001	100	1.4369e-10	2.1425e-09	1.3967e-09	2.1379e-08
0.0066	150	5.1276e-11	8.9552e-10	4.9789e-10	9.4531e-09
0.005	200	2.4807e-11	5.1250e-10	2.3913e-10	5.2231e-09
0.004	250	1.4159e-11	3.3574e-10	1.3536e-10	3.2852e-09
0.0033	300	8.9572e-12	2.3606e-10	8.5045e-11	2.2475e-09

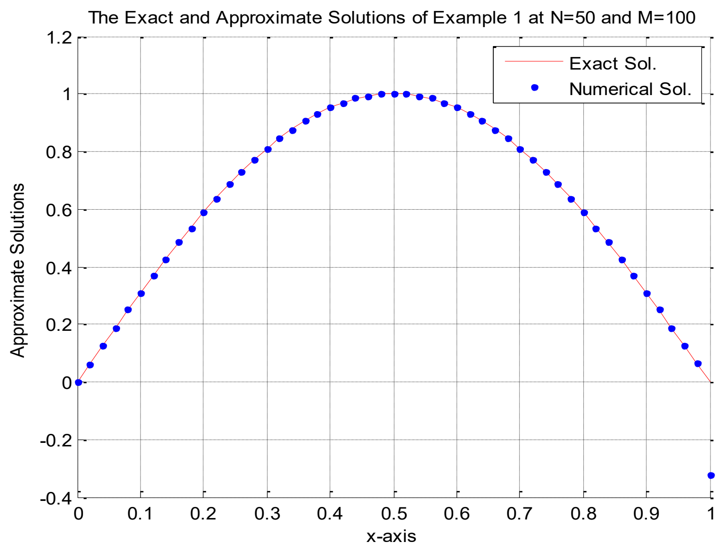


Table 2: L_∞ and L_2 at $\Delta t = 0.0001$ and $\Delta t = 0.00001$ of Example 2

h	M	L_2 (Δt =0.00001)	L_∞ (Δt =0.00001)	L_2 (Δt =0.0001)	L_∞ (Δt =0.0001)
0.1	10	2.5442e-08	1.1873e-07	2.5436e-07	1.1896e-06
0.02	50	5.1248e-10	6.8644e-09	5.1274e-09	6.9042e-08
0.001	100	1.2357e-10	2.3619e-09	1.2391e-09	2.3981e-08
0.0066	150	5.9420e-11	1.3380e-09	5.9774e-10	1.3809e-08
0.005	200	3.6572e-11	9.1623e-10	3.6898e-10	9.6677e-09
0.004	250	2.5452e-11	6.9236e-10	2.5708e-10	7.4957e-09
0.0033	300	1.9054e-11	5.5548e-10	1.9205e-10	6.1758e-09

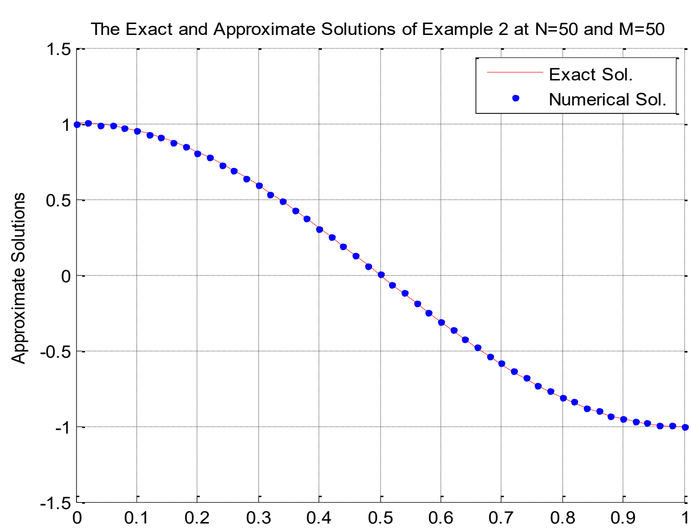
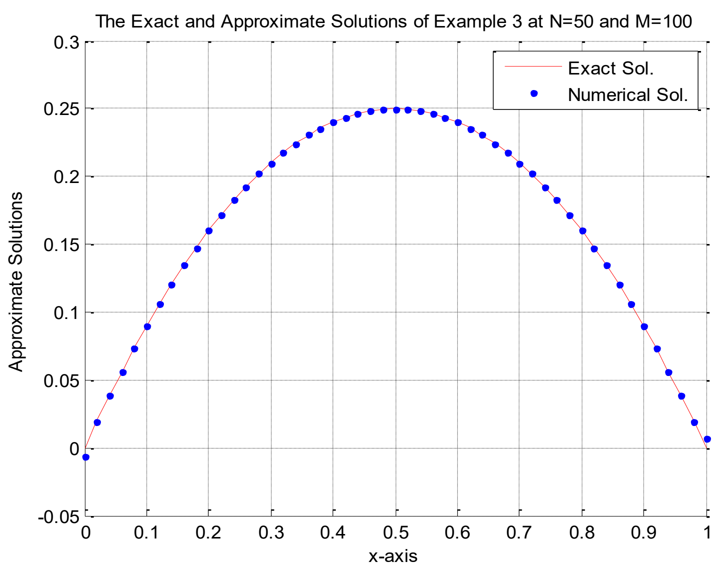


Table 3: L_∞ and L_2 *at* $\Delta t = 0.0001$ and $\Delta t = 0.00001$ of Example 3

h	M	L_2 (Δt =0.00001)	L_∞ (Δt =0.00001)	L_2 (Δt =0.0001)	L_∞ (Δt =0.0001)
0.1	10	6.5919e-09	4.1885e-08	6.5417e-08	4.2026e-07
0.02	50	6.3302e-11	1.7249e-09	4.5916e-10	1.6248e-08
0.001	100	7.2371e-12	4.2376e-10	7.0307e-11	3.3139e-09
0.0066	150	1.7266e-12	1.8113e-10	2.3985e-11	1.1548e-09
0.005	200	7.7409e-13	9.6355e-11	1.0357e-11	5.1434e-10
0.004	250	5.0706e-13	5.7676e-11	5.2986e-12	2.6662e-10
0.0033	300	3.5806e-13	3.7170e-11	3.0909e-12	1.5354e-10



5 Conclusions

In this article, partial integro-differential equations with the weakly singular kernel were solved by using the cubic B-spline Galerkin method with quadratic B-spline as a weight function. Backward Euler scheme was used for time direction, and the cubic B-spline Galerkin method with quadratic weight function used for spatial derivative. The numerical solutions for N=50, 100 and 200 with Δt=0.0001 and 0.00001 at different time M are presented in Tables 1-3. From Figures 1-3 the numerical and the exact solutions are very consistent which signalizes the numerical solutions effectively. We calculated L_2, and L_∞ norms errors are varied to test the accuracy of the proposed method. In addition, the numerical results are in good agreement with the exact solutions. Moreover, when comparing the results obtained with [5] and [12] found this method gives good results. The numerical cubic B-spline Galerkin scheme with quadratic B-spline as a weight function is an effectively and a unconditionally stable method.

References

[1] A. Abdul wahid: Approximation of Burgers equation using sextic B-spline Galerkin scheme with quantic weight function, Journal of University of Babylon for Pure and Applied Sciences, 26(8)(2018).

[2] A. Ali, Sh. Ahmad, S. Inayat, A. Shah, and F.I. Haq: A Quartic B-spline collocation technique for the solution of partial integro –differential equations with a weakly singular kernel, Sci.Int.(Lahore), 27(4) (2015) 2953-2958.

[3] H. O. Al-Humedi and A. Abd Al-wahed : Combine of B-Spline Galerkin Schemes with Change Weight Function, International Journal of Engineering Innovation & Research, 2(4)(2013) 2277 – 5668.

[4] M. Dehghan : Weighted finite difference techniques for the one-dimensional advection-diffusion equation. Applied Mathematics and Computation, 147(2004) 307-319.

[5] M. Gholamian and J. Saberi-Nadjafi: Cubic B-splines collocation method for a class of partial integro-differential equation, Alexandria Engineering Journal, 57(2017) 2157–2165.

[6] M. Gholamian , J. Saberi and A. Soheili: Cubic B-splines collocation method for solving a partial integro-differential equation with a weakly singular kernel, Computational Methods for Differential Equations, 7(3)(2018) 497-510.

[7] F. I. Haq, S. Islam and I. A. Tirmizi: A numerical technique for solution of the MRLW equation using quartic B-splines, Applied Mathematical Modelling, 34(2010) 4151–4160.

[8] A.R .Mitchell and D.F. Griffiths: The finite difference equation in partial differential equation. Johan Wiley & Sons New York (1980).

[9] R.C. Mittal and R.K. Jain : Redefined cubic B-splines collocation method for solving convection–diffusion equations, Applied Mathematical Modelling, 36(2012) 5555–5573.

[10] J. Noye :Numerical Solutions of Partial Differential Equation, North Holland Publishing company-Amsterdam. New York. Oxford. (1982).

[11] P.M .Prenter: spline and variational Method, John Wiley and Sons, New York , NY,USA. (1975).

[12] S.S. Siddiqi and S. Arshad: Numerical solution of convection-diffusion integro-differential equations with a weakly singular kernel. Journal of Basic and Applied Scientific Research, 3(11)(2013) 106-120.

[13] S.S. Siddiqi and S. Arshad: Cubic B-Spline for the Numerical Solution of Parabolic Integro-differential Equation with a Weakly Singular Kernel, Research Journal of Applied Sciences, Engineering and Technology, 7(10)(2014) 2065-2073.

[14] A.F. Soliman, M.S. El-Azab and A.M.A. El-Asyed: The numerical solution of integro-partial differential equations with sixth-degree b-spline functions, American Academic & Scholarly Research Journal, 4(2)(2012).

[15] A. F. Soliman, A M.A. El-Asyed and M. S. El-Azab: On the numerical solution of partial integro- differential equations, Mathematical Sciences Letters, 1(1) (2012) 71-80.

[16] D. B. Spalding: A novel finite difference formulation for differential expressions involving both first and second derivatives, International Journal for Numerical Method in Engineering, 4(1972) 551-559.

[17] J. Stoer and R. Bulirsch: Introduction to numerical analysis, second edition, springer-verlag, New York, 1991.

[18] T. Tang: A finite difference scheme for partial integro-differential equations with a weakly singular kernel, Applied Numerical Mathematic, 11(1993) 309-319.

[19] W. Tuzikiewics and M . Duda: Bioheat transfer equation. The Problem of FDM Explicit Scheme Stability, Jounal of Applied Mathematics and Computation Mechanics, 14(4)(2015) 139–144.

[20] H. Zhang, X.Han and X. Yang: Quintic B-spline collocation method for fourth order partial Integro-differential equations with a weakly singular kernel, Applied Mathematics and Computation, 219(2013) 6565–6575.

[21] H. Zhang and X. Hana: Quasi-wavelet method for time dependent fractional partial differential equation, International Journal of Computer Mathematics, 90(11)(2013) 2491–2507.