Journal of Education and Science (ISSN 1812-125X)

www.edusj.mosuljournals.com

Numerical Solution of the Fredholm Integro-Differential Equations using High-Order Compact Finite Difference Method

S. R. Saber[1] Y. A. Sabawi[1,2] M. S. Hasso[1]

[1] Department of Mathematics, Faculty of Science and Health, Koya University, Kurdistan Region - F.R. Iraq
[2] Department of mathematics Education, Faculty of Education, Tishk International University, Erbil, Iraq

Article information

Article history:
Received February 16, 2023
Accepted April 16, 2023
Available online September 01, 2023

Keywords:
Fredholm Integro-Differential Equation
Fourth-Order Compact Finite Difference Method (CFDM)
Sixth-Order Compact Finite Difference Method (CFDM)
Composite Boole's Rule

Correspondence:
Younis A. Sabawi
younis.abid@koyauniversity.org

Abstract

This work aims to present a numerical method for solving Fredholm integro-differential equations (FIDE). This work discusses the use of a fourth and sixth-order compact finite difference method (CFDM) based on composite Boole's rule to solve FIDE. The accuracy of the suggested schemes is computed through $l^2$and l^∞ norms and the efficiency of the approach is assessed through short CPU-time values. An important factor of the proposed methods is leading to a reduction in the computational cost of the schemes. This is a significant improvement over traditional methods, which often struggle to maintain high accuracy levels. The presented methods are shown to be the fourth and sixth order in space. Numerical experiments are presented to illustrate the performance of the suggested methods. Overall, the proposed method is a significant step forward in the field of solving FIDE problems. It offers a robust and efficient numerical approach that can achieve high levels of accuracy where exact solutions are hard to obtain.

DOI: 10.33899/edusj.2023.138218.1332,

Introduction

Initial and boundary value problems with integro-differential equations are common in applications of (bio-)engineering as well as physical and biological modeling. CFDM to approximate solutions to such problems, especially in the context of the ordinary and partial differential equation has attracted much interest [1-10].

However, comparatively, there has been less progress made in determining high-order CFDM in terms of integro-differential equations (IDE). Therefore, considerable works have been focusing on developing efficient high-order numerical schemes for approximating solutions of integro-differential equations. This work concentrates on the second order FIDE:

$$u''(x) + p\,u(x) = f(x) + \lambda \int_a^b k(x,t)\,u(t)dt, \qquad (1)$$

with Dirichlet boundary conditions:

$$u(a) = \alpha, \qquad u(b) = \beta. \qquad (2)$$

For $x, t \in [a, b]$, where λ, p, α, and β are constant values, $f(x)$ and $k(x,t)$ are known functions and $u(x)$ is the solution to be determined.

Recently, there has been a growing interest in using higher-order numerical methods for solving partial differential equations (ODEs) and (PDEs). One approach that has gained attention is the use of compact difference approximations, which can achieve a high level of accuracy with a relatively small number of grid points. These approximations make use of five grid points, corresponding to a compact patch of three cells surrounding a selected node, to cancel out second-order truncation error terms. This allows for the development of alternative, lower-derivative expressions that are equivalent to the higher-order truncation error terms [10]. This approach can lead to more efficient and accurate solutions for ODEs and PDEs.

Several numerical solutions of the integro-differential equations have been studied by compact finite difference methods including [11-13]. Numerous authors have developed numerical methods for integral and integro-differential equations recently, see references [14-19].

This work aims to derive a general formulation and approach for developing such higher-order compact (HOC) schemes for the second-order Volterra integral. This derivation is based on applying numerical quadrature rules along with the properties of ordinary differential equations. Furthermore, we compute the order of convergence numerically for each method. The proposed methods are tested on various PIDE to demonstrate their efficiency and accuracy in providing approximate solutions. The results show that the of order fourth and sixth CFDM is effective in solving PIDEs and can be used to obtain reliable solutions for a variety of applications.

The work is organized as follows. In section 2, the derivation of HOC method is given in detail. Some numerical experiments and algorithms are shown in section 3. Finally, conclusions are given in section 4.

2. Compact finite difference method

This section presents a way to develop a CFDM based on the fourth and sixth-order approximation for the FIDE.

2.1 Fourth-order (CFDM4)

To derive the CFDM4 for (1), applying $\delta_x^2 u_i = \frac{u_{i+1}-2u_i+u_{i-1}}{h^2}$ is a second-order central difference scheme [9], gives:

$$\delta_x^2 u_i = u_i'' + \frac{h^2}{12}u_i^{(4)} + O(h^4). \tag{3}$$

To obtain a compact $O(h^4)$ approximation, we take the derivative of Eq. (1) with respect to x, which gives:

$$u_i^{(4)} = -pu_i'' + f_i'' + \lambda \int_a^b k''_{(i,j)}\, u_j dt, \tag{4}$$

where $u_i = u(x), f_i = f(x), k_{i,j} = k(x,t)$ and $u_j = u(t)$. Inserting Eq. (4) into Eq. (3), we have:

$$\delta_x^2 u_i = u_i'' + \frac{h^2}{12}\left(-pu_i'' + f_i'' + \lambda \int_a^b k''_{(i,j)}\, u_j dt\right) + O(h^4). \tag{5}$$

Some simplification in the above equation implies that:

$$u_i'' = \frac{\delta_x^2 u_i - \frac{h^2}{12} f_i'' - \frac{\lambda h^2}{12}\int_a^b k''_{(i,j)}\, u_j\, dt}{1 - \frac{ph^2}{12}} + O(h^4). \tag{6}$$

Substituting Eq. (6) into Eq. (1) we obtain:

$$\delta_x^2 u_i + p\left(1 - \frac{ph^2}{12}\right)u_i = \left(1 - \frac{ph^2}{12}\right)f_i + \frac{h^2}{12}f_i'' + \lambda\left(1 - \frac{ph^2}{12}\right)\int_a^b k_{(i,j)}\, u_j dt + \frac{\lambda\, h^2}{12}\int_a^b k''_{(i,j)}\, u_j dt. \tag{7}$$

The integral parts on the right-hand side of Eq. (7) will be handled numerically using the composite Boole's rule [20] given by:

$$\int_{x_0}^{x_n} u(x)dx = \frac{2h}{45}\left[7u(x_0) + 32\sum_{j=1}^{\frac{n}{2}} u(x_{2j-1}) + 12\sum_{j=1}^{\left(\frac{n}{4}\right)} u(x_{4j-2}) + 14\sum_{j=1}^{\left(\frac{n}{4}\right)-1} u(x_{4j}) + 7u(x_n)\right]. \tag{8}$$

Therefore, using Eq. (8) for Eq. (7) we obtain:

$$\int_a^b k_{(i,j)} u_j\, dt = \frac{2h}{45}\left[7k_{(i,0)}\, u_0 + 32\sum_{j=1}^{\frac{n}{2}} k_{(i,2j-1)}\, u_{2j-1} + 12\sum_{j=1}^{\left(\frac{n}{4}\right)} k_{(i,4j-2)}\, u_{4j-2} \right.$$

$$\left. + 14\sum_{j=1}^{\left(\frac{n}{4}\right)-1} k_{(i,4j)}\, u_{4j} + 7k_{(i,n)}\, u_n\right] \tag{9}$$

$$\int_a^b k''_{(i,j)} u_j\, dt = \frac{2h}{45}\left[7k''_{(i,0)}\, u_0 + 32\sum_{j=1}^{\frac{n}{2}} k''_{(i,2j-1)}\, u_{2j-1} + 12\sum_{j=1}^{\left(\frac{n}{4}\right)} k''_{(i,4j-2)}\, u_{4j-2} \right.$$

$$\left. + 14\sum_{j=1}^{\left(\frac{n}{4}\right)-1} k''_{(i,4j)}\, u_{4j} + 7k''_{(i,n)}\, u_n\right]. \tag{10}$$

Substituting Eq. (9) and Eq. (10) into Eq. (7) then using second-order central differencing scheme we obtain:

$$\gamma_1 u_{i+1} + \gamma_2 u_i + \gamma_1 u_{i-1} - \sum_{j=1}^{\frac{n}{2}} a_{(i,j)}\, u_{2j-1} - \sum_{j=1}^{\frac{n}{4}} b_{(i,j)}\, u_{4j-2} - \sum_{j=1}^{\frac{n}{4}-1} c_{(i,j)}\, u_{4j} = f_i + \alpha_i u_0 + \beta_i u_n, \tag{11}$$

where:

$$\gamma_1 = 1, \qquad \gamma_2 = -2 + ph^2 - \frac{p^2 h^4}{12}$$

$$a_{(i,j)} = \left(\frac{32\,\lambda h^3}{27} - \frac{32\,\lambda p h^5}{270}\right) k_{(i,2j-1)} + \frac{32\,\lambda p h^3}{270}\left(k_{(i+1,2j-1)} + k_{(i-1,2j-1)}\right)$$

$$b_{(i,j)} = \left(\frac{12\,\lambda h^3}{27} - \frac{12\,\lambda p h^5}{270}\right) k_{(i,4j-2)} + \frac{12\,\lambda p h^3}{270}\left(k_{(i+1,4j-2)} + k_{(i-1,4j-2)}\right),$$

$$c_{(i,j)} = \left(\frac{14\,\lambda h^3}{27} - \frac{14\,\lambda p h^5}{270}\right) k_{(i,4j)} + \frac{14\,\lambda p h^3}{270}\left(k_{(i+1,4j)} + k_{(i-1,4j)}\right),$$

$$f_i = \frac{h^2}{12}\left(f_{i+1} + 10 f_i + f_{i-1}\right),$$

$$\alpha_i = \left(\frac{7\,\lambda h^3}{27} - \frac{7\,\lambda p h^5}{270}\right) k_{(i,0)} + \frac{7\,\lambda p h^3}{270}\left(k_{(i+1,0)} + k_{(i-1,0)}\right),$$

$$\beta_i = \left(\frac{7\,\lambda h^3}{27} - \frac{7\,\lambda p h^5}{270}\right) k_{(i,n)} + \frac{7\,\lambda p h^3}{270}\left(k_{(i+1,n)} + k_{(i-1,n)}\right).$$

2.2 Sixth-order (CFDM6)

Starting with the derivation of the CFDM6 for (1), gives:

$$\delta_x^2 u_i = u_i'' + \frac{h^2}{12} u_i^{(4)} + \frac{h^4}{360} u_i^{(6)} + O(h^6). \tag{12}$$

Eq. (12) includes both $O(h^2)$ and $O(h^4)$ terms since we want to approximate both of them in order to create an $O(h^6)$ scheme. Applying δ_x^2 to $u_i^{(4)}$, we obtain:

$$u_i^{(6)} = \delta_x^2 u_i^{(4)} + O(h^2). \tag{13}$$

Substituting Eq. (13) into Eq. (12) yields:

$$\delta_x^2 u_i = u_i'' + \frac{h^2}{12} u_i^{(4)} + \frac{h^4}{360}\left(\delta_x^2 u_i^{(4)} + O(h^2)\right) + O(h^6). \tag{14}$$

Go back to Eq. (1), with take derivative, becomes:

$$u_i^{(4)} = -p u_i'' + f_i'' + \lambda \int_a^b k''_{(i,j)}\, u_j dt. \tag{15}$$

Substituting Eq. (15) into Eq. (14), leads to:

$$\delta_x^2 u_i = u_i'' + \left(\frac{h^2}{12} + \frac{h^4}{360}\delta_x^2\right)\left(-pu_i'' + f_i'' + \lambda\int_a^b k_{(i,j)}''\, u_j dt\right) + O(h^6), \tag{16}$$

with some calculations, Eq. (16), yields:

$$u_i'' = \frac{\delta_x^2 u_i - \left(\frac{h^2}{12} + \frac{h^4}{360}\delta_x^2\right) f_i'' - \lambda\left(\frac{h^2}{12} + \frac{h^4}{360}\delta_x^2\right)\int_a^b k_{(i,j)}''\, u_j dt}{1 - p\left(\frac{h^2}{12} + \frac{h^4}{360}\delta_x^2\right)} + O(h^6). \tag{17}$$

Substituting Eq. (17) into Eq. (1) implies that:

$$\delta_x^2 u_i + p\left(1 - p\left(\frac{h^2}{12} + \frac{h^4}{360}\delta_x^2\right)\right) u_i = \left(1 - p\left(\frac{h^2}{12} + \frac{h^4}{360}\delta_x^2\right)\right) f_i + \left(\frac{h^2}{12} + \frac{h^4}{360}\delta_x^2\right) f_i''$$
$$+\lambda\left(1 - p\left(\frac{h^2}{12} + \frac{h^4}{360}\delta_x^2\right)\right)\int_a^b k_{(i,j)}\, u_j dt + \lambda\left(\frac{h^2}{12} + \frac{h^4}{360}\delta_x^2\right)\int_a^b k_{(i,j)}''\, u_j dt \tag{18}$$

Setting $\delta_x^2 f_i'' = (f_{i+1}'' - 2f_i'' + f_{i-1}'')/h^2$ and $\delta_x^2 k_{(i,j)}'' = \left(k_{(i+1,j)}'' - 2k_{(i,j)}'' + k_{(i-1,j)}''\right)/h^2$, gives:

$$\left(-2 + ph^2 - \frac{28}{360}p^2h^4\right)u_i + \left(1 - \left(\frac{p^2h^4}{360}\right)\right)(u_{i+1} + u_{i-1}) = \left(h^2 - \frac{28}{360}ph^4\right) f_i$$
$$-\frac{ph^4}{360}(f_{i+1} + f_{i-1}) + \frac{28}{360}h^4 f_i'' + \frac{h^4}{360}(f_{i+1}'' + f_{i-1}'') + \left(h^2 - \frac{28}{360}ph^4\right)\int_a^b k_{(i,j)}\, u_j\, dt$$
$$-\frac{ph^4}{360}\int_a^b \left(k_{(i+1,j)} + k_{(i-1,j)}\right) u_j\, dt + \frac{28}{360}h^4\int_a^b k_{(i,j)}''\, u_j\, dt + \frac{h^4}{360}\int_a^b \left(k_{(i+1,j)}'' + k_{(i-1,j)}''\right) u_j\, dt. \tag{19}$$

The integral parts on the right-hand side of Eq. (19) will be handled numerically using the composite Boole's rule given by (8):

$$\int_a^b k_{(i,j)} u_j\, dt = \frac{2h}{45}\Bigg[7k_{(i,0)}\, u_0 + 32\sum_{j=1}^{\frac{n}{2}} k_{(i,2j-1)}\, u_{2j-1} + 12\sum_{j=1}^{\left(\frac{n}{4}\right)} k_{(i,4j-2)}\, u_{4j-2}$$
$$+ 14\sum_{j=1}^{\left(\frac{n}{4}\right)-1} k_{(i,4j)}\, u_{4j} + 7k_{(i,n)}\, u_n\Bigg] \tag{20}$$

$$\int_a^b \left(k_{(i+1,j)} + k_{(i-1,j)}\right) u_j\, dt = \frac{2h}{45}\Bigg[7\left(k_{(i+1,0)} + k_{(i-1,0)}\right) u_0 + 32\sum_{j=1}^{\frac{n}{2}}\left(k_{(i+1,2j-1)} + k_{(i-1,2j-1)}\right) u_{2j-1}$$
$$+12\sum_{j=1}^{\left(\frac{n}{4}\right)}\left(k_{(i+1,4j-2)} + k_{(i-1,4j-2)}\right) u_{4j-2} + 14\sum_{j=1}^{\left(\frac{n}{4}\right)-1}\left(k_{(i+1,4j)} + k_{(i-1,4j)}\right) u_{4j} +$$
$$+7\left(k_{(i+1,n)} + k_{(i-1,n)}\right) u_n\Bigg] \tag{21}$$

$$\int_a^b k_{(i,j)}'' u_j\, dt = \frac{2h}{45}\Bigg[7k_{(i,0)}''\, u_0 + 32\sum_{j=1}^{\frac{n}{2}} k_{(i,2j-1)}''\, u_{2j-1} + 12\sum_{j=1}^{\left(\frac{n}{4}\right)} k_{(i,4j-2)}''\, u_{4j-2}$$
$$+ 14\sum_{j=1}^{\left(\frac{n}{4}\right)-1} k_{(i,4j)}''\, u_{4j} + 7k_{(i,n)}''\, u_n\Bigg] \tag{22}$$

$$\int_a^b \left(k_{(i+1,j)}'' + k_{(i-1,j)}''\right) u_j\, dt = \frac{2h}{45}\Bigg[7\left(k_{(i+1,0)}'' + k_{(i-1,0)}''\right) u_0 + 32\sum_{j=1}^{\frac{n}{2}}\left(k_{(i+1,2j-1)}'' + k_{(i-1,2j-1)}''\right) u_{2j-1}$$

$$+12\sum_{j=1}^{\left(\frac{n}{4}\right)}\left(k''_{(i+1,4j-2)}+k''_{(i-1,4j-2)}\right)u_{4j-2}+14\sum_{j=1}^{\left(\frac{n}{4}\right)-1}\left(k''_{(i+1,4j)}+k''_{(i-1,4j)}\right)u_{4j}$$

$$+7\left(k''_{(i+1,n)}+k''_{(i-1,n)}\right)u_n\Big]. \quad (23)$$

Substituting Eq. (20) –Eq. (23) into Eq. (19) we obtain:

$$\gamma_1 u_{i+1}+\gamma_2 u_i+\gamma_1 u_{i-1}-\sum_{j=1}^{\frac{n}{2}}a_{(i,j)}\,u_{2j-1}-\sum_{j=1}^{\frac{n}{4}}b_{(i,j)}\,u_{4j-2}-\sum_{j=1}^{\frac{n}{4}-1}c_{(i,j)}\,u_{4j}=f_i+\alpha_i u_0+\beta_i u_n \quad (24)$$

where:

$\gamma_1 = 1-\left(\frac{p^2h^4}{360}\right)$, $\gamma_2 = -2+ph^2-\frac{7}{90}p^2h^4$

$a_{(i,j)} = \left(\frac{64\,\lambda h^3}{45}-\frac{224\,\lambda ph^5}{2025}\right)k_{(i,2j-1)}-\frac{8\,\lambda ph^5}{2025}\left(k_{(i+1,2j-1)}+k_{(i-1,2j-1)}\right)+\frac{224\,\lambda h^5}{2025}k''_{(i,2j-1)}+$

$\frac{8\,\lambda h^5}{2025}\left(k''_{(i+1,2j-1)}+k''_{(i-1,2j-1)}\right)$

$b_{(i,j)} = \left(\frac{24\,\lambda h^3}{45}-\frac{84\,\lambda ph^5}{2025}\right)k_{(i,4j-2)}-\frac{3\,\lambda ph^5}{2025}\left(k_{(i+1,4j-2)}+k_{(i-1,4j-2)}\right)+\frac{84\,\lambda h^5}{2025}k''_{(i,4j-2)}+$

$\frac{3\,\lambda h^5}{2025}\left(k''_{(i+1,4j-2)}+k''_{(i-1,4j-2)}\right)$

$c_{(i,j)} = \left(\frac{28\,\lambda h^3}{45}-\frac{98\,\lambda ph^5}{2025}\right)k_{(i,4j)}-\frac{7\,\lambda ph^5}{4050}\left(k_{(i+1,4j)}+k_{(i-1,4j)}\right)+\frac{98\,\lambda h^5}{2025}k''_{(i,4j)}+\frac{7\,\lambda h^5}{4050}\left(k''_{(i+1,4j)}\right.$

$\left.+k''_{(i-1,4j)}\right)$

$f_i = \left(h^2-\frac{7}{90}ph^4\right)f_i-\frac{ph^4}{360}\left(f_{i+1}+f_{i-1}\right)+\frac{7}{90}h^4f''_i+\frac{h^4}{360}\left(f''_{i+1}+f''_{i-1}\right)$

$\alpha_i = \left(\frac{14\,\lambda h^3}{45}-\frac{49\,\lambda ph^5}{2025}\right)k_{(i,0)}-\frac{7\,\lambda ph^5}{8100}\left(k_{(i+1,0)}+k_{(i-1,0)}\right)+\frac{49\,\lambda h^5}{2025}k''_{(i,0)}+\frac{7\,\lambda h^5}{8100}\left(k''_{(i+1,0)}+k''_{(i-1,0)}\right)$

$\beta_i = \left(\frac{14\,\lambda h^3}{45}-\frac{49\,\lambda ph^5}{2025}\right)k_{(i,n)}-\frac{7\,\lambda ph^5}{8100}\left(k_{(i+1,n)}+k_{(i-1,n)}\right)+\frac{49\,\lambda h^5}{2025}k''_{(i,n)}+\frac{7\,\lambda h^5}{8100}\left(k''_{(i+1,n)}+k''_{(i-1,n)}\right).$

3. Numerical Experiments

The section shows the accuracy of a proposed method, using MATLAB programming. The error norms of l^2 and l^∞ are used to measure the error between the numerical and analytical solutions.
We denote by E errors terms given by:

$$\mathrm{E}(x) = u\,(x) - U_{Appro.}(x)$$

Let us introduce the three accuracy indicators when using space step size h, as follows:

- The Absolute (pointwise) error as:

$$\mathcal{E}(x) = |\mathrm{E}(x_i)|$$

- The l^∞-norm and l^2-norm of the error as:

$$l^\infty(\mathrm{E},h) = \max_{0\le i\le N}|\mathrm{E}(x_i)|\,, \qquad l^2(\mathrm{E},h) = \sqrt{h\sum_{i=0}^{N}|\mathrm{E}(x_i)|^2}$$

- The order of convergence $(Rate)$ is calculated as:

$$Rate = \frac{log\left(\frac{Error(N_1)}{Error(N_2)}\right)}{log\left(\frac{N_2}{N_1}\right)}$$

Algorithm

Input: $N(rem(N,4)=0), a, b, p,$ and boundary condition u_0 and u_n , where $(a = x_0, b = x_n)$.

Set: $h = \frac{b-a}{N}$.
for $i \longleftarrow 0$ to N do
  for $j \longleftarrow 0$ to N do
    $x_i = a + ih$.
    $t_j = a + jh$.
  end for
end for
for $i \longleftarrow 1$ to $N-1$ do
  $C_i = f_i + \alpha_i u_0 + \beta_i u_n$.
  for $j \longleftarrow 1$ to $\left(\frac{N}{2}\right)$ do
    $\mathcal{B}_{(i,2j-1)} \longleftarrow -a_{(i,j)}$.
  end for
  for $j \longleftarrow 1$ to $\left(\frac{N}{4}\right)$ do
    $\mathcal{B}_{(i,4j-2)} \longleftarrow -b_{(i,j)}$.
  end for
  for $j \longleftarrow 1$ to $\left(\frac{N}{4} - 1\right)$ do
    $\mathcal{B}_{(i,4j)} \longleftarrow -c_{(i,j)}$.
  end for
end for
for $i \longleftarrow 1$ to $N-1$ do
  $\mathcal{B}_{(i,i)} = \mathcal{B}_{(i,i)} + \gamma_2$
  if $i <= N-2$ then
    $\mathcal{B}_{(i,i+1)} = \mathcal{B}_{(i,i+1)} + \gamma_1$
    $\mathcal{B}_{(i+1,i)} = \mathcal{B}_{(i+1,i)} + \gamma_1$
  end if
end for
$C = [C_1 - \gamma_1 u_0 \,; C_2 : C_{n-2} \,; C_{n-1} - \gamma_1 u_n]$
Output: $U \longleftarrow \mathcal{B} \setminus C$

Example 1: Consider FIDE:

$$u''(x) + 5u(x) = 4\, si\, n\, x + \frac{10\,\pi}{3} co\, s(x) + \frac{5}{3}\int_{0}^{2\pi} \cos(x)\, t\, u(t)\, dt,$$

with boundary conditions: $u(0) = 0,\ u(2\pi) = 0,$
and exact solution is $u(x) = sin\,(x)$.

Table 1. Numerical Results for Example 1, by using CFDM4 and CFDM6 with $N = 12,\ h = 0.5236$ and $0 \leq x \leq 2\pi$

x_i	u(x)	$\mathbf{U}_{Appro.}(x)$ CFDM4	$\mathbf{U}_{Appro.}(x)$ CFDM6
$\pi/6$	5.0000e-01	5.0059e-01	4.9998e-01

$\pi/3$	8.6603e-01	8.6615e-01	8.6606e-01
$\pi/2$	1.0000e+00	9.9967e-01	1.0001e+00
$2\pi/3$	8.6603e-01	8.6627e-01	8.6605e-01
$5\pi/6$	5.0000e-01	5.0143e-01	4.9991e-01
π	1.2246e-16	1.8444e-03	-1.4901e-04
$7\pi/6$	-5.0000e-01	-4.9923e-01	-5.0009e-01
$4\pi/3$	-8.6603e-01	-8.6693e-01	-8.6600e-01
$3\pi/2$	-1.0000e+00	-1.0016e+00	-9.9992e-01
$5\pi/3$	-8.6603e-01	-8.6704e-01	-8.6599e-01
$11\pi/6$	-5.0000e-01	-5.0007e-01	-5.0002e-01
$l^2(E,h)$		2.4130e-03	1.7052e-04
$l^\infty(E,h)$		1.8444e-03	1.4901e-04
CPU-time		1.784562	1.895163

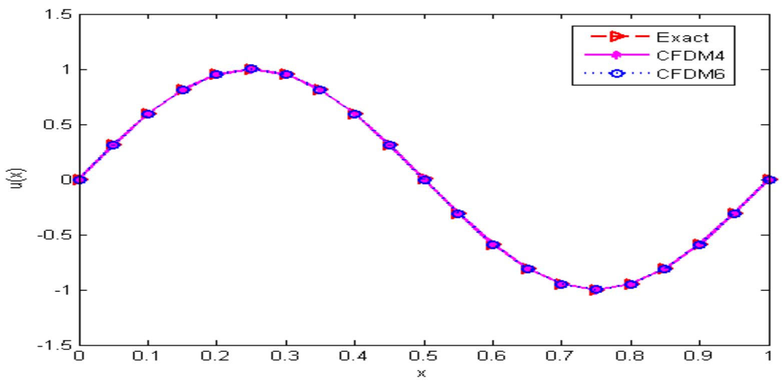

Figure 1: Exact and Approximate Solution of CFDM4 and CFDM6 for Example 1 with N=20 and h = 0.3142

Table 2: Rate Convergence of CFDM4 and CFDM6 for $l^2(E,h)$ in Example 1

N	l^2 −CFDM4	Rate	l^2 −CFDM6	Rate
12	2.4130e-03		1.7052e-04	
24	1.4037e-04	4.1035	2.3573e-06	6.1767
48	8.6536e-06	4.0198	3.5835e-08	6.0396
96	5.3909e-07	4.0047	5.5610e-10	6.0099

Table 3: Rate Convergence of CFDM4 and CFDM4 for $l^\infty(E,h)$ in Example 1

N	l^∞ −CFDM4	Rate	l^∞ −CFDM6	Rate
12	1.8444e-03		1.4901e-04	
24	1.0831e-04	4.0899	2.0584e-06	6.1777
48	6.8042e-06	3.9926	3.1289e-08	6.0397

96	4.2413e-07	4.0038	4.8557e-10	6.0098

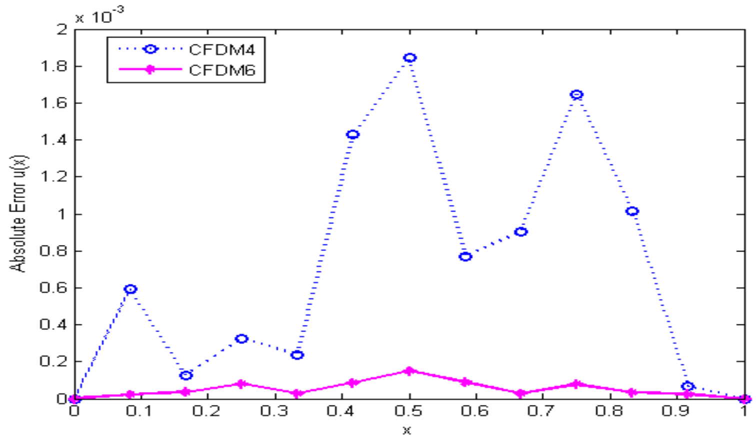

Figure 2: Comparison Absolute Error of CFDM4 and CFDM6 for Example 1 with N=12 and h=0.5236

Example 2. Consider FIDE:

$$u''(x) - 2u(x) = 2e^{-2x} - \left(\frac{-9\,e^{-8}+1}{2}\right)x^4 + \int_0^4 2x^4 t\,u(t)\,dt$$

with boundary conditions: $u(0) = 1,\ u(4) = e^{-8}$
and the exact solution is $u(x) = e^{-2x}$.

Table 4. Numerical Results for Example 2, by using CFDM4 and CFDM6 with $N = 12,\ h = 0.3333$ and $0 \leq x \leq 4$

x_i	u(x)	$U_{Appro.}(x)$ CFDM4	$U_{Appro.}(x)$ CFDM6
0.3333	5.1342e-01	5.1339e-01	5.1341e-01
0.6667	2.6360e-01	2.6357e-01	2.6360e-01
1	1.3534e-01	1.3531e-01	1.3534e-01
1. 3333	6.9483e-02	6.9470e-02	6.9488e-02
1.6667	3.5674e-02	3.5672e-02	3.5683e-02
2	1.8316e-02	1.8325e-02	1.8330e-02
2. 3333	9.4036e-03	9.4249e-03	9.4246e-03
2.6667	4.8279e-03	4.8608e-03	4.8561e-03
3	2.4788e-03	2.5210e-03	2.5129e-03
3. 3333	1.2726e-03	1.3184e-03	1.3087e-03
3.6667	6.5339e-04	6.8951e-04	6.8145e-04
$l^2(E,h)$		5.5724e-05	4.0040e-05
$l^\infty(E,h)$		4.5808e-05	3.6067e-05
CPU-time		1.353563	1.324556

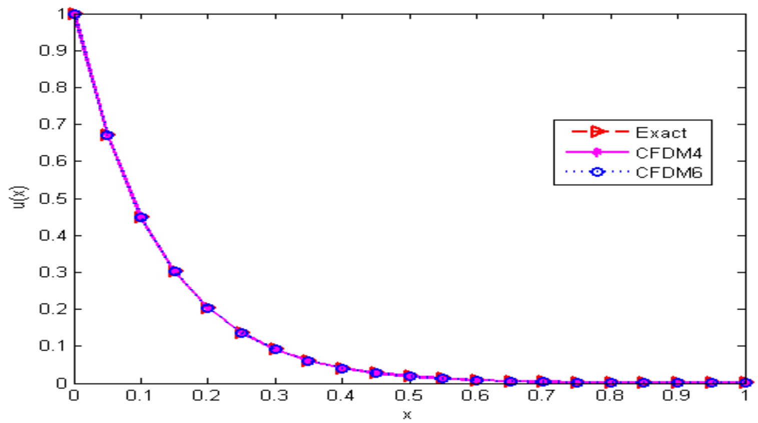

Figure 3. Exact and Approximate Solution of CFDM4 and CFDM6 for Example 2 with $N = 20$ and $h = 0.2$

Table 5. Rate Convergence of CFDM4 and CFDM6 for $l^2(E,h)$ in Example 2

N	l^2 −CFDM4	Rate	l^2 −CFDM6	Rate
12	5.5724e-05		4.0040e-05	
24	2.4233e-06	4.5233	7.7961e-07	5.6825
48	1.4023e-07	4.1111	1.2921e-08	5.9150
96	8.6612e-09	4.0171	2.0494e-10	5.9784

Table 6. Rate Convergence of CFDM4 and CFDM6 for $l^\infty(E,h)$ in Example 2

N	l^∞ −CFDM4	Rate	l^∞ −CFDM6	Rate
12	4.5808e-05		3.6067e-05	
24	2.0689e-06	4.4687	6.9973e-07	5.6877
48	1.3230e-07	3.9670	1.1646e-08	5.9089
96	8.3039e-09	3.9939	1.8469e-10	5.9786

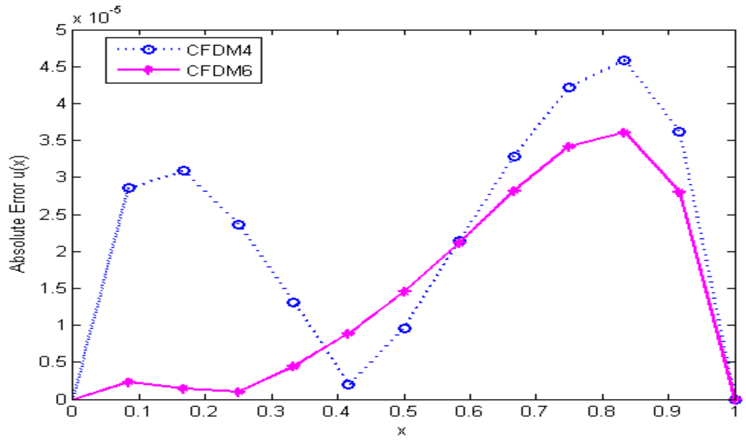

Figure 4. Comparison Absolute Error of CFDM4 and CFDM6 for Example 2 with $N = 12$ and $h = 0.3333$

Example 3. Consider FIDE:

$$u''(x) + 3u(x) = 2\cos(x) + 3\pi e^{2x} + \frac{3}{2}\int_0^{\pi} e^{2x}t^2\, u(t)\, dt$$

with boundary conditions: $u(0) = 1,\ u(\pi) = -1$
and the exact solution is $u(x) = \cos(x)$.

Table 7. Numerical results for Example 3, by using CFDM4 and CFDM6 with $N = 12$ $h = 0.2618$ and $0 \le x \le \pi$

x_i	u(x)	$\mathbf{U}_{Appro.}(x)$ CFDM4	$\mathbf{U}_{Appro.}(x)$ CFDM6
$\pi/12$	9.6593e-01	9.6594e-01	9.6592e-01
$\pi/6$	8.6603e-01	8.6606e-01	8.6602e-01
$\pi/4$	7.0711e-01	7.0715e-01	7.0710e-01
$\pi/3$	5.0000e-01	5.0006e-01	4.9999e-01
$5\pi/12$	2.5882e-01	2.5888e-01	2.5881e-01
$\pi/2$	6.1232e-17	5.0538e-05	-3.5289e-06
$7\pi/12$	-2.5882e-01	-2.5878e-01	-2.5882e-01
$2\pi/3$	-5.0000e-01	-4.9999e-01	-5.0000e-01
$3\pi/4$	-7.0711e-01	-7.0711e-01	-7.0710e-01
$5\pi/6$	-8.6603e-01	-8.6605e-01	-8.6602e-01
$11\pi/12$	-9.6593e-01	-9.6595e-01	-9.6592e-01
$l^2(E,h)$		6.1699e-05	9.1201e-06
$l^\infty(E,h)$		5.7455e-05	7.9073e-06
CPU-time		1.927249	1.825943

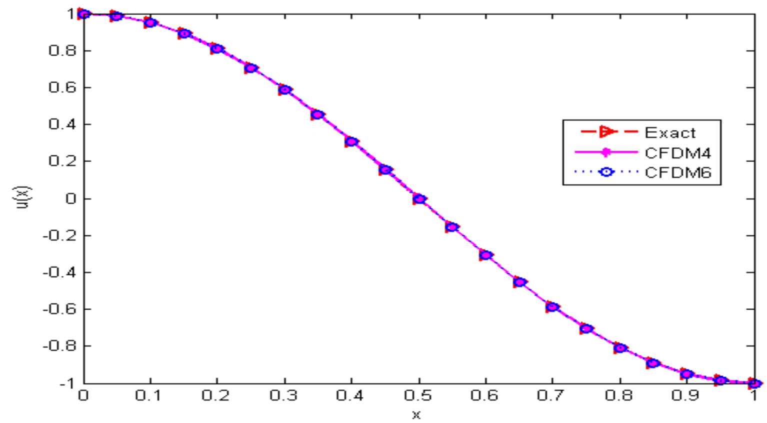

Figure 5. Exact and Approximate Solution of CFDM4 and CFDM6 for Example 3 with $N = 20$ and $h = 0.1571$

Table 8. Rate Convergence of CFDM4 and CFDM6 for $l^2(E, h)$ in Example 3

N	l^2 −CFDM4	Rate	l^2 −CFDM6	Rate
12	6.1699e-05		9.1201e-06	
24	4.1583e-06	3.8912	1.3713e-07	6.0554
48	2.6449e-07	3.9747	2.1224e-09	6.0137
96	1.6602e-08	3.9938	3.3054e-11	6.0047

Table 9. Rate Convergence of CFDM4 and CFDM6 for $l^\infty(E, h)$ in Example 3

N	l^∞ −CFDM4	Rate	l^∞ −CFDM6	Rate
12	5.7455e-05		7.9073e-06	
24	3.8721e-06	3.8912	1.2174e-07	6.0213
48	2.4621e-07	3.9752	1.8841e-09	6.0138
96	1.5456e-08	3.9937	2.9363e-11	6.0037

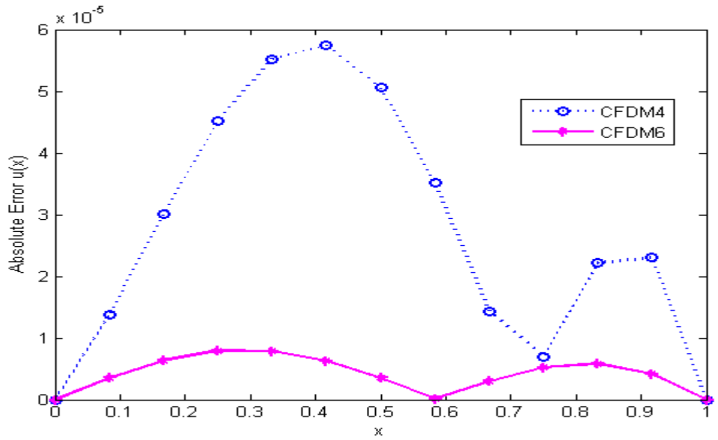

Figure 6. Comparison Absolute Error of CFDM4 and CFDM6 for Example 3 with $N = 12$ and $h = 0.3333$

To provide the summary of the proposed method to find the approximate solutions based on applying compact finite difference on FIDE of examples (1-3) that have been illustrated in Tables (1-9). The error norms of l^2 and l^∞ are reported in Tables (2,3,5,6,8 and 9) of the fourth order for space levels and compared with the results of the sixth order. From Tables (1-9) the results of the sixth order are better than the results from the fourth. One of the reasons is due to the errors produced by the sixth order scheme being much close to zero and the obtained numerical solutions indicate that the method is reliable and yields result compatible with analytical solutions. In addition, the scheme is shown that the fourth and sixth-orders converge in space.

Conclusion

In this paper, we proposed a robust and efficient numerical scheme for solving FIDE problems using a compact finite difference method based on fourth and sixth orders. The key idea of this research was to implement a combination of fourth and sixth orders, with the composite Boole's rule to solve FIDE, resulting in a highly accurate and computationally efficient numerical solution FIDE. Additionally, the accuracy of the proposed method is demonstrated by considering three test problems. The precision of the scheme has been measured by considering several test problems and calculating l^2 and l^∞ error norms for different space levels. From Tables (1-9) and Figures (1,3 and 5), numerical experiments demonstrated that the results that are obtained from the proposed method are efficient, reliable, fruitful, and powerful. Overall, the proposed method is a significant step forward in the field of solving FIDE problems. It offers a robust and efficient numerical approach that can achieve high levels of accuracy. In the future, this work can be solved by finite element methods for more details see [21-27].

Acknowledgment

Authors would like to thank of the financial support from Koya University.

Conflict of interest

The author has no conflict of interest.

Reference

1. H.-H. Cao, L.-B. Liu, Y. Zhang and S.-m. Fu, "A fourth-order method of the convection–diffusion equations with Neumann boundary conditions," Applied Mathematics and Computation, vol. 217, no. 22, pp. 9133-9141, 15 July 2011.
2. Y. Fu, Z. Tian and Y. Liu, "A Compact Exponential Scheme for Solving 1D Unsteady Convection-Diffusion Equation with Neumann Boundary Conditions," arXiv preprint arXiv:1805.05728, 2018 May 15.

3. C. Yao, Y. Zhang, J. Chen, X. Ling, K. Jing, Y. Lu and E. Fan, "Development of a fourth-order compact finite difference scheme for simulation of simulated-moving-bed process," Scientific reports, vol. 10, no. 1, pp. 1-13, 08 May 2020.
4. M. A. Pirdawood and Y. A. Sabawi, "High-order solution of Generalized Burgers–Fisher Equation using compact finite difference and DIRK methods," In Journal of Physics: Conference Series, vol. 1999, no. 1, p. 012088, 2021.
5. J. Biazar and M. B. Mehrlatifan, "A compact finite difference scheme for reaction-convection-diffusion equation," Chiang Mai Journal of Science, vol. 3, pp. 1559-1568, 2017.
6. Y. A. Sabawi, M. A. Pirdawood and M. l. Sadeeq, "A compact Fourth-Order Implicit-Explicit Runge-Kutta Type Method for Solving Diffusive Lotka–Volterra System," In Journal of Physics: Conference Series, vol. 1999, no. 1, p. 012103, 2021.
7. P. Roul, V. P. Goura and R. Agarwal, "A compact finite difference method for a general class of nonlinear singular boundary value problems with Neumann and Robin boundary conditionsApplied Mathematics and Computation 350," Applied Mathematics and Computation 350, pp. 283-304, 2019.
8. Y. Cai, J. Fu, J. Liu and T. Wang, "A fourth-order compact finite difference scheme for the quantum Zakharov system that perfectly inherits both mass and energy conservation," Applied Numerical Mathematics, vol. 178, pp. 1-24, 2022.
9. M. Nabavi, M. S. Kamran and J. Dargahi, "A new 9-point sixth-order accurate compact finite-difference method for the Helmholtz equation," Journal of Sound and Vibration 307, no. 3-5, pp. 972-982, 2007.
10. F. M. Okoro and A. E. Owoloko, "Compact finite difference schemes for Poisson equation using direct solver," Journal of Mathematics and Technology, vol. 3, pp. 2078-0257, 2010.
11. A. F. Soliman, A. M. A. El-Asyed and M. S. El-Azab, "Compact Finite Difference Schemes for Solving a Class of WeaklySingular Partial Integro-differential Equations," Mathematical Sciences Letters, vol. 1, no. 1, pp. 53-60, 2012.
12. A. F. Soliman and M. S. El-Azab, "Compact Finite Difference Schemes for Partial integro- differential Equations," American Academic & Scholarly Research Journal, vol. 4, no. 1, pp. 6-13, 2012.
13. J. Zhao and R. M. Corless, "Compact finite difference method for integro-differential equations," Applied mathematics and computation, vol. 177, no. 1, pp. 271-288, 2006.
14. E. Yusufoğlu, "Improved homotopy perturbation method for solving Fredholm type integro-differential equations," Chaos, Solitons & Fractals 41, no. 1, pp. 28-37, 2009.
15. Garba, B. Danladi and S. L. Bich, "On solving linear Fredholm integro-differential equations via finite difference-Simpson's approach," Malaya Journal of Matematik (MJM), vol. 8, no. 2, pp. 469-472, 2020.
16. R. I. Esa, "Approximate solution of Fredholm Integro Differential equation using Quadrature Formulas methods," International Journal of Scientific Research in Science, Engineering and Technology, vol. 9, no. 6, pp. 284-291, 2022.
17. Chen, Jian, M. He and Y. Huang, "A fast multiscale Galerkin method for solving second order linear Fredholm integro-differential equation with Dirichlet boundary conditions," Journal of Computational and Applied Mathematics 364, p. 112352, 2020.
18. Muthuvalu, M. Sundaram, E. Aruchunan, M. K. M. Ali, J. V. L. Chew, A. Sunarto, R. Lebelo and J. Sulaiman., "Complexity Reduction Approach for Solving Second Kind of Fredholm Integral Equations," Symmetry 14, vol. 14, no. 5, p. 1017, 2022.
19. Cakir, H. Guckir, F. Cakir and M. Cakir, "A novel numerical approach for Fredholm integro-differential equations," Computational Mathematics and Mathematical Physics, vol. 62, no. 12, pp. 2161-2171, 2022.
20. R. S. Esfandiari, Numerical methods for engineers and scientists using MATLAB®, Crc Press, 2017.
21. Y. A. Sabawi, A Posteriori Error Analysis in Finite Element Approximation for Fully Discrete Semilinear Parabolic Problems, 2021.
22. M. Sabawi, "A posteriori error analysis for semidiscrete semilinear parabolic problems," 2018 Al-Mansour International Conference on New Trends in Computing, Communication, and Information Technology (NTCCIT), pp. 58-61, 2018.
23. M. A. M. Sabawi, Discontinuous Galerkin timestepping for nonlinear parabolic problems, University of Leicester: PhD dissertation, 2018.
24. Y. Sabawi, "Posteriori Error bound For Fullydiscrete Semilinear Parabolic Integro-Differential equations," In Journal of Physics: Conference Series, vol. 1999, no. 1, pp. 012085,IOP Publishing, 2021.
25. Y. A. Sabawi, "Adaptive discontinuous Galerkin methods for interface problems," in PhD Thesis, University of Leicester, UK, 2017.
26. Y. A. Sabawi, "A Posteriori $ L_ {\infty}(H^{1}) $ Error Bound in Finite Element Approximation of Semdiscrete Semilinear Parabolic Problems," Baghdad, Iraq, IEEE, 2019, pp. 102-106.
27. Y. Sabawi, "Adaptive discontinuous Galerkin methods for interface problems," (Doctoral dissertation, University of Leicester), 2017.

الحل العددي لمعادلة فريدهولم التكاملية التفاضلية باستخدام طريقة الفروق المحدودة المدمجة عالية الترتيب

سورمي رسول صابر [1] و يونس عبد سبعاوي [2,1]* و محمد شامي حسو [1]

[1] قسم علوم الرياضيات، كلية العلوم والصحة، جامعة كويه، كوردستان العراق , العراق.
[2] قسم تربية الرياضيات، كلية التربية ، جامعة تشك الدوليه، كوردستان العراق , العراق

الخلاصة

الهدف من هذا العمل هو تقديم طريقة عددية لحل معادلة فريدهولم التكاملية التفاضلية. ويناقش هذا العمل استخدام طريقة الفروق المحدودة المدمجة من الرتبة الرابعة والسادسة بناءً على قاعدة Boole المركبة لحل .FIDE. دقة الحل للطرق المقترحة يتم حسابها من خلال المعايير l^{∞} l^2 and ويتم تقييم كفاءة الطريقة المقترحة في هذا العمل من خلال قيم وقت وحدة المعالجة المركزية الصغيرة . عامل مهم في الطرق المقترحة التي تؤدي بدوها إلى انخفاض في التكلفة العمليات الحسابية للطرق . ويعد هذا العمل تحسنًا كبيرًا مقارنة بالطرق التقليدية ، والتي غالبًا ما تكافح من اجل الحفاظ على مستويات دقة عالية من الحل. يظهر أن الطرق المقدمة هي الترتيب الرابع والسادس في الفضاء. العديد من الأمثلة العددية تم تقد يمها لتوضيح أداء الطريقة المقترحة. بشكل عام ، تعد الطريقة المقترحة خطوة مهمة إلى الأمام في مجال حل مشاكل .FIDE إنه يوفر نهجًا عدديًا قويًا وفعالًا قادرًا على تحقيق مستويات عالية من الدقة حيث يصعب الحصول على حلول دقيقة.