Numerical Solutions of Nonlinear Fredholm Integro-Differential Equations using Fifth Order Runge-kutta Method

Atefa J. Saleh, Luma J. Barghooth and Maan A. Rasheed

Department of Mathematics, College of Basic Education, Mustansiriyah University, Baghdad, Iraq

Email: maan.rasheed.edbs@uommustansiriyah.edu.iq

Abstract. This paper is considered with the numerical solution of a type of second-order nonlinear Fredholm integro-differential Equations (FIDs). The proposed scheme is based on Runge-kutta methods of order fifth. Moreover, two numerical examples are considered to show the capability and the accuracy of the proposed methods compared with other Runge-kutta methods of less order. Finally, we apply the least square errors (LSE) formula to make numerical comparisons between the numerical and the exact solutions. The obtained results show that the proposed method is superior and more accurate than Runge-kutta methods of orders two, three, and four.

Keywords: Fredholm Integro-Differential Equations, Range Kutta methods, lest square errors, Kernel function, Integrable function.

1. Introduction

The general second order non-linear Fredholm integro-differential equations (FIDEs) [1,2] takes the form:

\[ f''(t) = g(t) + \int_a^b k(t,s)f(s)\,ds ; \quad t \in I = [a,b] \]  

(1)

where, the continuous function \( k \) with respect to \( t, s \) and \( f \), is the kernel; the initial condition \( f(a) = f_0 \); the interval \( [a, b] \) is divided into equal \( n \)-subintervals; \( h = (b - a) / n \), \( t_0 = a \), \( t_n = b \), and \( t_j = a + jh \), \( j = 0, 1, 2, ..., n \).

In general, an integro-differential equations is equation involving one, or more unknown functions with both differentiation and integral operations on time [3, 4]:

\[ f^{(n)}(t) + \sum_{i=0}^{n-1} P_i(t) f^{(i)}(t) = g(t) + \lambda \int_a^b K(t,s)f(s)\,ds \]  

(2)

where \( K(t, s), \, g(t), \, f(t), \, P_i(t) (i = 0, 1, 2, ..., n - 1) \), are known functions; \( \lambda \) is a scalar parameter.

Integro-Differential Equations (IDE) have some applications in various fields such as engineering, science, industrial mathematical, control theory, financial mathematics and queuing theory, see [5,6].

It is clear that, unlike to the ordinary (partial) differential equations, in the integro-differential equations and integral equations, the unknown variable appears under the integral sign, so that, integro-differential equations are difficult to be solved analytically. Therefore, many authors have
developed some advanced methods for solving this type of problems numerically, such as Yildirim [7], Gachpazan [8], and Zarebnia [9]. Also, we refer to other references, such as [10-14].

In this work, we consider the numerical solutions for second-order nonlinear Fredholm Integro-Differential Equation (2), based on Runge-kutta methods of order fifth. The work aims to show the capability and the accuracy of the proposed method compared with other Runge-kutta methods of less order.

This paper is divided into six sections. In the second section, the reduction technique of IDE to IE is explained. Section three is devoted to the description of Runge-kutta methods with order two, three, four and Fifth. The proposed algorithm steps is stated in section four. In section five, two numerical examples are given to show the comparative results between Runge-kutta schemes of different orders, with the exact solution. In the last section, we present the results discussion and state some conclusions based on the obtained numerical results.

2. Reduction IDE to IE

Reduction of IDE to an integral equation can be used for analyzing variety types of (FIDEs).

Theorem 1. [15]. Let \( g, k \) be iterate \( L_2 \) integrable function on interval \([a, b] \); \( P_i \in C^n[a, b] \), then equation (2) becomes:

\[
\left[ D^n + \sum_{i=1}^{n-1} P_i(t)D^i \right] f(t) = g(t) + \int_a^b K(t,s)f(s)ds \quad t \in [a, b]
\]

with the initial condition \( f(a) = f_0, f'(a) = f_1, \ldots, f^{(n)}(a) = f_n \)

The above equation is reduced to linear (FIE) as follows:

\[
f(t) = G_n(t) + \int_a^b K_n(t,s)f(s)ds
\] (3)

where

\[
G_n(t) = \sum_{i=0}^{n-1} \frac{f_i t^i}{i!} + \frac{1}{(n - 1)!} \int_a^t (t - s)^{n-1} f(s)ds + \sum_{k=0}^{n-2} \sum_{j=k+1}^{n-1} \sum_{i=k+j}^{n-1} (-1)^k p_j^{(k)}(a) A_{k,n-i-1} \frac{f_i t^{k+j-n-k} t^i}{i!}
\]

and

\[
K_n(t,s) = \frac{1}{(n - 1)!} \int_a^s (t - z_1)^{n-1} k(z_1,s)dz_1 - \sum_{k=0}^{n-1} \sum_{j=0}^k (-1)^j p_j^{(k)}(s) B_{n-k-1,i} \frac{(t - s)^k}{k!}
\]

A and B are special constant matrices, of dimension (n-2x(n-2)) and (n-1x(n-1)), respectively.

3. Method Description

The kernel in equation (3) can be rewritten as:

\[
k[t,s,f(s)] = \sum_r u_r(t) v_r(s,f(s))
\]

It follows that

\[
f(t) = g(x) + \sum_r u_r(t) v_r(t) \quad t > 0
\] (4)
where
\[ Z_r(t) = v_r(t, f(s)), Z_r(0) = 0 \] \hspace{1cm} (5)

Applying the Runge-kutta forms to this system yields that
\[ Z_r(\alpha ph) = h \sum_{q=0}^{p-1} \beta_{pq} v_r(\alpha_q h, f(\alpha_q h)), p = 1,2,\ldots, m \] \hspace{1cm} (6)

where the parameter \( \{\alpha_p\} \) satisfy \( 0 = \alpha_0 \leq \ldots \leq \alpha_m = 1 \) and the weights \( \{\beta_{pq}\} \) satisfy
\[ \sum_{q=0}^{p-1} \beta_{pq} = \alpha_p, \ P = 1,2,\ldots, m \] \hspace{1cm} (7)

The number \( Z_r(h) \) is the required to be \( O(h^{m+1}) \) approximation to \( Z_r(h) \) for \( m \leq 5 \).

By substituting (6) in (4), it follows that
\[ f(\alpha ph) = g(\alpha ph) + \sum_r u_r(\alpha ph) Z_r(\alpha ph) \]
\[ f(\alpha ph) = g(\alpha ph) + h \sum_r u_r(\alpha ph) \sum_{q=0}^{p-1} \beta_{pq} v_r(\alpha_q h, f(\alpha_q h)) \]
\[ f(\alpha ph) = g(\alpha ph) + h \sum_{q=0}^{p-1} \beta_{pq} k[\alpha ph, \alpha_q h, f(\alpha_q h)] \] \hspace{1cm} (8)

We rewrite equation (1) as follows:
\[ f(\alpha ph) = g(\alpha ph) + \sum_{j=0}^{p-2} f_{\alpha p-1 h} K[\alpha ph, s, f(s)] ds + f_{\alpha p-1 h} K[\alpha ph, s, f(s)] ds \] \hspace{1cm} (9)

where \( \alpha_j h \leq s \leq \alpha_{j-1} \)

Thus
\[ f(n, p) = F_n(\alpha ph) + h \sum_{q=0}^{p-1} \beta_{pq} k[\alpha ph, \alpha_q h, f(\alpha_q h)] \]
\[ F_n(\alpha ph) = g(\alpha ph) + \sum_{j=0}^{p-1} h \left( \sum_{q=0}^{m} \beta_{pq} K[\alpha ph, \alpha_q h, f(\alpha_q h)] \right), \]
\[ p = 1,2,\ldots, m \] \hspace{1cm} (11)

With suitable choices for the weight (\( \beta_{pq} \)) and the parameter (\( \alpha_p \)), in equation (11), the type of scheme is obtained.

3.1. Second order Runge–kutta Scheme:

Set \( m = 2 \), the parameters (\( \alpha_p \)) and weight (\( \beta_{pq} \)) found from equation (7), are as follows:
\[ \alpha_0 = 0, \alpha_1 = 1/2, \alpha_2 = 1, \beta_{00} = 0, \beta_{10} = \beta_{20} = \beta_{21} = 1/2 \]

We define:
\[ f(n, 0) = f(n - 1, 2) \]
\[ f(n, 1) = F_n(t_{n+1}) + hK[t_{n+1}, t_n, f(n, 0)] \]
\[ f(n, 2) = F_n(t_{n+1}) + h/2[K[t_{n+1}, t_n, f(n, 0)] + K[t_{n+1}, t_{n+1}, f(n, 1)] \]

where
\[ F(t_n) = g(t_n) + h/2 \sum_{j=0}^{n-1} (K[t_n, t_j, f(j, 0)] + K[t_n, t_{j+1}, f(j, 1)]) \]

3.2. Third order Runge–kutta Scheme

Let
\[ \alpha_0 = 0, \alpha_1 = 1/2, \alpha_2 = \alpha_3 = 1, \beta_{10} = 1/2, \beta_{20} = -1, \beta_{21} = 2, \beta_{30} = \beta_{32} = 1/6 \]
With suitable choice for the weight ($\beta_{pq}$) and the parameter ($\alpha_p$) in equation (10), we obtain
\[
f(n, 0) = f(n - 1, 3) \\
f(n, 1) = F_n(t_{n+1}, \frac{1}{2}hK[t_{n+\frac{1}{2}}, f(n, 0)]) \\
f(n, 2) = F_n(t_{n+1}, h[-K[t_{n+1}, t_n, f(n, 0)] + 2K[t_{n+1}, t_{n+1/2}, f(n, 1)]] \\
f(n, 3) = F_n(t_{n+1}, \frac{1}{6}h[K[t_{n+1}, t_n, f(n, 0)] + 4K[t_{n+1}, t_{n+1/2}, f(n, 1)] + k[t_{n+1}, t_{n+1/2}, f(n, 2)])
\]
where
\[
F(t_n) = g(t_n) + \frac{h}{6}\sum_{j=0}^{n-1} (K[t, t_j, f(j, 0)] + 4K[t, t_{j+1/2}, f(j, 1)] + K[t, t_{j+1}, f(j, 2)])
\]

### 3.3. Fourth Order Runge–Kutta Scheme

Applying the common fourth order Runge–Kutta formula with $m = 4$ yields
\[
\begin{align*}
\alpha_0 &= 0, \alpha_1 = \frac{1}{2}, \alpha_2 = \frac{1}{2}, \alpha_3 = 1, \beta_1 = \frac{1}{2}, \beta_2 = 1/3, \beta_3 = 1/3 \\
\beta_4 &= 1/4, \beta_5 = 1/5, \beta_6 = 1/6, \beta_7 = 1/7 \\
\end{align*}
\]
By substituting the above values in equation (10), it follows that:
\[
f(n, 0) = f(n - 1, 4) \\
f(n, 1) = F_n(t_{n+1/2}, \frac{1}{2}hK[t_{n+1/2}, t_n, f(n, 0)]) \\
f(n, 2) = F_n(t_{n+1/2}, \frac{1}{2}h(K[t_{n+1/2}, t_{1/2}, f(n, 1)]) \\
f(n, 3) = F_n(t_{n+1}, hK[t_{n+1}, t_{n+1/2}, f(n, 2)] \\
f(n, 4) = \\

F_n(t_{n+1}) + \frac{h}{6}\sum_{j=0}^{n-1} 2K[t_{n+1}, t_{j+1/2}, f(j, 1)] + K[t_{n+1}, t_{j+1}, f(j, 2)]
\]

### 3.4. Fifth Order Runge–Kutta Scheme

By applying the common fifth order Runge–Kutta formula with $m = 4$, we obtain
\[
\begin{align*}
\alpha_0 &= 0, \alpha_1 = 1/3, \alpha_2 = 2/3, \alpha_3 = \frac{1}{3}, \alpha_4 = 1, \beta_1 = 1/3, \beta_2 = 1/3, \beta_3 = 1/3, \beta_4 = 1/3, \beta_5 = 1/3, \beta_6 = 1/3 \\
\beta_7 &= 1/4, \beta_8 = 1/5, \beta_9 = 1/6, \beta_{10} = 1/7 \\
\end{align*}
\]
Substituting these values in equation (10) yields that
\[
\begin{align*}
f(n, 0) &= f(n - 1, 5) \\
f(n, 1) &= f(n, 0) \\
f(n, 2) &= F_n(t_{n+1}, \frac{1}{3}h(K[t_{n+1}, t_n, f(n, 1)]) \\
f(n, 3) &= F_n(t_{n+1}, \frac{2}{3}h(-K[t_{n+1}, t_n, f(n, 1)] + K[t_{n+1}, t_{n+1/2}, f(n, 2)]) \\
f(n, 4) &= F_n(t_{n+1}, t_{n+1/3}, f(n, 1)) + K[t_{n+1}, t_{n+1/3}, f(n, 2)] + K[t_{n+1}, t_{n+1/3}, f(n, 3)]
\end{align*}
\]
\[ f(n,5) = F(n)(t_{n+1}) + \frac{h}{8} \left( K[t_{n+1},t_{n},f(n,1)] + 3K[t_{n+1},t_{n+1/3},f(n,2)] + 3K[t_{n+1},t_{n+2/3},f(n,3)] \right) \]

where

\[ F(t_n) = g(t_n) + \frac{h}{8} \sum_{j=1}^{n-1} \left( K[t,t_j,f(j,1)] + 3K[t,t_{j+1/3},f(j,2)] + 3K[t,t_{j+2/3},f(j,3)] + K[t,t_{j+1},f(j,4)] \right) \]

4. Algorithm Steps

We can summarize the algorithm-steps of using Runge-Kutta schemes to find the numerical solutions for equation (3) as follows:

Step 1: Choosing the order \(m\) (2,3,4,5).

Step 2: Putting \(f(0,0) = g(0)\) and \(f(n,0) = f(n-1,m); n = 1,2,3,4,..\)

Step 3: Setting \(F(t) = g(t)\)

Step 4: Computing \(F(t); n = 1,2,3,..\)

Step 5: Evaluating \(f(n,1),...,f(n,m); n = 0,1,2,3,4,..\)

Step 6: Resulting \(f(t_{n+1}) = f_{n}^{(m)} + O(h^{m})\)

5. Numerical Examples

Example 1. Consider the non-linear FIDE problem:

\[ f'(t) = g(t) + \int_{a}^{b(t)} \left[ k(t,s,f(s))' \right] ds \quad 0 \leq t \leq 1 \]

By reducing this FIDE to IE, we get

\[ f(t) = g(t) + \int_{a}^{b(t)} \left[ k(t,s,f(s)) \right] ds; \quad 0 \leq t \leq 1 \]

where

\[ g(t) = e^{-t} \]

\[ k(t,s,f(s)) = \exp(-(t-s)) * (f(s) + \exp(-f(s))) \]

\[ 0 \leq s \leq t, 0 \leq t \leq 1 \]

One can show that the exact solution of this problem takes the form

\[ f(t) = \log(t + \exp(1)) \]

Table 1 shows the comparative results between Runge-Kutta schemes of different orders, with the exact solution, for \(f(t)\) at \(t = t_i = i \cdot h\), \(h = 0.1\), \(i = 0,1,2,..,10\).

| \(t\)     | 2\(^{nd}\)-Order | 3\(^{rd}\)-Order | 4\(^{th}\)-Order | 5\(^{th}\)-Order | Exact     |
|----------|------------------|------------------|------------------|------------------|-----------|
| 0.000000 | 1.000000         | 1.000000         | 1.000000         | 1.000000         | 1.000000  |
| 0.100000 | 1.0360222        | 1.0360099        | 1.0361280        | 1.0361249        | 1.0369674 |
| 0.200000 | 1.0707226        | 1.0709920        | 1.0709955        | 1.0709950        | 1.070995  |
| 0.300000 | 1.1043844        | 1.1046830        | 1.1046860        | 1.1046866        | 1.1046877 |
| 0.400000 | 1.1386895        | 1.1372760        | 1.1372830        | 1.1372855        | 1.1372822 |
| 0.500000 | 1.1683247        | 1.168390         | 1.1688780        | 1.1688499        | 1.1688476 |
| 0.600000 | 1.1988119        | 1.1994370        | 1.1994480        | 1.1994477        | 1.1994471 |
| 0.700000 | 1.2283875        | 1.2291250        | 1.2291390        | 1.2291356        | 1.2292380 |
| 0.800000 | 1.2571054        | 1.2579880        | 1.2579740        | 1.2579728        | 1.2579728 |
| 0.900000 | 1.2850123        | 1.2859820        | 1.2859988        | 1.2859963        | 1.2889993 |
| 1.000000 | 1.3125274        | 1.3132430        | 1.3132918        | 1.3132674        | 1.3187617 |
| L.S.E    | 0.00000033        | 0.0000012        | 0.0000001        | 0.0000000        | 0.0000000  |
| R.T.     | 0:0:1:25          | 0:0:40           | 0:0:1:37         | 0:0:1:35         |           |
Example 2:
Consider non-linear FIDE problem:

\[ f'(t) = g(t) + \int_a^b [k(t,s)f(s)] \, ds \quad 0 \leq t \leq 1 \]

By reducing this FIDE to IE, we get

\[ f(t) = g(t) + \int_a^b [k(t,s)f(s)] \, ds \quad 0 \leq t \leq 1 \]

where

\[ g(t) = \frac{1}{2} - \frac{1}{2} e^{t^2} - t^2 \]

\[ k[t,s,f(s)] = (t-s) \exp(2ts + f(s)) \quad 0 \leq s \leq t, \quad 0 \leq s \leq 1 \]

One can show that the exact solution of this problem takes the form

\[ f(t) = -t^2 \]

Table 2 shows the comparative results between Runge–Kutta schemes of different orders, with the exact solution, for \( f(t) \) at \( t = t_i = i \, h, \quad h = 0.1, \quad i = 0, 1, 2, \ldots, 10 \).

| t      | 2\textsuperscript{nd} Order | 3\textsuperscript{rd} Order | 4\textsuperscript{th} Order | 5\textsuperscript{th} Order | Exact  |
|--------|------------------------------|-------------------------------|-------------------------------|-------------------------------|--------|
| 0.0000000 | 0.0000000 | 0.0000000 | 0.0000000 | 0.0000000 | 0.0000000 |
| 0.1000000 | -0.0105667 | -0.0067958 | -0.0100010 | -0.0101000 | -0.0101000 |
| 0.2000000 | -0.0400720 | -0.0324913 | -0.0399220 | -0.0400222 | -0.0421000 |
| 0.3000000 | -0.1602667 | -0.1577809 | -0.1599778 | -0.1600011 | -0.1634000 |
| 0.4000000 | -0.3604872 | -0.3522675 | -0.3599865 | -0.3601100 | -0.3611000 |
| 0.5000000 | -0.5505223 | -0.5467474 | -0.5499885 | -0.5501100 | -0.5521000 |
| 0.6000000 | -0.7405672 | -0.7352675 | -0.7409675 | -0.7411000 | -0.7431000 |
| 0.7000000 | -0.9305053 | -0.9265888 | -0.9396949 | -0.9401100 | -0.9433000 |
| 0.8000000 | -1.1204432 | -1.1177478 | -1.139651 | -0.1401100 | -0.1411000 |
| 0.9000000 | -1.3103854 | -1.3088332 | -1.320678 | -1.3214000 | -1.3222000 |
| 1.0000000 | -1.5003278 | -1.4991288 | -1.5199866 | -1.5204000 | -1.5222000 |

L.S.E

R.T.

0:0:1:33

6. Discussion and Conclusions
In this work, we consider the numerical solutions to a type of second-order nonlinear Fredholm Integro-Differential Equations. Firstly, we reduce the nonlinear (FIDEs) to (FIEs). Secondly, we construct the Runge–Kutta methods of order 2, 3, 4, and 5 to solve the considered problems. We apply the least square errors (LSE) formula to make numerical comparisons between the numerical and the exact solutions. The obtained results in Tables (1) and (2) show that the fifth-order method is superior and more accurate than other used methods.

Acknowledgments
The authors would like to thank Mustansiriyah University for supporting the present work.

References
[1] M. Safavi, A. A. Khajehnasiri, Numerical solution of nonlinear mixed Volterra-Fredholm integro-differential equations by two-dimensional block-pulse functions, Cogent Mathematics & Statistics 5: 1521084, 1-12, (2018)
[2] M. H. Daliri Birjandi, J. Saberi-Nadjafi, and A. Ghorbani, An efficient numerical method for a class of nonlinear Volterra integro-differential equations, Journal of Applied Mathematics, ID 7461058, 1-7, (2018)
[3] N. S. Christodoulou, An algorithm using Runge-Kutta methods of orders 4 and 5 for systems of ODEs, International Journal of Numerical Methods and Applications, 2(1), 47-57, (2009)
[4] Ali Filiz, Numerical solution of linear Volterra integro-differential equation using Runge-Kutta-Fehlberg method, Applied and Computational Mathematics, 3(1), 9-14, (2014)
[5] A. J. Saleh, R. E. Esa, and A. F. Jameel, Numerical treatment of non-linear Volterra integro-differential equation by using Runge-Kutta methods, AIP Conference Proceedings, 2138, 030034 (2019).
[6] S. Shoki, Numerical Solution of Linear Volterra Integro-differential Equations, M.Sc. thesis, University of Technology, Baghdad-Iraq, (2002)
[7] A. Yildirim, Solution of BVPs for fourth-order integro-differential equations by using Homotopy perturbation method, Computers and Mathematics with Applications, 56(12), 3175-3180, (2008)
[8] M. Gachpazan, Numerical scheme to solve integro-differential equations system, Journal of Advanced Research in Scientific Computing, 1(1), 11-21, (2009)
[9] M. Zarebnia, Z. Nikpour, Solution of linear Volterra integro-differential equations via Sinc functions, International Journal of Applied Mathematical and Computation, 2(1), 1-10, (2010)
[10] Ali Filiz, A fourth-order robust numerical method for integro-differential equations, Asian Journal of Fuzzy and Applied Mathematics, 1(1), 28-33, (2013)
[11] Ali Filiz, Numerical Method for a linear Volterra integro-differential equation with Cash-Karp method, Asian Journal of Fuzzy and Applied Mathematics, 2(1), 1-11, (2014)
[12] B. H. Lichae, J. Biazar, and Z. Ayati, A class of Runge–Kutta methods for nonlinear Volterra integral equations of the second kind with singular kernels, Advances in Difference Equations, 349, 1-19, (2018)
[13] H. O. Al-Humedi, A. F. Shoushan, A Combination of the orthogonal polynomials with least – squares method for solving high-orders Fredholm-Volterra integro-differential equations, Al-Qadisiyah Journal of Pure Science, 26(1), 20–38, (2021)
[14] S. M. Yassein, Reliable iterative method for solving Volterra - Fredholm integro differential equations, Al-Qadisiyah Journal of Pure Science, 26(2), 1-11, (2021)
[15] C.T. H. Baker, and G. F. Miller, Treatment of integral equations by numerical methods, Academic Press Wc. London, (1983)