مجلة كلية التربية الاساسية
كلية التربية الاساسية – الجامعة المستنصرية
Journal of the College of Basic Education Vol.32 (NO. 135) 2026, pp. 110-121

Multistep and Runga-Kutta Methods for Hammerstein-Fredholm-type Integral Equation

Atifa J. Saleh
Department of Mathematics, College of Basic Education ,
Mustansiriya University, Baghdad, Iraq
atifa_jalil.edbs@uomustansiriyah.edu.iq

Abstract:
The aim of this paper is to study some numerical methods for solving a special type of integral equation known as the Hammerstein-Fredholm-type integral equation (HFSKIE). The study focuses on the numerical solutions obtained through the multi-step method.

Keywords: Hammerstein-Fredholm-second kind integral equation, Newton-Gregory method, Fourth Runge-Kutta method, Repeated Simpson rule.

1. Introduction

It is abundantly clear that Fredholm equations and initial value problems for ordinary differential equations share a significant degree of affinity with one another. To be more specific, any problem of this nature can be recast as a set of Fredholm equations. The approaches that are used to solve the initial value problem are therefore of great assistance in identifying methods that can be used to deal with Fredholm FIE of second type equations. In this manner, we dealt with the linear FIE of the second class integral equation system.
Since last decades, many authors have studied the numerical solutions of Fredholm integral equations, see for instance [1-6].
The methods for solving Eq. (FIE) considered in this search fall one into one class, Runga-Kutta method. The Runge-kutta methods were used to find a numerical solution of (ODE), [7,8]. And it used to treat linear Fredholm integral equations [9] and [10] Al-Rawi [11] used this method to treat first kind I.E. of convolution type moreover Al-Nasir [12] used this method to approximate the integral of linear Fredholm integral equation of second kind. Since the last decades, various numerical methods for approximating the solutions of Hammerstein integral equations have been proposed. For Fredholm–Hammerstein integral equations, the classical method of successive approximations was introduced in [13]. A variation of the Nystrom method was presented in [14]. A collocation type method was developed in [15]. In [16], Brunner applied a collocation-type method to nonlinear Fredholm–Hammerstein integral equations and integro-differential equations, and discussed its connection with the iterated collocation method. Guoqiang [17,18] introduced and discussed the asymptotic error expansion of a collocation-type method for Fredholm–Hammerstein integral equations.

Recently, in [19], the numerical solutions of Fredholm-Hammerstein integral equation of the first kind are computed using a finite difference method decomposition with Nyström method.

2. Multi Step Method

Multi step method is derived by expressing the equation in sequence of equidistant points: $X_n = X_0 + nh, (X_0 = a \text{ and } X_n = b)$, ($h$ is fixed). And by approximating the integral term by some suitably chosen quadrature formula [8].

In this paper multi step methods including (Trapezoidal rule, repeated Simpson rule and Newton -Gregory) are used to solve Eq. (HF).

2.1. Newton-Gregory Method as quadrature rule

Newton-Gregory Method is one of quadrature rule, which could also be used to approximate the function on interval $[a, b]$ as follows:

Let T(f) represent the Trapezoidal rule approximation to

$$\int_a^b f(x)dx$$

At integrate h, $T(f) = \frac{1}{2}h\{f(a) + 2\sum_{i=1}^{n-i} f(a+ih) + f(b)\}$

$$\int_a^K f(x)dx = T(f) + \sum_{i=1}^{m} \frac{B_2}{(2i)!}h^2\{f^{(2,-1)}(a) - f^{(2,-1)}(b)\} + R_f$$

$$\int_a^b f(x)dx = T(f) + \frac{1}{12}h^2\{f'(a) - f'(b)\} - \frac{1}{720}h^4\{f^3(a) - f^3(b)\} + \cdots + \frac{B_{2m}}{(2m)!}h^{2m}\{f^{(2m-1)}(a) - f^{(2m-1)}(b)\} + R_f 0 \quad (1.1)$$

Where the reminder R is given by,

$$R_f \frac{B_{2m+2}}{(2m)!}h^{(2m+2)}(b-a)f^{(2m+2)}(\epsilon), \ldots (a \le \epsilon \le b).$$

And coefficient $(B_1 = -\frac{1}{2},\ B_2 = \frac{1}{6}, B_4 = -\frac{1}{30}, B_6 = \frac{1}{24}, B_8 = -1/30, B_{10} = 5/60,$ and $B_3 = B_5 = B_7 = 0)$ are called the Bernouli numbers [10].

The necessary condition for f(x) is that the derivative of the function with respect to height should be continuous.

Newton-Gregory is obtained by replacing the derivatives in eq. (1.1) by forward and backward differences. We get

$$\Delta f_0 = f(a+h) - f(a), \nabla f_n = f(b) - f(bb+h) \ldots etc$$

Write $f_0 = f(a), f_n = f(b)$, then we have:

$$\int_a^b f(x)dx = \frac{1}{12}h[5f(X_0) + 13f(x_1) + f(X_2) + \cdots + f(X_{n-2}) + 13f(X_{n-1}) + 5f(X_n)] \quad (1.2)$$

Which is called the composite Newton-Gregory formula [20]

2.2. "Applications of Integral and Differential Calculus for Civil Engineers"

The Fundamental Theorem of Calculus

Suppose that the function f is continuous on an interval I containing the point a.

PART I. Let the function F be defined on I by

$$F(x) = \int_a^x f(t)\,dt.$$

Then F is differentiable on I, and $F'(x) = f(x)$ there. Thus, F is an antiderivative of f on I:

$$\frac{d}{dx}\int_a^x f(t)\,dt = f(x).$$

PART II. If $G(x)$ is *any* antiderivative of $f(x)$ on I, so that $G'(x) = f(x)$ on I, then for any b in I we have

$$\int_a^b f(x)\,dx = G(b) - G(a).$$

3. Approximate solutions by multistep methods

The notation we will use is as follows: $f(rh)$ represents the value of the solution to equation (HF) at the position rh. f_r will provide an approximate solution at the same position. In general, let

$$\int_0^{rh} \emptyset(y)dx = h\sum_{j=0}^{r} W_{rj}\,\emptyset(jh) + E_r\,(\emptyset). \quad (1.3)$$

denotes a quadrature formula with an equal interval and a remainder Er, [12]. The weights (W_{rj}) are assumed to be given or selected. Using the assistance of equation (1.3) , then (HF) can be written as:

$$f(rh) = g_r + h\sum_{j=0}^{r} W_{rj}\,K[rh,jh]u\big(f(jh)\big) + E_r(K). \quad (1.4)$$

Therefore, assuming the error term $E_r(K)$ is insignificant or disregarded, we can derive the following set of equations. The objective is to find a reliable estimate for f at the quadrature points rh, $r = 1,2\ldots$

$$f_r = g_r + h\sum_{j=0}^{r} W_{rj}\,K[rh,jh]u\big(f(jh)\big), \qquad r = 1,2 \quad (1.5)$$

And

$$h = \frac{X_0 - X_n}{n}.$$

3.1. Day's staring procedure

One drawback of multi-step methods is the requirement for starting values, necessitating the use of a unique starting technique. If the kernel is suitably smooth, it may be feasible to obtain a power series representation for f in the vicinity of the origin, where the required initial values can be determined [9].

Day's develops a number of starting procedures, which are useful in a variety of circumstances. We give one of Day's methods here. The particular onewe describe given f_1, f_2, f_3 each with a truncation error of:

Define

$$f_{11} = g_1 + hk(h,0)u\big(g(X_0)\big), Where. f_0 = g(X_0)$$

$$f_{12} = g_1 + \frac{h}{2}\{K(h,0)u\big(g(X_0)\big) + K(h,h)u(f_{11})\}$$

Then,

$$f_1 = g_1 + \frac{h}{6}\{K(h,0)u\big(g(X_0)\big) + 4K\left(h,\frac{h}{2}\right)u(f_{13}) + k((h,h)u(f_{12})\}$$

Next let,

$$f_{21} = g_2 + 2hk(2h,0)u(f_1)$$

$$f_2 = g_2 + \frac{h}{3}\{k(2h,0)u\big(g(x_0)\big) + 4k(2h,h)u(f_1) + k(2h,2h)u\,(F_{21})\}$$

Finally, with

$$f_{31} = g_3 + \frac{3h}{2}\{k(3h,h)u(f_1) + k(3h,h)u(f_3)\}$$

We obtain,

$$f_3 = g_3 + \frac{3h}{8}\{k(3h,0)u\big(g(x_0)\big) + 3k(3h,h)u(f_1) + 3k(3h,2h)u\,(F_2) + k(3h,3h)u\,(f_3)\}$$

3.2. Trapezoidal rule

In the simplest case when eq. (1.3) is the Trapezoidal rule then eq. (1.5) have the simple form,

$$f_3 = g_r + \frac{h}{2}\sum_{j=v}^{r-1} W_{rj}\,K(X_r,t_j)u\left(f(t_j)\right) + w_r hk\,(x_r,t_j)u(f_{r1}) \qquad (1.6)$$

Where $r = 1,2,\ldots,n$.

Using Day's starting procedure to evaluate f_{r1} as follows:

$$f_{11} = g_1 + hk(x_1,x_0)u(g(t_0), Where..f_0 = g(t_0)$$

$$f_{r1} = g_r + \frac{r}{(r-1)}h\sum_{j=1}^{r-1} K(X_r,t_j)u\left(f(t_j)\right), \qquad r = 2,3,\ldots n$$

Where the weights are $w_{r0} = w_{rr} = 1$ $and\ w_{rj} = 2, (1 \le j \le r-1)$.

$w_1 = W = 1$ and $w_{rj} = 2, (1 \le j \le r-1)$

Algorithm of the method

- **Step 1:** fix $f_0 = g(x_0)$
- **Step 2:** calculate f_{11} and f_{r1}
- **Step 3:** evaluate f_r

3.3. Repeated Simpson's rule:

A convenient and simple continuation of Day's starting procedure can be based on Simpson's rule in the following manner when r is even we can use repeated Simpson's rule immediately to give:

$$f_r = g_r + \frac{h}{3}\sum_{j=0}^{r-1} W_{rj}\ K(X_r, t_j)u\left(f(t_j)\right) + \frac{h}{3}k\ \ (x_r, t_j)u(f_{r1}), r = 2,4, \dots \quad (1.7)$$

However, when r is odd a different strategy is required and mention the local truncation error of the 3/8 -this rule is used at the upper end to give.

$$f_r = g_r + \frac{h}{3}\sum_{j=0}^{r-3} W_{rj}\ K(X_r, t_j)u\left(f(t_j)\right) + \frac{3}{8}h\{k(x_r, t_{r-3)}\ u(f_{r-3}) + 3k(x_r, t_{r-2)}u(f(t_{r-2})) + 3k(x_r, t_{r-1})u(f(t_{r-1}) + k(x_r, t_r)u(f_r)) \quad (1.8)$$

But,

$$f_r = g_1 + \frac{h}{6}\left\{K(x_1, t_0)u(g(f_0)) + 4k\left(x_1, \frac{t_1}{2}\right)u(f_{13}) + k(X_1, t_1)u(f_{12})\right\} \quad (1.9)$$

If we go back to section (3-1) we can evaluate f_{11}, f_{12}, f_{13} and f_{r1}, $r = 1,2, \dots, n$, by Day's starting procedure.

Algorithm of the method

- **Step 1:** fix $f_0 = g(x_0)$.
- **Step 2:** If r is even then calculate f_r, by eq. (1. 7)
- **Step 3:** If r is odd then calculate f_1, f_2, f_3 and f_1 by Day's starting
- **Step 4:** evaluate f_r by Eq. (1. 8).

3.4. Newton-Gregory Method

Another type of multi-step method, Newton-Gregory is used to find the approximate solution of Eq. (HF), as follows:

Let $f_0 = g(t_0)$

Define,

$$f_{11} = g_1 + hk(x_1, 1_0)u(g(t_0)).$$

Then,

$$f_1 = g_1 + \frac{h}{2}\{K(x_1, t_0)u(g(t_0)) + k(x_1, t_1)u(f_{11})\},$$

and

$$f_{21} = g_2 + 2h\{K(x_2, t_{10})u(f_1)\}.$$

Then,

$$f_2 = g_2 + \frac{h}{3}\{K(x_2, t_0)u(f(t_0)) + 4k\ (x_2, t_1)u(f_1) + k(x_2, t_2)u(f_{21})\}.$$

Next

$$f_{r1} = g_r + \frac{r}{(r-1)}\ h\sum_{j=1}^{r-1} K(X_r, t_j)u(f_j); r = 3, 4, \dots$$

Hence

$$f_r = g_r + \frac{h}{12}\Big\{5K(x_r,t_0)u(g(t_0)) + 13k\,(x_r,t_1)u(f_1) + 12\sum_{j=2}^{r-2} K(X_r,t_j)\,u\big(f(t_j)\big) + 13K\,(x_r,t_{r-1})u(f(t_{r-1} + 5k(x_r,t_r)u(f_r)\Big\}$$

Algorithm of the method

- **Step 1:** fix $f_0 = g(x_0)$.
- **Step 2:** calculate f_{11}, f_1, f_{21}, f_2 and f_{r1}.
- **Step 3:** evaluate f_r

4. Runga-Kutta Methods

Runga-Kutta methods are widely used and extensively studied algorithms in the solution of ordinary differential equations. To develop these methods, one begins by formulating a general approach for computing the solution at a given point $x_n + h$ based on the solution at previous points .

First, we recall that Runga-Kutta method for initial value problem:

$$f'(x) = k(x,f(x)), f\,(0) = 0.$$

Can be written as the sequence of operations defined by,

$$f_{j+1} = f_i + h\sum_{j=0}^{P-1} A_{pj}\,K\,(s_i + \alpha_j h)\,u(f_{i+\alpha j}). \qquad (1.10)$$

Where the parameters $\{\alpha_j\}$ satisfy $0 = \alpha_1 \leq \alpha_2 \leq \cdots \leq \alpha_m = 1$ and the weights $\{A_{pj}\}$ satisfy,

$$\sum_{j=0}^{r-1} A_{rj} = \alpha_j, r = 1,2\,\ldots, P-1, \qquad (1.11)$$

With f_r an approximation to the solution at $x = x_r = a + rh$, [12].

Eq. (1.10) is called single -step Runga-Kutta equation. It is possible to generate Runga-Kutta method of higher order using (1.10) equation.

4.1. Solution of (HFSKIE) by using Runga-Kutta method

The method defined in eq. (1.10) can be extended to give a class of Runga-Kutta methods for the solution of equation (HF). Setting $x=x_1$ we have numerical integration:

$$f(x_i) = g(x_i) + \int_a^{a+ih} k(a+ih,t)u\,\big(f(t)\big)dt.\ \ i = 1,2,\ldots,n.$$

$$f(x_i) = g(x_i) + \sum_{j=0}^{i-1}\int_{a+ih}^{a+(i+1h)} k(a+ih,t)u\,\big(f(t)\big)dt.\, i = 1,2,\ldots,n. \qquad (1.12)$$

An approximation f_i *to* $f(x_i)$ can be determine from the following equation,

$$f_i = g(x_i) + h \sum_{j=0}^{i-1} \sum_{q=0}^{p-1} A_{pq}\, k\left(a + ih, a + \left(j + \alpha_q h\right)\right) u(f\left(t_j + \alpha_q\right). \quad (1.13)$$

For $\in (x_i, x_{i+1})$) we can write eq. (1.12) as follows:

$$f(x) = g(x_i) + \sum_{j=0}^{i-1} \int_{x_j}^{x_j+1} k(x,t)u\left(f(t)\right)dt + \int_{x_j}^{x} k(x,t)u\left(f(t)\right)dt. \quad (1.14)$$

Then set $x = x_i + \alpha_r h,\ r = 1,2, \dots, p-1$ and approximating the integral,

$$\int_{x_j}^{x_j+\alpha_r h} k(x_i + \alpha_r h, t)u\left(f(t)\right)dt = \sum_{q=0}^{r-1} A_{rq}\, k(x_i + \alpha_r h, x_i + \alpha_q h\,)\, u\left(f\left(t_{i+a_q}\right)\right). \quad (1.15)$$

Term in eq. (1.15) by:

We can apply the Runga-Kutta for (HFSKIE) of the from (1.12) as follows:

$$P(i,0) = f_1$$

$$P[i,r] = L_i(x_1 + \alpha_r h) + h \sum_{q=0}^{r-1} A_{pq}\, k\left(x_1 + \alpha_r h\, x_1 + \alpha_q h\right) u(P[i,q])\Big). \quad (1.16)$$

$$f_{1-ap} = L_1(x_1 + \alpha_p h) + h \sum_{q=0}^{r-1} A_{pq}\, k\left(x_1 + \alpha_p h,, x_1 + \alpha_q h\right) u(P[i,q])\Big). \quad (1.17)$$

Where,

$$L_1(X) = g(x) + \sum_{j=0}^{i-1} \sum_{q=0}^{p-1} A_{pq}\, k(x_1 + \alpha_p h, x_1 + \alpha_q h)u(P)u(P[i,q]). \quad (1.18)$$

And $L_0(X) = g(x),\ i = 0, 1, \dots, n - 1,\ r = 1, 2, \dots, p-1$ and $f(0) = g(0)$.

For suitable choice to the weight A_{pq}, and the parameter a in equations (1.16), (1.17) and (1.18) we get the following formula.

4.2. Runga-Kutta fourth order

The common fourth order Runga-Kutta method with m=4 is given by : $\alpha_0 = 0,\ \alpha_1 = \alpha_2 = i/2$ $, \alpha_3 = \alpha_4 = i\ , A_{10} = A_{21} = 1/2$ $, A_{20} = A_{30} = A_{31} = 0\ \ A_{32} = 1, A_{10} = A_{13} = 1/6\ , A_{41} = A - 42 = 1/3$

By substituting these values into equations. (1.16), (1.17) and (1.18) we have:

$$f_0 = g_0$$

$$P[i,0] = f_i$$

$$P[i,1] = Li\left(x_i + \frac{h}{2}\right) + \frac{h}{2}K\left(x_i + \frac{h}{2}, x_i\right)u\,(P[i,0])$$

$$P[i,1] = Li\left(x_i + \frac{h}{2}\right) + \frac{h}{2}\{K\left(x_1 + \frac{h}{2}, x_1 + \frac{h}{2}\right)u\,(P[i,0])$$

$$P[i,1] = Li\,(x_i + h) + h\,\{K\left(x_i + h, x_i + \frac{h}{2}\right)u\ \ (P[i,0])\}$$

$$f_{i-1} = L_1(x_1 + h) + \frac{h}{6}\ \{k(x_1 + h, x_1)u(P[i,0] + 2k\left(x_1 + h, x_1 + \frac{h}{2}\right)u(p[i,1]$$
$$+\ k\left(x_1 + h, x_1 + \frac{h}{2}\right)u(p[i,2] + k\,(x_1 + h, x_1 + h)u(p[i,3]$$

Where

$$Li\,(x) = g(x) + \frac{h}{6}\sum_{j=0}^{i-1}\ \{k(x, x_j)u(p[j,0]) + 2k\left(x, x_j + \frac{h}{2}\right)u(p[i,1]$$
$$+\ 2k\left(x, x_j + \frac{h}{2}\right)u(p[i,2]) + k(x, x_j + h2\ )u(p[i,3])$$

And $L_0(x) = g(x), i = 1,2,\dots,n$

4.3. Algorithm of the method:

- **Step 1:** fix $f_i = g(x_0)$
- **Step 2:** put $P[i,0]\ = f_i$
- **Step 3:** compute $L_i(x)$
- **Step 4:** compute $p[i,1], p[i,2], p[i,3]$.
- **Step 5:** evaluate f_{i+1}.

5. Test Examples

In this section, three numerical examples are given to show the accuracy of the proposed schemes: Multi step and fourth-order Runga-Kutta Methods (MRK). All the computational codes are written by MATLAB (R2020a) software.

Text example 1: Consider the problem:

$$f(x) = sin(x) + \int_0^1\ (x - t)sin(f(t))dt.$$

With exact solution,

$$f(x) = x.$$

This example was solved by running programs TRA, RE-SIM, NGR, and RK4. Table (1) is a summary of Numerical solution with exact solution and the least square errors (L.S.E) and running time (R.T.) for $h = 0.1$ and $x = x_i = a + ih$, $i = 0,1,2,\dots,10$.

Table (1): The results of example 1 by running TRA, RE-SIM, NGR, and RK4 programs.

X	TRA	RE-SIM	NGR	RK4	Exact $f(x) = x$
0.0	0.000000	0.000000	0.000000	0.000000	0.000000
0.1	0.099833	0.100000	0.100000	0.100000	0.100000
0.2	0.199497	0.200000	0.200000	0.200000	0.200000
0.3	0.399326	0.300000	0.29984	0.300000	0.300000
0.4	0.499151	0.400000	0.39984	0.400000	0.400000
0.5	0.598973	0.500002	0.49984	0.500000	0.500000
0.6	0.499151	0.600001	0.59984	0.600000	0.600000
0.7	0.698792	0.700002	0.69983	0.700000	0.700000
0.8	0.798606	0.800002	0.79983	0.799999	0.800000
0.9	0.898417	0.900003	0.89983	0.899999	0.900000
1	0.998225	1.000002	0.999983	0.999999	1.000000
L.S.E	0.000011	0.000000	0.000000	0.000000	-
R.T.	0:0:0:5	0:0:0:5	0:0:0:5	0:0:0:55	-

Test example 2: consider the problem:

$$f(x) = e^{-x} + \int_0^1 e^{-(x-t)}\left(f(t) + e^{-f(t)}\right)dt.$$

With exact solution

$$f(x) = ln\, ln\,(x + e)\ .$$

Test example 2 was solved by running program (MRK), Table (2) is a summary of numerical solution with exact solution and least square error (L.S.E) and running time (R.T) for $n = 10$ so that $h = 0.1$.

Table (2): The results of example 2 by running a program (MRK).

X	TRA	RE-SIM	NGR	RK4	Exact $In(x+e)$
0.000	1.000000	1.000000	1.000000	1.000000	1.000000
0.100	1.036028	1.035828	1.035828	1.036127	1.036127
0.200	1.071245	1.070959	1.70959	1.070995	1.070995
0.30	1.105052	1.104638	1.104766	1.104688	1.104688
0.400	1.137749	1.137212	1.137339	1.137282	1.137282
0.500	1.169406	1.168744	1.168876	1.168848	1.168848
0.600	1.200087	1.199320	1.199440	1.199447	1.199447
0.700	1.229849	1.228959	1.229089	1.229138	1.229138
0.800	1.258745	1.257766	1.257875	1.257973	1.257973
0.900	1.286825	1.285722	1.285747	1.285999	1.285999
1.000	1.314132	1.312955	1.313049	1.313262	1.313262
L.S.E	0.00000368	0.000000371	0.00000018	0.00000000	-
R. T.	0:0:0:97	0:0:0:97	0:0:0:97	0:0:1:15	-

Test example 3: consider the problem:

$$f(x) = x^2 - \frac{17}{30}x^6 + \int_0^1 (2x+t)\big(f(t)\big)^2 dt.$$

With exact solution

$$f(x) = x^2.$$

Test example 3 was solved by running program (MRK), Table (3) is a summary of numerical solution with exact solution and least square error (L.S.E) and running time (R.T) for $n = 10$ so that $h = 0.1$.

Table (3): The results of example 3 by running a program (MRK).

X	TRA	RE-SIM	NGR	RK4	Exact $In(x+e)$
0.000	0.000000	0.000000	0.000000	0.000000	0.000000
0.100	0.010001	0.010000	0.010000	0.010000	0.010000
0.200	0.040017	0.040002	0.040002	0.040000	0.040000
0.30	0.090085	0.090011	0.090035	0.090000	0.090000
0.400	0.160259	0.159998	0.160080	0.160000	0.160000
0.500	0.250583	0.249951	0.250109	0.250000	0.250000
0.600	0.361018	0.359715	0.359981	0.359999	0.360000
0.700	0.491292	0.488835	0.489278	0.489998	0.490000
0.800	0.640567	0.636628	0.636920	0.640000	0.640000
0.900	0.806624	0.799638	0.800199	0.810002	0.810000
1.000	0.983598	0.973569	0.972103	0.999970	1.000000
L.S.E	0.000283884	0.000818779	0.000884345	0.000000009	-
R. T.	0:0:0:10	0:0:0:10	0:0:0:10	0:0:1:00	-

6. Conclusions

Various numerical methods were employed to solve the Hammerstein-Fredholm - Second kind integral equation (HFSKIE) numerically. The numerical results were evaluated by comparing them to the matching precise solutions. Programs were meticulously coded, intricate flowcharts were meticulously crafted, complex examples were meticulously solved, and highly satisfactory outcomes were achieved. To conclude, the approaches employed in this thesis have demonstrated their efficacy in numerically solving (HFSKIE) and obtaining precise outcomes. Specifically, the multi-step and Runge-Kutta methods were utilized for solving (HFSKIE). The findings of this study indicate that employing the multi-step and Runge-Kutta approach yields very accurate and optimal results, as seen in the Tables that present a comparison of the outcomes achieved from solving test cases 1, 2, and 3, respectively.

Acknowledgements

The authors would like to thank Mustansiriyah University (www.uomustansiriyah.edu.iq) Baghdad-Iraq for supporting the present work.

References

1. Atifa J. Saleh, Luma J. Barghooth, Maan A. Rasheed, Numerical Solutions of Nonlinear Fredholm Integro Differential Equations using Fifth Order Runge-kutta Method, *Journal of Physics Conference Series*, 2021(012093):1-7.doi.org/10.1088/1742-6596/1999/1/0120

2. Rawaa I. Esa, Atefa J. Saleh, Numerical Treatment of First Order Volterra Integro-Differential Equation Using Non-Polynomial Spline Functions, *Iraqi Journal of Science*, (Special Issue) The First Conference of Mathematics , 2020.

3. Kherchouche, K., Bellour, A., & Lima, P. (2023). Numerical solution of nonlinear third-kind Volterra integral equations using an iterative collocation method. International Journal of Computer Mathematics, 100(11), 2063–2076. https://doi.org/10.1080/00207160.2023.2260007

4. Hongli Sun, Yanfei Lu, Numerical solutions to one dimensional linear Volterra–Fredholm integral equations based on LS-SVM model, *Journal of Computational and Applied Mathematics*, vol. 451, 2024.

5. Azin, H., Mohammadi, F. Approximate solution of Volterra-Fredholm integral equations using generalized barycentric rational interpolant. Appl. Math. J. Chin. Univ. 39, 220–238 (2024). https://doi.org/10.1007/s11766-024-3711-x

6. Iman A. Dhari and Muna M. Mustafa, Numerical Solution of Linear Volterra Integral Equation of the Second Kind with Delay Using Lagrange Polynomials. (2024). *Iraqi Journal of Science*, 65(3), 1541-1547. https://doi.org/10.24996/ijs.2024.65.3.30
7. Rice, J. R. (2014). *Numerical methods in software and analysis*. Elsevier.
8. Conte, S. D., & De Boor, C. (2017). *Elementary numerical analysis: an algorithmic approach*. Society for Industrial and Applied Mathematics.
9. Delves, L. M., & Walsh, J. (1974). Numerical solution of integral equations.
10. Baker, C. T., & Miller, G. F. (1982). Treatment of integral equations by numerical methods: based on the proceedings of a symposium held in Durham from 19-29 July, 1982, organized under auspices of the London Mathematical Society.
11. Al-Rawi, S. N. (1995). *Numerical Solution of First kind Integral Equations of Convolutions Type* (Doctoral dissertation, M. Sc. thesis, University of Technology).
12. Al-Nasir R.H.H.M., Numerical solution of Volterra Integral equations of the second kind, M.Sc. thesis, University of Technology, 1990
13. F.G. Tricomi, Integral Equations, Dover, 1982.
14. L.J. Lardy, A variation of Nystrom's method for Hammerstein equations, J. Integral Equations 3 (1982) 123–129.
15. S. Kumar, I.H. Sloan, A new collocation-type method for Hammerstein integral equations, J. Math. Comput. 48 (1987) 123–129.
16. H. Brunner, Implicitly linear collocation method for nonlinear Volterra equations, J. Appl. Num. Math. 9 (1982) 235–247.
17. H. Guoqiang, Asymptotic error expansion variation of a collocation method for Volterra–Hammerstein equations, J. Appl. Num. Math. 13 (1993) 357–369.
18. Y. Ordokhania, M. Razzaghi, Solution of nonlinear Volterra–Fredholm–Hammerstein integral equations via a collocation method and rationalized Haar functions, Applied Mathematics Letters 21 (2008) 4-9.
19. Somia Guechi, Numerical Solution of Volterra-Hammerstein Integral Equation of the First Kind by Finite Difference Method Decomposition with Nyström Method, *Mathematical Modelling of Engineering Problems*, 11(4), 2024, 1029-1036
20. Singh, S. P., Burry, J. H. W., & Watson, B. (Eds.). (2012). *Approximation theory and spline functions* (Vol. 136). Springer Science & Business Media.