Numerical Solution for System of Second Kind Fredholm Integral Equations by using Quadrature Scheme and HSKSOR Iteration

1N. Z. F. M. Radzuan*, 2M. N. Suardi and 3J. Sulaiman

1,2,3Mathematics with Economics Programme, Faculty of Science and Natural Resources, Universiti Malaysia Sabah, 88400 Kota Kinabalu, Sabah, Malaysia.
*Corresponding Author

Abstract

In this paper, the application of the Half-Sweep Kaud Successive Over Relaxation (HSKSOR) iterative method is extended to solve system of second kind Fredholm integral equations. The effectiveness HSKSOR method is examined by solving a linear system which is generated from the discretization of the system of second kind Fredholm integral equations. The formulation and implementation of the proposed methods are also presented. Some numerical simulations are carried out to show that the proposed method is superior compared to the standard methods.

Keywords: System of Fredholm integral equations; quadrature scheme; half-sweep Kaudd successive over relaxation iteration

1. INTRODUCTION

Integral equations (IEs) have been one of the principal mathematical models in different fields such as engineering, chemistry, physics and biology. The integral equations is often associated with the boundary value problem [1]. Therefore, many researchers give their focus in solving these equations.
Consider the following system of Fredholm integral equations of second kind

\[ c_r(x) = y_r(x) + \sum_{s=1}^{m} \int_{a}^{b} K_{rs}(x, t) c_s(t) \, dt \]  \hspace{1cm} (1)

where \( c_s(x) \) is an unknown function, \( K_{rs}(x, t) \) is a Kernel function, \( c_r(x) \) is a known function and \( y_r \) and \( K_{rs} \) are continuous functions \([2]\). Recently, many different methods have been proposed to approximate the solution of integral equations systems \([3, 4]\). Babolian et al. \([5]\) used Adomian decomposition method to obtain the solution of system (1). The Homotopy perturbation method \([6]\) and its modification \([7]\) were proposed by Javidi and Golbabai. The convergence analysis of Sinc-collocation method for approximating the solution of the integral equations system was proposed by Rashidinia and Zarebnia \([8]\). Maleknejad et al. \([9]\) presented a Taylor expansion method for a second kind Fredholm integral equation system with smooth or weakly singular kernel. Next, triangular functions method for the solution of system of Fredholm integral equations has been proposed by Almasieh and Roodaki \([10]\). Other methods being used to solve the problem (1) are Rationalized Haar functions method \([11]\), Block–Pulse functions method \([12]\), Expansion methods \([13]\), Decomposition method \([14]\), Orthogonal Triangular functions method \([15]\) and Bernstein collocation method \([16]\). The approximate solution of system (1) can be obtained by solving the resulting system of linear equations. The organization of the paper is as follows. In next section, the discretization of trapezoidal approximation equation via quadrature scheme. The latter section of this paper will discuss the formulation of the proposed iterative methods. Besides that, some numerical experiments is mentioned in Section 4 meanwhile the conclusion in the last section.

2. DISCRETIZATION OF TRAPEZOIDAL APPROXIMATION EQUATION

By considering numerical techniques, there are many methods that can be used to discretize the system of Fredholm integral equations into linear systems. This paper proposes the discretization of problem (1) by using the first order of quadrature scheme, namely trapezoidal rule to produce the quadrature approximation equation in order to generate system of linear equations. Prior to that, consider the quadrature scheme be defined as follows

\[ \int_{a}^{b} c(t) \, dt = \sum_{j=0}^{n} A_j \, c(t_j) + \varepsilon_n(c) \]  \hspace{1cm} (2)

where \( t_j \), \( A_j \) and \( \varepsilon_n(c) \) are the quadrature point in the interval \([a, b]\), weights quadrature and truncation error respectively \([17]\). By considering the trapezoidal rule which is the first order quadrature scheme, let the interval \([a, b]\) be divided into several sets \([x_0, x_1, x_2, ..., x_n]\) with the number of \((n)\) subintervals of equal width as shown in Figure 1. Meanwhile, Figure 2 shows the finite grid networks in order to form the full- and half-sweep quadrature approximation equations.
Based on Figure 2, both full- and half-sweep iterations will compute the approximate values onto each interior node points of type ● only until the iterative convergence is achieved. Then, other approximate solutions at remaining points (points of type ○) are computed using the direct method [17, 18]. By imposing equation (2) into equation (1) and neglecting the error, $\varepsilon_n(y)$, a system of linear equations can be formed to obtain the approximate solutions of $c(t)$ at the nodes $x_0, x_1, \ldots, x_n$. To do this, we need to derive trapezoidal approximation equation for a system of integral equations defined as follows

$$c_r(x_i) - \sum_{s=1}^{m} A_j K_{rs}(x_i, t_j)c_{s,j} = y_r(x_i)$$  \hspace{1cm} (3)
From equation (3), the weights quadrature coefficient, $A_j$ in equation (2) can be defined as

$$A_j = \begin{cases} \frac{1}{2} gh, & j = 0, n \\ gh, & j = \text{otherwise} \end{cases}$$

where the constant step size, $h$ is defined as follows

$$h = \frac{b - a}{n}$$

and $n$ is the number of subintervals in the interval $[a, b]$. Meanwhile, the value of $g$ corresponds to 1 and 2 which represents the full- and half-sweep cases respectively [17, 19].

By considering the approximate equation (3) at each point, $x_i, i = 0, 1, 2, \ldots, n$, the following generated linear system can be easily shown in matrix form as follows

$$Kc = y$$

where

$$K = \begin{bmatrix} 1 - A_0 K_{rs}(x_0, t_0) & -A_j K_{rs}(x_0, t_1) & \cdots & -A_j K_{rs}(x_0, t_n) \\ -A_j K_{rs}(x_1, t_0) & 1 - A_j K_{rs}(x_1, t_1) & \cdots & -A_j K_{rs}(x_1, t_n) \\ \vdots & \vdots & \ddots & \vdots \\ -A_j K_{rs}(x_n, t_0) & -A_j K_{rs}(x_n, t_1) & \cdots & 1 - A_j K_{rs}(x_n, t_n) \end{bmatrix},$$

$$c = [c_r(x_0) \ c_r(x_1) \ \cdots \ c_r(x_n)]^T,$$

$$y = [y_r(x_0) \ y_r(x_1) \ \cdots \ y_r(x_n)]^T.$$  

Evidently, $K$, $c$ and $y$ are known as the coefficient matrix of the linear system, unknown vector and known vector respectively.

### 3. FORMULATION OF ITERATIVE METHODS

By referring to the system of linear equations in equation (4), this linear system will be solved iteratively by using Full-Sweep Gauss-Seidel (FSGS), Full-Sweep KSOR (FSKSOR) and HSKSOR iterative methods. These three iterative methods can be classified as a family of point iterative methods.

#### 3.1 GS Iteration Scheme

Since implementations of these three iterative methods based on the point iteration approach, we need to consider again the coefficient matrix, $K$ in equation (4). As we
Numerical Solution for System of Second Kind Fredholm Integral Equations...

know, the coefficients matrix, $K$ of the linear system (4) can be manipulated to derive for the formulation of different iterative methods. To do this, let the coefficient matrix, $K$ of the system of linear equations (4) be decomposed as

$$K = L + D + U$$

(5)

where $U$, $L$ and $D$ is an upper triangular matrix, lower triangular matrix and a diagonal matrix. By using the definition of equation (5), the linear systems (4) can be rewritten

$$(L + D + U)c = y$$

(6)

By referring to the linear system (6), the general scheme of the GS iterative method can be stated in matrix form as

$$c^{(k+1)} = (D - L)^{-1}Uc^{(k)} + (L - D)^{-1}y$$

(7)

Or it can be shown that the formulation of this iteration scheme can be identified in point iteratively as

$$c_i^{(k+1)} = \frac{1}{K_{ii}} \left( y_i - \sum_{j=1}^{i-1} K_{ij} c_j^{(k+1)} - \sum_{j=i+1}^{n} K_{ij} c_j^{(k)} \right)$$

(8)

By referring to Figure 1 and equation (8), Algorithm 1 shows the implementation of FSGS iterative method.

**Algorithm 1: FSGS scheme**

i. Set initial value $c^{(0)} = 0$.

ii. Calculate the coefficient matrix, $K$.

iii. Calculate the vector, $y$.

iv. For $i = 0,1, ..., n$, calculate

$$c_i^{(k+1)} = \frac{1}{K_{ii}} \left( y_i - \sum_{j=1}^{i-1} K_{ij} c_j^{(k+1)} - \sum_{j=i+1}^{n} K_{ij} c_j^{(k)} \right)$$

v. Check the convergence test, $|c_i^{(k+1)} - c_i^{(k)}| < \varepsilon = 10^{-10}$. If yes, go to step (vi). Otherwise go back to step (iv).

vi. Display numerical solution.

3.2. KSOR Iteration Scheme

The KSOR iterative method is a new variant of SOR method that was introduced by Youssef [20]. The advantage of this method is enables to update the first component in
the first equation of the first step that reflects the rapid convergence at the beginning [20]. Relaxation parameter, \( \omega \) for KSOR method has less sensitivity than the SOR [21]. The general formula of the KSOR iteration scheme can be stated as [20]

\[
c_i^{(k+1)} = c_i^{(k)} + \frac{\omega}{K_{ii}} \left( y_i - \sum_{j=1}^{i-1} K_{ij} c_j^{(k+1)} - \sum_{j=i+1}^{n} K_{ij} c_j^{(k)} - K_{ij} c_j^{(k+1)} \right)
\]  

(9)

According to equation (9) and Figures 1 and 2, the implementation of FSKSOR and HSKSOR iteration schemes may be elaborated in Algorithm 2.

**Algorithm 2:** FSKSOR and HSKSOR schemes

i. Set initial value \( c^{(0)} = 0 \).

ii. Calculate the coefficient matrix, \( K \).

iii. Calculate the vector, \( y \).

iv. For \( i = 0, g, 2g, \ldots, n - g, n \) and \( j = 0, g, 2g, \ldots, n - g, n \) calculate

\[
c_i^{(k+1)} = c_i^{(k)} + \frac{\omega}{K_{ii}} \left( y_i - \sum_{j=g}^{i-g} K_{ij} c_j^{(k+1)} - \sum_{j=i+g}^{n} K_{ij} c_j^{(k)} - K_{ij} c_j^{(k+1)} \right)
\]

v. Check the convergence test, \( |c_i^{(k+1)} - c_i^{(k)}| < \varepsilon = 10^{-10} \). If yes, go to step (vi).

Otherwise go back to step (iv).

vi. Display numerical solution.

**4. NUMERICAL EXPERIMENT**

In order to analyze the effectiveness of the three proposed iterative methods, several numerical tests were conducted. For numerical comparison, the FSGS method acts as the control method. Three criteria will be considered in comparison such as number of iteration (Iter), time of iteration in seconds (Time) and maximum absolute error (Error). In addition, convergence test for the implementation of the iterative methods considered the tolerance error, \( \varepsilon = 10^{-10} \) in various grid sizes.

A) Problem 1 [16]

\[
c_1(x) = \frac{2}{3} e^x - \frac{1}{4} + \int_0^1 \left( \frac{1}{3} e^x t c_1(t) + t^2 c_2(t) \right) dt
\]  

(10a)

\[
c_2(x) = \frac{3}{2} x - x^2 + \int_0^1 (x^2 e^{-t} c_1(t) - xc_2(t)) dt
\]  

(10b)
The exact solutions for the system of Fredholm integral equations (10) are \( c_1(x) = e^x \) and \( c_2(x) = x \).

B) Problem 2 [16]
\[
c_1(x) = \frac{x}{18} + \frac{17}{36} + \int_0^1 \frac{(x+t)}{3} (c_1(t) + c_2(t))dt
\tag{11a}
\]
\[
c_2(x) = x^2 - \frac{19}{12} x + 1 + \int_0^1 xt (c_1(t) + c_2(t))dt
\tag{11b}
\]

The exact solutions for the system of Fredholm integral equations (11) are \( c_1(x) = x + 1 \) and \( c_2(x) = x^2 + 1 \).

C) Problem 3 [2]
\[
c_1(x) = \frac{5}{6} x^2 - \frac{25}{12} x + 1 + \int_0^1 x(1+t)c_1(t)dt + \int_0^1 x^2 tc_2(t)dt
\tag{12a}
\]
\[
c_2(x) = x^4 - \frac{1}{5} x^2 - \frac{7}{12} x + \int_0^1 xt c_1(t)dt + \int_0^1 (x^2 - xt)c_2(t)dt
\tag{12b}
\]

The exact solutions for the system of Fredholm integral equations (12) are \( c_1(x) = x^2 + 1 \) and \( c_2(x) = x^4 \).

Table 1. Comparison of number of iterations (Iter), execution time in seconds (Time) and maximum absolute error (Error) on iterative methods for Problem 1.

<table>
<thead>
<tr>
<th>Iter</th>
<th>M</th>
<th>FSGS</th>
<th>FSKSOR</th>
<th>HSKSOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>800</td>
<td>19</td>
<td>14</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>1200</td>
<td>19</td>
<td>14</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>1600</td>
<td>19</td>
<td>14</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>2000</td>
<td>19</td>
<td>14</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>2400</td>
<td>19</td>
<td>14</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Time (second)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>800</td>
<td>0.21</td>
<td>0.13</td>
<td>0.03</td>
<td></td>
</tr>
<tr>
<td>1200</td>
<td>0.42</td>
<td>0.24</td>
<td>0.08</td>
<td></td>
</tr>
<tr>
<td>1600</td>
<td>0.73</td>
<td>0.32</td>
<td>0.14</td>
<td></td>
</tr>
<tr>
<td>2000</td>
<td>1.15</td>
<td>0.67</td>
<td>0.23</td>
<td></td>
</tr>
<tr>
<td>2400</td>
<td>1.64</td>
<td>0.94</td>
<td>0.31</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Error</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>800</td>
<td>2.1425e-03</td>
<td>2.1425e-03</td>
<td>4.2708e-03</td>
<td></td>
</tr>
<tr>
<td>1200</td>
<td>1.4236e-03</td>
<td>1.4236e-03</td>
<td>2.8476e-03</td>
<td></td>
</tr>
<tr>
<td>1600</td>
<td>1.0748e-03</td>
<td>1.0748e-03</td>
<td>2.1358e-03</td>
<td></td>
</tr>
<tr>
<td>2000</td>
<td>8.5461e-04</td>
<td>8.5461e-04</td>
<td>1.7087e-03</td>
<td></td>
</tr>
<tr>
<td>2400</td>
<td>7.1277e-04</td>
<td>7.1277e-04</td>
<td>1.4239e-03</td>
<td></td>
</tr>
</tbody>
</table>
Table 2. Comparison of number of iterations (Iter), execution time in seconds (Time) and maximum absolute error (Error) on iterative methods for Problem 2.

<table>
<thead>
<tr>
<th></th>
<th>M</th>
<th>FSGS</th>
<th>FSKSOR</th>
<th>HSKSOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>800</td>
<td>33</td>
<td>17</td>
<td>17</td>
<td></td>
</tr>
<tr>
<td>1200</td>
<td>33</td>
<td>17</td>
<td>17</td>
<td></td>
</tr>
<tr>
<td>1600</td>
<td>33</td>
<td>17</td>
<td>17</td>
<td></td>
</tr>
<tr>
<td>2000</td>
<td>33</td>
<td>17</td>
<td>17</td>
<td></td>
</tr>
<tr>
<td>2400</td>
<td>33</td>
<td>17</td>
<td>17</td>
<td></td>
</tr>
<tr>
<td><strong>Time (second)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>800</td>
<td>0.49</td>
<td>0.14</td>
<td>0.06</td>
<td></td>
</tr>
<tr>
<td>1200</td>
<td>1.09</td>
<td>0.32</td>
<td>0.10</td>
<td></td>
</tr>
<tr>
<td>1600</td>
<td>1.93</td>
<td>0.98</td>
<td>0.18</td>
<td></td>
</tr>
<tr>
<td>2000</td>
<td>2.98</td>
<td>1.53</td>
<td>0.27</td>
<td></td>
</tr>
<tr>
<td>2400</td>
<td>4.29</td>
<td>2.21</td>
<td>0.40</td>
<td></td>
</tr>
<tr>
<td><strong>Error</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>800</td>
<td>2.0349e-03</td>
<td>2.0349e-03</td>
<td>4.0687e-03</td>
<td></td>
</tr>
<tr>
<td>1200</td>
<td>1.3568e-03</td>
<td>1.3568e-03</td>
<td>2.7131e-03</td>
<td></td>
</tr>
<tr>
<td>1600</td>
<td>1.0177e-03</td>
<td>1.0177e-03</td>
<td>2.0350e-03</td>
<td></td>
</tr>
<tr>
<td>2000</td>
<td>8.1416e-04</td>
<td>8.1416e-04</td>
<td>1.6281e-03</td>
<td></td>
</tr>
<tr>
<td>2400</td>
<td>6.7848e-04</td>
<td>6.7848e-04</td>
<td>1.3568e-03</td>
<td></td>
</tr>
</tbody>
</table>

Table 3. Comparison of number of iterations (Iter), execution time in seconds (Time) and maximum absolute error (Error) on iterative methods for Problem 3.

<table>
<thead>
<tr>
<th></th>
<th>M</th>
<th>FSGS</th>
<th>FSKSOR</th>
<th>HSKSOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>800</td>
<td>134</td>
<td>41</td>
<td>41</td>
<td></td>
</tr>
<tr>
<td>1200</td>
<td>134</td>
<td>41</td>
<td>41</td>
<td></td>
</tr>
<tr>
<td>1600</td>
<td>134</td>
<td>41</td>
<td>41</td>
<td></td>
</tr>
<tr>
<td>2000</td>
<td>134</td>
<td>41</td>
<td>41</td>
<td></td>
</tr>
<tr>
<td>2400</td>
<td>134</td>
<td>41</td>
<td>41</td>
<td></td>
</tr>
<tr>
<td><strong>Time (second)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>800</td>
<td>1.21</td>
<td>0.37</td>
<td>0.12</td>
<td></td>
</tr>
<tr>
<td>1200</td>
<td>2.70</td>
<td>0.79</td>
<td>0.25</td>
<td></td>
</tr>
<tr>
<td>1600</td>
<td>4.62</td>
<td>1.38</td>
<td>0.44</td>
<td></td>
</tr>
<tr>
<td>2000</td>
<td>7.12</td>
<td>2.15</td>
<td>0.68</td>
<td></td>
</tr>
<tr>
<td>2400</td>
<td>10.14</td>
<td>3.15</td>
<td>0.97</td>
<td></td>
</tr>
</tbody>
</table>
According to the three considered systems of Fredholm integral equations of second kind in equations (10), (11) and (12), all the results of numerical experiments were recorded in Tables 1, 2 and 3. The numerical results showed that the FSKSOR and HSKSOR iterative methods have reduced the number of iteration approximately 26.32%-69.40% as compared to FSGS method in solving three considered problems. In term of execution time, the FSKSOR iteration has reduced by approximately 38.10%-71.43% whereas the HSKSOR iteration has reduced approximately 80.00%-90.94% compared to the FSGS method. Based on the numerical results, it seems clearly the HSKSOR iteration has the fastest time compared to FSGS and FSKSOR iteration at various grid sizes. Therefore, it can be concluded that the HSKSOR iterative method is more efficient than FSGS and FSKSOR iterations in terms of execution time and complexity.

**CONCLUSION**

In this paper, HSKSOR method has been successfully applied in solving system of second kind Fredholm integral equations. Initially, this problem have been discretized by using the trapezoidal rule to derive the corresponding trapezoidal approximation equation. The linear system generated from this trapezoidal approximation equation has been solved iteratively via FSGS, FSKSOR and HSKSOR iterative methods. By referring Tables 1, 2 and 3 and Figures 1 and 2, the numerical results show that implementation of the HSKSOR method solved the three test problems with fastest execution time. However, the number of iteration of HSKSOR method is similar as FSKSOR method but these two iterations have less number of iteration compared to FSGS method. In terms of accuracy, numerical solutions obtained via HSKSOR method are in good agreement compared to the FSGS and FSKSOR methods. Shortly, it can be summarized that the HSKSOR method is superior to FSGS and FSKSOR methods, especially in the aspect of execution time. Overall, since this paper just considered HSKSOR iterative methods, future study can be extended to investigate the proposed approximate solutions through modified proposed iterative method, MKSOR [22] and block iterative methods as discussed in EDG [18, 23], EG [24, 25], MEG [26] and AGE [27].
ACKNOWLEDGEMENT

The research is fully supported by UMSGreat, GUG0138-1/2017. The authors fully acknowledged Ministry of Higher Education (MOHE) and Universiti Malaysia Sabah for the approved fund which makes this important research viable and effective.

REFERENCES


