Comparison of Closed Repeated Newton-Cotes Quadrature Schemes with Half-Sweep Iteration Concept in Solving Linear Fredholm Integro-Differential Equations

Elayara Aruchunan¹, Jumat Sulaiman²
¹School of Engineering and Science, Curtin University, 98009 Miri, Sarawak Malaysia
²Mathematics with Economics Programme, University Malaysia Sabah, 88999 Kota Kinabalu, Sabah, Malaysia
¹elayarah@yahoo.com, ²jumat@ums.edu.my

Abstract- The purpose of this paper is to apply half-sweep iteration concept with Gauss-Seidel (GS) iterative method namely Half-Sweep Gauss-Seidel (HSGS) method for solving high order closed repeated Newton-Cotes (CRNC) quadrature approximation equations associated with numerical solution of linear Fredholm integro-differential equations. Two different order of CRNC i.e. repeated Simpson’s $\frac{1}{3}$ and repeated Simpson’s $\frac{3}{8}$ schemes are considered in this research work. The formulation the implementation the proposed methods are explained. In addition, several numerical simulations and computational complexity analysis were carried out to authenticate the performance of the methods. The findings show that the HSGS iteration method is superior to the standard GS method. As well the high order CRNC quadrature schemes produced more precise approximation solution compared to repeated trapezoidal scheme.

Keywords- Linear Fredholm integro-differential equations, Newton-Cotes Closed Quadrature, central difference, Half-Sweep Gauss-Seidel.

I. INTRODUCTION

In this paper we focus on numerical solutions for first and second order Fredholm types of linear integro-differential equations. Generally, linear Fredholm integro-differential equations (LFIDEs) can be defined as follows

$$D^n + \sum_{i=0}^{n-1} P_i(x)D^i \int_a^b K(x,t)y(t)dt = g(x) \quad x,t \in [a,b]$$  \hspace{1cm} (1)

with most general boundary condition,

$$\sum_{j=0}^{n-1} r_j D^j y(x) \bigg|_{x=a} + r_{n+1} D^n y(x) \bigg|_{x=v} = a_j$$

where $K(x,t)$, $g(x)$, $P(x)$ for $i=0,...,n-1$ are known functions, $y(x)$ is the unknown function to be determined and $D^i y(x)$ denote the $i^{th}$ derivative of $y(x)$ with respect to $x$.

The linear LFIDEs occur in multiple diversified physical phenomena such as physical biology and engineering problems. Therefore numerical treatment is preferred in order to diagnose and solve the problems. In many application areas, it is necessary to use the numerical approach to obtain an approximation solution for the (1) such as finite difference-Gauss [1] Taylor collocation [2], Lagrange interpolation [3] and Taylor polynomial [4] and rationalized Haar functions [5] Tau [6]. Subsequently, generated system of linear equation has been solved by using iterative methods such as Conjugate Gradient [7], GMRES [8]. Based on extension work from [9], in this paper, discretization scheme based on family of closed repeated Newton-Cotes (CRNC) quadrature namely repeated Simpson $\frac{1}{3}$ (RS1) and repeated Simpson’s $\frac{3}{8}$ (RS2) along with finite difference schemes will be implemented to discretize (1). Then the generated linear system will be solved by using Half-Sweep Gauss-Seidel (HSGS) iterative method.

Actually, the HSGS represents combination of half-sweep iteration concept with standard Gauss-Seidel (GS) method. The standard GS method is also known as Full-Sweep Gauss Seidel (FSGS) method. The concept of the half-sweep iteration method has been introduced by Abdullah [10] via the Explicit Decoupled Group (EDG) iterative method to solve two-dimensional Poisson equation. Half-sweep iteration concept is also known as the complexity reduction approach [11]. Following that, the application of the half-sweep iteration concept with the iterative methods has been extensively studied by many researchers; see [12-14].
The rest of this work is organized as follows. In Section II, the derivation of the approximation equation is elaborated. In section III formulation of the FSGS and HSGS iterative methods are shown. Meanwhile, some numerical results are illustrated in Section IV to assert the effectiveness of the proposed methods and concluding remarks are given in Section V.

II. APPROXIMATION EQUATIONS

Figure 1: a) and b) show distribution of uniform node points for the full- and half-sweep cases respectively. The full- and half-sweep iteration concept will compute approximate values onto node points of type ● only until the convergence criterion is reached. Then other approximate solutions at the remaining points (points of the different type ○) can be computed using the direct method [10, 11, 12 and 14].

\[ A_j = \begin{cases} \frac{1}{2} ph, & j = 0, n \\ ph, & \text{otherwise} \end{cases} \]  
\[ A_j = \begin{cases} 1 \text{ ph,} & j = 0, n \\ \frac{4}{3} ph, & j = p, 3p, 5p, \ldots, n - p \\ \frac{2}{3} ph, & \text{otherwise} \end{cases} \]

where the constants step size, \( h \) is defined as

\[ h = \frac{b - a}{n} \]  
\( n \) is the number of subintervals in the interval \([a, b]\) and then consider the discrete set of points be given as \( x_i = a + ih \). The value of \( \rho \) which is corresponds to 1 and 2, represents the full- and half-sweep cases respectively.

B. Derivation of the Half-Sweep Finite Difference Schemes

In solving first order LFIDEs, differential part will be approximated by second order accuracy of first order derivative of finite difference scheme given by

\[ y'(x_i) = \frac{y(x_{i+1}) - y(x_{i-1})}{2h} + O(h^2) \]  
for \( i = 1, 2, n - 1 \). However at the point \( x_n \), second order accuracy of first derivative of backward difference, which is derived from the Taylor series expansion given as

\[ y'(x_n) = \frac{3y(x_n) - 4y(x_{n-1}) + y(x_{n-2})}{2h} + O(h^2) \]  
are considered. For solving second order LFIDEs, the second derivative central difference schemes can be derived as

\[ y''(x_i) = \frac{y(x_{i+1}) - 2y(x_i) + y(x_{i-1})}{h^2} + O(h^2) \]  
where \( h \) is size interval between nodes as mentioned in (6). For (7), (8) and (9) have the same order of the truncation error where mostly under our control because we can choose number of terms from the expansion of Taylor series. In order to obtain the finite grid work network for formulation of the full- and half-sweep central difference approximation equations over (1), the (2), (3) and (4) can be rewritten in general form as

\[ y'(x_i) \approx \frac{y(x_{i+p}) - y(x_{i-p})}{2ph}. \]
for \( i = 1, 2, 3, \ldots, n - p \) and
\[
y'(x_n) = \frac{3y(x_n) - 4y(x_{n-p}) + y(x_{n-2p})}{2ph},
\]
(11)

for \( i = n \)

the second derivative of second order central difference schemes can be derived as
\[
y''(x) = \frac{y(x_{i+p}) - 2y(x_i) + y(x_{i-p})}{(ph)^2} + O(h^2)
\]
(12)

where,
\[
E = \begin{bmatrix}
d_{p,p} & b_{p,2p} & d_{p,3p} & \cdots & d_{p,n-2p} & d_{p,n-p} & d_{p,n} \\
c_{2p,p} & a_{2p,2p} & b_{2p,3p} & \cdots & d_{2p,n-2p} & d_{2p,n-p} & d_{2p,n} \\
d_{3p,p} & c_{3p,2p} & a_{3p,3p} & \cdots & d_{3p,n-2p} & d_{3p,n-p} & d_{3p,n} \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
d_{n-2p,p} & d_{n-2p,2p} & d_{n-2p,3p} & \cdots & a_{n-2p,n-2p} & b_{n-2p,n-p} & d_{n-2p,n} \\
d_{n-p,p} & d_{n-p,2p} & d_{n-p,3p} & \cdots & e_{n-p,n-2p} & a_{n-p,n-p} & b_{n-p,n} \\
d_{n,n-p} & d_{n,n-2p} & d_{n,n-3p} & \cdots & b_{n,2p} & e_{n,n-p} & h_n
\end{bmatrix} \begin{bmatrix}
a_{i,j} \\
b_{i,j} \\
c_{i,j} \\
d_{i,j} \\
e_{i,j} \\
h_{i,j}
\end{bmatrix} = \begin{bmatrix}
2h_y + \left(2hA_pK_{p,0} + 1\right)y_0 \\
2h_y + \left(2hA_pK_{2,0}\right)y_0 \\
2h_y + \left(2hA_pK_{3,0}\right)y_0 \\
\vdots \\
2h_y + \left(2hA_pK_{n-2,0}\right)y_0 \\
2h_y + \left(2hA_pK_{n-p,0}\right)y_0 \\
2h_y + \left(2hA_pK_{n,0}\right)y_0
\end{bmatrix} \begin{bmatrix}
y_n(x_p) \\
y_n(x_{2p}) \\
y_n(x_{3p}) \\
\vdots \\
y_n(x_{n-2p}) \\
y_n(x_{n-p}) \\
y_n(x_n)
\end{bmatrix}
\]
(15)

where \( E^* y = f^* \)

and
\[
E^* = E^T E
\]

\[
f^* = E^T f
\]

Now the linear system (15) can be solved iteratively via FSGS and HSGS iterative methods.

For second order LFIDEs, equations (2) and (11) will be substituted into (1) to generate linear system either by the full-sweep or half-sweep approximation equation can be simply shown as
\[
G y_n = \ell
\]
(16)

where

\[
G = \begin{bmatrix}
\sigma_{1,p} & \zeta_{1,2p} & \tau_{1,3p} & \ldots & \tau_{1,p-3p} & \tau_{1,p-2p} & \tau_{1,p-p} \\
\zeta_{2,p} & \sigma_{2,2p} & \zeta_{2,3p} & \ldots & \tau_{2,p-3p} & \tau_{2,p-2p} & \tau_{2,p-p} \\
\tau_{3,p} & \zeta_{3,2p} & \sigma_{3,3p} & \ldots & \tau_{3,p-3p} & \tau_{3,p-2p} & \tau_{3,p-p} \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
\tau_{p-3,p} & \tau_{p-2,2p} & \tau_{p-1,3p} & \ldots & \sigma_{p-3,2p} & \zeta_{p-3,3p} & \tau_{p-2,p-p} \\
\tau_{p-2,p} & \tau_{p-1,2p} & \tau_{p,3p} & \ldots & \tau_{p-2,2p} & \sigma_{p-2,3p} & \zeta_{p-1,p-p} \\
\tau_{p-1,p} & \tau_{p,2p} & \tau_{p,3p} & \ldots & \tau_{p-1,2p} & \tau_{p,3p} & \sigma_{p-1,p-p}
\end{bmatrix},
\]

where \( G \) is a positive definite, non symmetric coefficient matrix, \( f \) is given function, and \( y \) is unknown function to be determined.

III. FORMULATION OF FSGS AND HSGS ITERATIVE METHODS

In this section, generated system of linear equation of first order and second order LFIDEs will be solved by using FSGS and HSGS iterative methods as shown in (15) and (16). For first order LFIDEs, let the coefficient matrix, \( E^* \) be decomposed into

\[
E^* = D - L - U
\]

where \( D, \ L, \) and \( -U \) are diagonal, strictly lower triangular and strictly upper triangular matrices respectively. In fact, the both iterative methods attempt to find a solution to the system of linear equations by repeatedly solving the linear system using approximations to the vector \( y \). Iterations for both methods continue until the solution is within a predetermined acceptable bound on the error. By determining values of matrices \( D, \ -L \) and \( -U \) as stated in (15), the general algorithm for FSGS and HSGS iterative methods to solve (1) would be generally described in Algorithm 1.

Algorithm 1: FSGS and HSGS algorithms

(i) Initializing all the parameters. Set \( k = 0 \).

1. \( i = 1, 2, \ldots, n - 1 \) and \( j = 1, 2, \ldots, n \), calculate

\[
y_i^{(k+1)} = \frac{1}{E_{ii}} \left( f_{i} - \sum_{j=p+2}^{n} E_{ij} y_{j}^{(k+1)} - \sum_{j=p+2}^{n} E_{ij} y_{j}^{(k)} \right)
\]

(ii) Convergence test.

If there error of tolerance \( \|y_i^{(k+1)} - y_i^{(k)}\| \leq \varepsilon = 10^{-10} \) is satisfied, then algorithms stop.

(iv) Else, set \( k = k+1 \) and go to step (ii).

For second order LFIDEs, the general algorithm for FSGS and HSGS iterative methods to solve (1) would be generally described in Algorithm 2.

\[
G = D - L - U
\]

where \( D, \ -L \) and \( -U \) are diagonal, strictly lower triangular and strictly upper triangular matrices respectively.
Algorithm 2: FSGS and HSGS algorithms

(i) Initializing all the parameters. Set \( k = 0 \).

(ii) For \( i = 1, 2, \ldots, n-p \) and \( j = 1, 2, \ldots, n-p \), calculate
\[
y^{(k+1)}(x) = \frac{1}{G_{ij}} \left( \ell_i - \sum_{j=p+2}^{p} G_{ij} y^{(k+1)} - \sum_{j=p+3}^{n-1} G_{ij} y^{(k)} \right)
\]

(iii) Convergence test.
If the error of tolerance \( \|y^{(k+1)} - y^{(k)}\| \leq \varepsilon = 10^{-10} \) is satisfied, then algorithms stop.

(iv) Else, set \( k = k+1 \) and go to step (ii).

IV. NUMERICAL SIMULATIONS

In order to evaluate the performances of the HSGS iterative methods described in the previous section, several numerical experiments were carried out. In this paper, we will only consider well posed equations and the case where \( a = 0 \) and \( b = 1 \).

Problem 1 [16]. Consider the first order LFIDE
\[
y(x) = \frac{1}{3} x + \int_{0}^{1} xy(t) dt , \quad 0 \leq x \leq 1
\]
with boundary condition
\[
y(0) = 0
\]
and exact solution is
\[
y(x) = x.
\]

Problem 2 [17]. Consider the second order LFIDE
\[
y''(x) = x - 2 + \int_{0}^{1} 60(x-t)y(t) dt , \quad 0 \leq x \leq 1
\]
with boundary conditions
\[
y(0) = 0 \quad \text{and} \quad y(1) = 0
\]
and exact solution given as
\[
y(x) = x.
\]

There are three parameters considered in numerical comparison such as number of iterations, execution time and maximum absolute error. As comparisons, the Standard or Full Sweep Gauss-Seidel (FSGS) method acts as the control of comparison of numerical results. Throughout the simulations, the convergence test considered the tolerance error of the \( \varepsilon = 10^{-10} \).

V. CONCLUSIONS

In this work, we have implemented half-sweep iterative method on high order closed composite Newton Cotes quadrature schemes to solve LFIDEs. Based on Table III and Table IV, the half-sweep iteration concept on quadrature and central difference schemes with GS iterative method have decreased the number of iterations and execution time approximately 83.05% respectively for problem 2. Based on Table 1 and Table 2 the accuracy of numerical solutions for CD-RS1 and CD-RS2 schemes are more accurate than the CD-RT scheme. Overall, the numerical results have shown that the HSGS method is more superior in term of number of iterations and the execution time than standard method.

<table>
<thead>
<tr>
<th>Arithmetic Operations Per Node</th>
<th>ADD/SUB</th>
<th>MUL/DIV</th>
</tr>
</thead>
<tbody>
<tr>
<td>FSGS</td>
<td>( n(n-1) )</td>
<td>( n(n+1) )</td>
</tr>
<tr>
<td>HSGS</td>
<td>( \frac{n}{2}, \frac{n}{2} + 1 )</td>
<td>( \frac{n}{2}, \frac{n}{2} + 1 )</td>
</tr>
</tbody>
</table>

REFERENCES


Elayaraja Aruchunan from Malaysia. MSc, BSc, degree of Mathematics from University Malaysia Sabah (UMS). The author’s major field of study is Numerical Analysis and his area of interest in Integro-differential equation (IDE), Integral equations (IE), Ordinary Differential Equation (ODE) and Partial Differential Equation (PDE). His is currently lecturing in Curtin University Sarawak in School of Engineering and Science. His has published more than 20 publications. He is also as a member in International Linear Algebra Society (ILAS), International Association of Computer Science and Information (IACSIT) and International Association of Engineers (IAENG).

<table>
<thead>
<tr>
<th>Mesh Size</th>
<th>Schemes &amp; Methods</th>
<th>Number of iteration</th>
<th>Execution time</th>
<th>Maximum absolute error</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>FSGS</td>
<td>HSGS</td>
<td>FSGS</td>
</tr>
<tr>
<td>24</td>
<td>CD-RT</td>
<td>7814</td>
<td>2901</td>
<td>5.93</td>
</tr>
<tr>
<td></td>
<td>CD-RS1</td>
<td>7964</td>
<td>2962</td>
<td>6.53</td>
</tr>
<tr>
<td></td>
<td>CD-RS2</td>
<td>7811</td>
<td>2903</td>
<td>6.47</td>
</tr>
<tr>
<td>48</td>
<td>CD-RT</td>
<td>23006</td>
<td>7814</td>
<td>108.77</td>
</tr>
<tr>
<td></td>
<td>CD-RS1</td>
<td>23428</td>
<td>7964</td>
<td>120.52</td>
</tr>
<tr>
<td></td>
<td>CD-RS2</td>
<td>23004</td>
<td>7811</td>
<td>116.62</td>
</tr>
<tr>
<td>72</td>
<td>CD-RT</td>
<td>45002</td>
<td>14536</td>
<td>684.15</td>
</tr>
<tr>
<td></td>
<td>CD-RS1</td>
<td>45756</td>
<td>14810</td>
<td>730.15</td>
</tr>
<tr>
<td></td>
<td>CD-RS2</td>
<td>45001</td>
<td>14534</td>
<td>723.21</td>
</tr>
<tr>
<td>96</td>
<td>CD-RT</td>
<td>73430</td>
<td>23006</td>
<td>2469.69</td>
</tr>
<tr>
<td></td>
<td>CD-RS1</td>
<td>74614</td>
<td>23428</td>
<td>2753.81</td>
</tr>
<tr>
<td></td>
<td>CD-RS2</td>
<td>73428</td>
<td>23004</td>
<td>2631.58</td>
</tr>
<tr>
<td>120</td>
<td>CD-RT</td>
<td>107988</td>
<td>33174</td>
<td>10347.03</td>
</tr>
<tr>
<td></td>
<td>CD-RS1</td>
<td>109685</td>
<td>33759</td>
<td>10460.84</td>
</tr>
<tr>
<td></td>
<td>CD-RS2</td>
<td>107987</td>
<td>10950</td>
<td>11571.30</td>
</tr>
</tbody>
</table>
### TABLE IV.
Comparison of a number of iterations, execution time (seconds) and maximum absolute error for iterative methods using CD-RT, CD-RS1 and CD-RS2 discretization schemes for Problem 2.

<table>
<thead>
<tr>
<th>Mesh Size</th>
<th>Schemes &amp; Methods</th>
<th>Number of iteration</th>
<th>Execution time</th>
<th>Maximum absolute error</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>FSGS</td>
<td>HSGS</td>
<td>FSGS</td>
</tr>
<tr>
<td>24</td>
<td>CD-RT</td>
<td>502</td>
<td>130</td>
<td>0.22</td>
</tr>
<tr>
<td></td>
<td>CD-RS1</td>
<td>497</td>
<td>134</td>
<td>0.29</td>
</tr>
<tr>
<td></td>
<td>CD-RS2</td>
<td>497</td>
<td>134</td>
<td>0.31</td>
</tr>
<tr>
<td>48</td>
<td>CD-RT</td>
<td>2101</td>
<td>502</td>
<td>0.49</td>
</tr>
<tr>
<td></td>
<td>CD-RS1</td>
<td>2097</td>
<td>497</td>
<td>0.50</td>
</tr>
<tr>
<td></td>
<td>CD-RS2</td>
<td>2097</td>
<td>497</td>
<td>0.57</td>
</tr>
<tr>
<td>72</td>
<td>CD-RT</td>
<td>4628</td>
<td>1183</td>
<td>1.17</td>
</tr>
<tr>
<td></td>
<td>CD-RS1</td>
<td>4625</td>
<td>1179</td>
<td>1.18</td>
</tr>
<tr>
<td></td>
<td>CD-RS2</td>
<td>4625</td>
<td>1179</td>
<td>1.20</td>
</tr>
<tr>
<td>96</td>
<td>CD-RT</td>
<td>8034</td>
<td>2101</td>
<td>2.30</td>
</tr>
<tr>
<td></td>
<td>CD-RS1</td>
<td>8032</td>
<td>2097</td>
<td>2.34</td>
</tr>
<tr>
<td></td>
<td>CD-RS2</td>
<td>8032</td>
<td>2097</td>
<td>2.69</td>
</tr>
<tr>
<td>120</td>
<td>CD-RT</td>
<td>12278</td>
<td>3251</td>
<td>4.37</td>
</tr>
<tr>
<td></td>
<td>CD-RS1</td>
<td>12276</td>
<td>3249</td>
<td>4.10</td>
</tr>
<tr>
<td></td>
<td>CD-RS2</td>
<td>12276</td>
<td>3249</td>
<td>4.77</td>
</tr>
</tbody>
</table>