Mathematics and Statistics 10(3): 486-497, 2022 http://www.hrpub.org
DOI: 10.13189/ms.2022.100304

Half-Sweep Refinement of SOR Iterative Method via Linear Rational Finite Difference Approximation for Second-Order Linear Fredholm Integro-Differential Equations

Ming-Ming Xu[1,2,*], Jumat Sulaiman[1,3], Nur Afza Mat Ali[1]

[1]Faculty of Science and Natural Resources, Universiti Malaysia Sabah, 88400 Kota Kinabalu, Sabah, Malaysia
[2]School of Mathematics and Information Technology, Xingtai University, 054001 Xingtai, Hebei, China
[3]Preparatory Centre for Science and Technology, Universiti Malaysia Sabah, 88400 Kota Kinabalu, Sabah, Malaysia

Received February 15, 2022; Revised April 7, 2022; Accepted April 22, 2022

Cite This Paper in the following Citation Styles

(a): *[1] Ming-Ming Xu, Jumat Sulaiman, Nur Afza Mat Ali , "Half-Sweep Refinement of SOR Iterative Method via Linear Rational Finite Difference Approximation for Second-Order Linear Fredholm Integro-Differential Equations," Mathematics and Statistics, Vol. 10, No. 3, pp. 486-497, 2022. DOI: 10.13189/ms.2022.100304.*

(b): *Ming-Ming Xu, Jumat Sulaiman, Nur Afza Mat Ali (2022). Half-Sweep Refinement of SOR Iterative Method via Linear Rational Finite Difference Approximation for Second-Order Linear Fredholm Integro-Differential Equations. Mathematics and Statistics, 10(3), 486-497. DOI: 10.13189/ms.2022.100304.*

Abstract The numerical solutions of the second-order linear Fredholm integro-differential equations have been considered and discussed based on several discretization schemes. In this paper, the new schemes are developed derived on the hybrid of the three-point half-sweep linear rational finite difference (3HSLRFD) approaches with the half-sweep composite trapezoidal (HSCT) approach. The main advantage of the established schemes is that they discretize the differential terms and integral term of second-order linear Fredholm integro-differential equations into the algebraic equations and generate the corresponding linear system. Furthermore, the half-sweep (HS) concept is combined with the refinement of the successive over-relaxation (RSOR) iterative method to create the new half-sweep successive over-relaxation (HSRSOR) iterative method, which is implemented to get the numerical solution of a system of linear algebraic equations. Apart from that, the classical or full-sweep Gauss-Seidel (FSGS) and full-sweep successive over-relaxation iterative (FSSOR) methods are presented, which serve as the control method in this paper. In the end, we employed FSGS, FSRSOR and HSRSOR methods to obtain numerical solutions of three examples and make a detailed comparison from three aspects of the number of iterations, elapsed time and maximum absolute error. Numerical results demonstrate that FSRSOR and HSRSOR methods have lesser iterations, faster elapsed time, and are more accurate than FSGS. In addition, HSRSOR is the most effective of the three methods. To sum up, this paper has successfully proposed the applicability and superiority of the new HSRSOR method based on 3HSLRFD-HSCT schemes.

Keywords Second-Order Integro-Differential Equations, Half-Sweep Refinement of SOR Iterative Method, Three-Point Half-Sweep Linear Rational Finite Difference Schemes, Half-Sweep Composite Trapezoidal Scheme

1. Introduction

Integro-differential equations (IDEs) have been used extensively in biological models, economics, oscillation theory, ocean circulations, control theory of industrial mathematics and other fields [1-3] since they were introduced by Volterra [4]. Due to inherent complexity, it is not easy to obtain analytical solutions for most IDEs, so numerical methods are often used to approximate their exact solutions, including the shifted Legendre spectral collocation method [5], a two-grid finite element method [6], Atangana–Baleanu approach [7], Pseudo-operational matrix method [8], and Euler wavelets method [9]. The goal of this paper is to inspect the numerical technique to approximate the solution of the general linear second-order Fredholm integro-differential equations (FIDEs) of form

$$y''(x) = \alpha(x)y'(x) + \beta(x)y(x) + \gamma(x) + \int_{\varphi}^{\phi} K(x,u)y(u)du, \tag{1}$$

$\varphi \le x \le \phi$, with boundary conditions at two points

$$y(\varphi) = y_{\varphi},\ y(\phi) = y_{\phi},$$

where φ and ϕ are constant, the functions $\alpha(x)$, $\beta(x)$, $\gamma(x)$ and the kernel $K(x,u)$ are known, the $y(x)$ is an unknown function that needs to be determined.

In recent years, a growing number of numerical techniques have been outlined to solve the numerical solutions of problem (1), such as, variational iteration method [10], reproducing kernel Hilbert space method [11], Hermite Trigonometric Wavelets method [12], Non-standard finite difference method [13], Adomian Decomposition Method [14], A multiscale Galerkin method [15], exponential spline [16], hybridizable discontinuous Galerkin method [17]. In this study, we mainly applied three schemes, namely, the three-point half-sweep linear rational finite difference (3HSLRFD), the half-sweep composite trapezoidal (HSCT) and half-sweep refinement of Successive Over-Relaxation (HSRSOR), which are the combination of half-sweep (HS) [18] iteration concept and three-point linear rational finite difference (3LRFD), composite trapezoidal (CT) [19] and refinement of Successive Over-Relaxation (RSOR) [20,21] respectively. First, let us review the HS iteration concept, which was first proposed by Abdullah [18] in 1991, mainly for solving two-dimensional Poisson equations. The main advantage of the half-sweep iteration method is that it reduces the computing cost of full-sweep linear systems created by matching approximation equations. Because of the low computing complexity of this approach, more research has been undertaken in [22-26] to reveal the feasibility of the HS iteration. In addition to these one-stage iteration concepts, many combinations of HS iteration concepts with two-stage iterative techniques, such as HSIADE [27], HSAGE [28,29], HSAM [30], and HSGM [31], have been developed to solve linear problems. They demonstrated that the two-stage iterative methods they devised are among the most efficient iterative approaches for solving any system of linear equations. Furthermore, the basic multigrid techniques have been enhanced by the introduction of a family of HS multigrid methods [32,33]. As a result, Hassan et al. [34,35] developed a series of FDTD algorithms for tackling wave propagation issues based on this notion. Meanwhile, the effectiveness and feasibility of the half-sweep iteration approach have been successfully applied and worked on robotic path planning [36,37].

The key idea of this concept is to take only half of the points in the solving domain of the problem to be solved, which naturally reduces the number of iterations and speeds up the elapsed time. Considering these advantages, future discussions in this paper will focus on extending it in conjunction with 3LRFD, CT and RSOR, three schemes to obtain the numerical solutions of problem (1).

The linear rational finite difference (LRFD) [38] scheme is a derivative of linear barycentric rational interpolation (LBRI), which offers a better approximation than the finite difference (FD) method, particularly when approximating the derivative value of a point near the end of the interval. Therefore, it has been favored by researchers in recent years. For instance, it is applied to solve the problem of VIDEs [39], delay VIDEs [40], stiff ODEs [41] and first-order FIDEs [42-44]. Meanwhile, this also motivates us to extend the LRFD method to the three-point LRFD (3LRFD) scheme to discretize the differential terms of the FIDEs. In this paper, the 3HSLRFD and HSCT methods are combined to establish the new 3HSLRFD-HSCT discretization schemes, which are used to discretize the $y'(x)$, $y''(x)$ and $\int_{\varphi}^{\phi} K(x,u)y(u)du$ terms over problem (1) to obtain the corresponding approximation equations, then generate the corresponding linear system. Finally, the newly constructed HSRSOR scheme is implemented to find the numerical solution of the corresponding linear system.

We shall divide this paper into four sections besides the present one. In the second section, three new methods of 3HSLRFD, HSCT and HSRSOR are constructed, and the numerical solution of problem (1) is derived in detail by using these three schemes. The third section preforms the numerical experiments. The final section summarises conclusions based on the findings work.

2. Methodology

In this section, the newly constructed 3HSLRFD, HSCT and HSRSOR schemes are used to acquire the numerical solution of problem (1). The solution procedure is done in two stages. The first stage is to construct the new 3HSLRFD-HSCT discretization schemes of the problem (1). The second stage is by using the HSRSOR iterative method to obtain the numerical solution of the corresponding approximation equation for the problem (1).

Divide the domain $[\varphi,\phi]$ into N equal-length subintervals

$$h = \frac{\phi - \varphi}{N},\ x_i = u_i = \varphi + ih,\ i = 0,1,\cdots,N.$$

In this paper, let $N = 2^{\mu}$, $\mu \ge 1$. Before constructing these three new methods, let us review the HS iteration concept [18]. The general solution procedure is based on the full-sweep (FS) iteration concept, which takes all the interval's partition points. On the other hand, the HS iteration method only takes even partition points into account. Figure 1 depicts the grid-point distribution. The iteration methods FS and HS are represented by (a) and (b), respectively. According to Figure 1, the FS and HS iteration concepts will only compute approximate values onto node points of type ● until the convergence condition is met. Then, using the direct method [18,22-31], other approximate solutions can be computed at the remaining points (points of a different type ○).

In comparison to h for each grid size of the FS iteration, each grid size of the HS iteration is $2h$, implying that the latter is significantly faster and more computationally efficient than the former.

Considering the distribution of uniformly points in the HS case, we begin to combine the HS iteration method alone with the 3LRFD, CT and RSOR schemes, respectively, to build three new schemes, namely 3HSLRFD, HSCT and HSRSOR.

2.1. The Three-Point Half-Sweep Linear Rational Finite Difference Scheme

As mentioned at the beginning of this section, the first step in the solution process is to hybridize the 3HSLRFD and HSCT formulas to develop a fast and reliable algorithm for constructing the approximate equation of the problem (1). In this subsection, we attempt to construct the new 3HSLRFD schemes, primarily applied to discretize the differential terms of the problem (1).

First, we will discuss the LRFD method [38], which is based on LBRI as stated in the introduction section. Let us consider a grid

$$\varphi = x_0 < x_1 \cdots < x_m = \phi,$$

for the given interval $[\varphi, \phi]$ and corresponding values $y(x_j)$, $j = 0,1,\cdots,m$. The LBRI for these data will be as follows:

$$Y_{F_m}(x) = \sum_{j=0}^{m} \left(\frac{\left(\dfrac{\xi_{F_j}}{x - x_j} \right) y(x_j)}{\left(\displaystyle\sum_{j=0}^{m} \frac{\xi_{F_j}}{x - x_j} \right)} \right), \qquad (2)$$

where the weights ξ_{F_j}, were proposed by Floater and Hormann in 2007, the weights formulas for nodes with the same step size h, are given as [45]

$$\xi_{F_j} = \frac{(-1)^{j-d}}{2^d} \sum_{s \in J_{F_j}} \binom{d}{j-s}, \qquad (3)$$
$$(j = 0,1,\cdots,m)$$

with

$$J_{F_j} := \left\{ s \in \{0,1,\cdots,m-d\} \mid j-d \le s \le j \right\}.$$

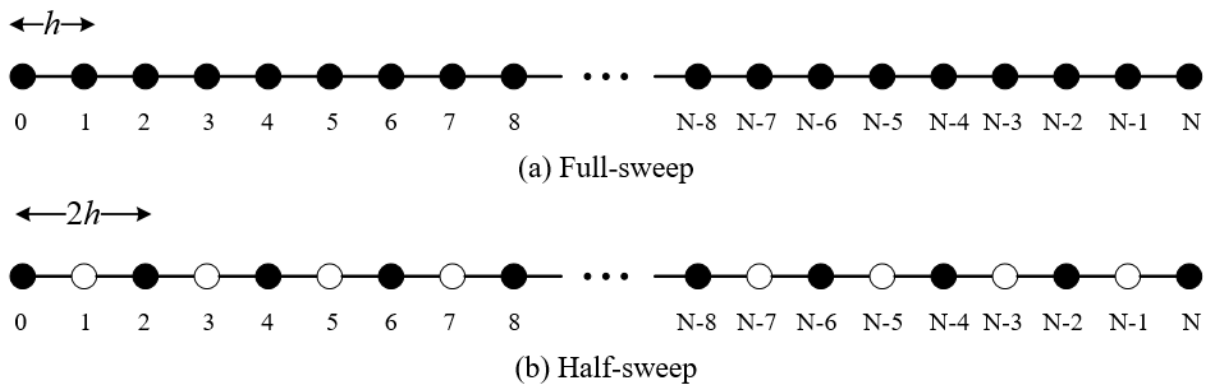

Figure 1. Distribution of the uniform mesh size for full- and half-sweep cases

In 2012, Klien and Berrt [38] introduced the derivatives of LBRI, and the formulas of LRFD to approximate the $y'(x)$ and $y''(x)$ on $x_0 < x_1 \cdots < x_m$ are written as

$$y'(x) \approx Y'_{F_m}(x_i) = \frac{1}{h}\sum_{j=0}^{m} D^{(1)}_{F_{i,j}} y(x_j), \quad (i = 0,1,\cdots,m) \tag{4}$$

and

$$y''(x) \approx Y''_{F_m}(x_i) = \frac{2}{h^2}\sum_{j=0}^{m} D^{(2)}_{F_{i,j}} y(x_j), \quad (i = 0,1,\cdots,m) \tag{5}$$

where

$$D^{(1)}_{F_{i,j}} = \begin{cases} \dfrac{1}{i-j}\dfrac{\xi_{F_j}}{\xi_{F_i}}, & j \neq i, \\ -\displaystyle\sum_{l=0;l\neq i}^{m} D^{(1)}_{F_{i,l}}, & j = i, \end{cases} \quad (i, j = 0,1,\cdots,m) \tag{6}$$

and

$$D^{(2)}_{F_{i,j}} = \begin{cases} \dfrac{1}{i-j}\left(\dfrac{\xi_{F_j}}{\xi_{F_i}} D^{(1)}_{F_{i,j}} - D^{(1)}_{F_{i,j}}\right), & j \neq i, \\ -\displaystyle\sum_{l=0;l\neq i}^{m} D^{(2)}_{F_{i,l}}, & j = i. \end{cases} \quad (i, j = 0,1,\cdots,m) \tag{7}$$

Based on the introduction of the HS iteration concept and referring to (2)-(3) in this section. The half-sweep LBRI (HSLBRI) method on $x_0 < x_2 \cdots < x_m$ (m here and below is even) can be built as follows

$$Y_{H_m}(x) = \sum_{j=0,2}^{m} \left(\frac{\left(\dfrac{\xi_{H_j}}{x - x_j}\right) y(x_j)}{\left(\displaystyle\sum_{j=0,2}^{m} \frac{\xi_{H_j}}{x - x_j}\right)} \right), \tag{8}$$

where

$$\xi_{H_j} = \frac{(-1)^{\frac{j}{2}-d}}{2^d} \sum_{s \in J_{H_j}} \begin{pmatrix} d \\ \dfrac{j}{2} - s \end{pmatrix}, \quad (j = 0,2,\cdots,m) \tag{9}$$

with

$$J_{H_j} := \left\{ s \in \left\{0,1,\cdots,\frac{m}{2} - d\right\} \mid \frac{j}{2} - d \leq s \leq \frac{j}{2} \right\}.$$

In a similar way to derive HSLBRI, based on the HS iteration concept and referring to (4)-(7), half-sweep linear rational finite difference (HSLRFD) schemes to approximate the $y'(x)$ and $y''(t)$ on $x_0 < x_2 \cdots < x_m$ are written as

$$y'(x) \approx Y'_{H_m}(x_i) = \frac{1}{h}\sum_{j=0,2}^{m} D^{(1)}_{H_{i,j}} y(x_j), \quad (i=0,2,\cdots,m) \tag{10}$$

and

$$y''(x) \approx Y''_{H_m}(x_i) = \frac{2}{h^2}\sum_{j=0,2}^{m} D^{(2)}_{H_{i,j}} y(x_j), \quad (i = 0,2,\cdots,m) \tag{11}$$

with

$$D^{(1)}_{H_{i,j}} = \begin{cases} \dfrac{1}{i-j}\dfrac{\xi_{H_j}}{\xi_{H_i}}, & j \neq i, \\ -\displaystyle\sum_{l=0,2,l\neq i}^{m} D^{(2)}_{H_{i,l}}, & j = i, \end{cases} \quad (i, j = 0,2,\cdots,m) \tag{12}$$

and

$$D^{(2)}_{H_{i,j}} = \begin{cases} \dfrac{1}{i-j}\left(\dfrac{\xi_{H_j}}{\xi_{H_i}} D^{(1)}_{H_{i,j}} - D^{(1)}_{H_{i,j}}\right), & j \neq i, \\ -\displaystyle\sum_{l=0,2;l\neq i}^{m} D^{(2)}_{H_{i,l}}, & j = i. \end{cases} \quad (i, j = 0,2,\cdots,m) \tag{13}$$

In contrast to the HSLBRI scheme in (8) and HSLRFD schemes in (10)-(11) developed based on the HS iteration concept in this paper, the LBRI scheme in (2) and LRFD schemes in (4)-(5) are also referred to as FSLBRI and FSLRFD, respectively. We need to conside three-point to derive the 3HSLRFD scheme, so assume $m = 2$ in (2).

Equation (8) is the interpolation function on $(m/2+1)$ interpolation nodes. In the following, we only consider the interpolation on $x_{i-2}, x_i, x_{i+2}, i = 2,4,\cdots,N-2,$ and in combination with (9)-(13), we quickly derive the 3HSLRFD formulas

$$y'(x) \approx Y'(x_i) + \varepsilon^{(1)}(x_i), (i = 2,4,\cdots,N-2) \tag{14}$$

and

$$y''(x) \approx Y''(x_i) + \varepsilon^{(2)}(x_i), \ (i = 2,4,\cdots,N-2) \tag{15}$$

where $\varepsilon^{(1)}(x_i)$ are $\varepsilon^{(2)}(x_i)$ truncation errors. Then the expression of $Y'(x_i)$ and $Y''(x_i)$ can be defined as

$$Y'(x_i) = \frac{1}{h}\sum_{j=i-2,i}^{i+2} D^{(1)}_{i,j} y(x_j), \tag{16}$$

and

$$Y''(x_i) = \frac{2}{h^2}\sum_{j=i-2,i}^{i+2} D^{(2)}_{i,j} y(x_j), \tag{17}$$

where

$$D^{(1)}_{i,j} = \begin{cases} \dfrac{1}{i-j}\dfrac{\xi_{i,j}}{\xi_{i,i}}, & j \neq i, \\ -\left(D^{(1)}_{i,i-2} + D^{(1)}_{i,i+2}\right), & j = i, \end{cases} \quad (i, j = 2,4,\cdots,N-2) \tag{18}$$

and

$$D_{i,j}^{(2)} = \begin{cases} \dfrac{1}{i-j}\left(\dfrac{\xi_{i,j}}{\xi_{i,i}}D_{i,i}^{(1)} - D_{i,j}^{(1)}\right), & j \neq i, \\ -\left(D_{i,i-2}^{(2)} + D_{i,i+2}^{(2)}\right), & j = i. \end{cases} \quad (i,j = 2,4,\cdots,N-2) \tag{19}$$

In this study, to derive an approximation equation of the problem (1), we mainly used (14) and (15) to discretize the first and second derivative terms of (1) respectively. For the 3HSLRFD schemes, we take $d = 1$, and the corresponding values of $\xi_{i,i}$, $D_{i,j}^{(1)}$ and $D_{i,j}^{(2)}$ are shown in Tables 1 and 2, the corresponding order of error accuracy can then be calculated using Taylor series expansion as follows: $\left|\varepsilon^1(x_i)\right| = O(h^2), \left|\varepsilon^2(x_i)\right| = O(h^2)$.

Table 1. The values of $\xi_{i,j}$

$\xi_{i,i-2}$	$\xi_{i,i}$	$\xi_{i,i+2}$
$-1/2$	1	$-1/2$

Table 2. The values of $D_{i,j}^{(1)}$ and $D_{i,j}^{(2)}$

$D_{i,i-2}^{(1)}$	$D_{i,i}^{(1)}$	$D_{i,i+2}^{(1)}$	$D_{i,i-2}^{(2)}$	$D_{i,i}^{(2)}$	$D_{i,i+2}^{(2)}$
$-1/4$	0	$1/4$	$1/8$	$-1/4$	$1/8$

2.2. The Half-Sweep Composite Trapezoidal Scheme

This subsection elaborates the HSCT scheme, which is derived from a family of quadrature methods and is used to discretize the integral term of the problem (1), which coincides to differential terms in subsection 2.1.

In general, the quadrature scheme is written as

$$\int_{\varphi}^{\phi} y(u)du = \sum_{j=0,1}^{N} C_{F_j} y(u_j) + \tilde{\delta}_N(y), \tag{20}$$

where

C_{F_j} are the numerical coefficients;

$\tilde{\delta}_N(y)$ is the truncation error.

To construct the quadrature scheme in getting the approximation equations of (1), we look at the composite trapezoidal (CT) scheme. As a result, the C_{F_j} based on the CT formula is as follows

$$C_{F_j} = \begin{cases} \dfrac{1}{2}h, & j = 0, N, \\ h, & j = 1,2,\cdots,N-1. \end{cases} \tag{21}$$

In this paper, we also call (20) as a full-sweep composite trapezoidal (FSCT) scheme. In contrast with the FSCT scheme of (20), The HSCT formula is obtained by combining the HS iteration method with the CT formula as follows

$$\int_{\varphi}^{\phi} y(u)du = \sum_{j=0,2}^{N} C_j y(u_j) + \delta_N(y), \tag{22}$$

where

$$C_j = \begin{cases} h, & j = 0, N, \\ 2h, & j = 2,4,\cdots,N-2. \end{cases} \tag{23}$$

By substituting (14), (15) and (22) into (1), the general form of the 3HSLRFD-HSCT approximate equation can be constructed as

$$\frac{1}{h^2}\sum_{j=i-2,i}^{i+2} D_{i,j}^{(2)} y_j = \frac{1}{h}\alpha_i \sum_{j=i-2,i}^{i+2} D_{i,j}^{(1)} y_j + \beta_i y_i + \gamma_i + \sum_{j=0,2}^{N} C_j K_{i,j} y_j, \quad (i = 2,4,\cdots,N-2) \tag{24}$$

where $y_i = y(x_i)$, $\alpha_i = \alpha(x_i)$, $\beta_i = \beta(x_i)$, $\gamma_i = \gamma(x_i)$ and $K_{i,j} = K(x_i, x_j)$.

Focusing on the approximation equations (24), the related linear system can be easily expressed as

$$My = F, \tag{25}$$

where $M = \tilde{M}^T \tilde{M}, F = \tilde{M}^T \tilde{F}$,

$$y^T = \begin{bmatrix} y_2 & y_4 & \cdots & y_{N-4} & y_{N-2} \end{bmatrix},$$

$$F^T = \begin{bmatrix} F_2 & F_4 & \cdots & F_{N-4} & F_{N-2} \end{bmatrix},$$

$$\tilde{F}^T = \begin{bmatrix} \gamma_2 + hK_{2,0}y_0 + hK_{2,N}y_N - \dfrac{\alpha_2 y_0}{4h} - \dfrac{y_0}{4h^2} \\ \gamma_4 + hK_{4,0}y_0 + hK_{4,N}y_N \\ \vdots \\ \gamma_{N-4} + hK_{N-4,0}y_0 + hK_{N-4,N}y_N \\ \gamma_{N-2} + hK_{N-2,0}y_0 + hK_{N-2,N}y_N + \dfrac{\alpha_{N-2}y_N}{4h} - \dfrac{y_N}{4h^2} \end{bmatrix}.$$

The first step of the solution process has now been completed, and we have obtained the 3HSLRFD-HSCT discretization schemes for the derivation of approximation equation (24) and used them to generate the corresponding linear system (25).

2.3. The Half-Sweep Refinement of Successive Over-Relaxation Scheme

We will move on to the second stage to find the numerical solution for the linear system (25). In general, there are direct and iterative approaches for solving the linear system. The former includes Cramer's rule, Gauss elimination, and LU decomposition suitable for solving a linear system with a low-scale coefficient matrix to get its approximate solution. Whereas, in practical problems, the coefficient matrix of a linear system is often large and dense, so the latter has incomparable advantages. Simultaneously, we know from the previous section that the distinctive feature of the linear system's coefficient matrix M is a dense matrix since the integral term adopts the HSCT discretization scheme. Thus, we apply iterative methods to obtain numerical solutions for the linear system (25).

In this study, we construct a new HSRSOR iterative method which is the combined form of HS concept and RSOR scheme, to solve a numerical solution of the linear system (25). To accomplish this, first let decompose the coefficient matrix M as the sum of three matrices, as shown below.

$$M = P - Q - R, \tag{26}$$

where

P is the matrix of diagonal;
Q is the matrix of strictly lower triangular;
R is the matrix of strictly upper triangular.

Therefore, the general formula for the HSSOR iterative scheme can be demonstrated [19,20,45,46].

$$y^{(k+1)} = \left(P - \omega Q\right)^{-1}\left((1-\omega)P + \omega R\right)y^{(k+1)} + \omega\left(P - \omega Q\right)^{-1}F, \tag{27}$$

where ω is relaxation factor, k is the number of iterations and

$$y^{(k)} = \left[y_2^{(k)} \quad y_4^{(k)} \quad \cdots \quad y_{N-2}^{(k)}\right]^T.$$

It can be reduced as the HSGS iterative method by taking ω=1 in (27).

By substituting (26) in (25), we get

$$\left(P - Q - R\right)y = F,$$

$$y = y + \omega(P - \omega Q)^{-1}(F - My).$$

Thus, the RSOR solution is denoted as

$$y^{(k+1)} = y^{(k+1)} + \omega\left(P - \omega Q\right)^{-1}\left(F - My^{(k+1)}\right), \tag{28}$$

Since $y^{(k+1)}$ appears on both sides of (28), we can substitute the $y^{(k+1)}$ on the right-hand side of (28) with (27) and come up at the HSRSOR method.

$$y^{(k+1)} = \left(\left(P - \omega Q\right)^{-1}\left((1-\omega)P + \omega R\right)\right)^2 y^{(k+1)} + \omega\left(I + \left(P - \omega Q\right)\left((1-\omega)P + \omega R\right)\right)\left(P - \omega Q\right)^{-1}F. \tag{29}$$

In contrast to HSGS, HSSOR, HSRGS, and HSRSOR iterative methods, the classical GS, SOR, RGS, and RSOR [21] iterative methods also known as FSGS, FSSOR, FSRGS, FSRSOR iterative methods, respectively. The convergence analysis of HSRSOR is similar to that of RSOR, which can be referred to [21].

To solve the problem (1), the HSRSOR iterative scheme is implemented to find the numerical solution of the linear system (25). The iteration process is repeated until the solution falls within a predetermined acceptable bound on the error, σ=10^{-10}. Algorithm 1 provides the general algorithm for the HSRSOR method of solving the linear system (25) with an approximate solution to the vector. In this case, Algorithm 1 is run through using MATLAB programming.

Algorithm 1: HSRSOR method

a. All values are being initialised. Set $k = 0$ and $y^{(\mathbf{0})} = \mathbf{0}$.

b. For $k = 0, 1, 2 \cdots$, perform

Compute

$$y^{(k+1)} = \left(\left(P - \omega Q\right)^{-1}\left((1-\omega)P + \omega R\right)\right)^2 y^{(k+1)} + \omega\left(I + \left(P - \omega Q\right)\left((1-\omega)P + \omega R\right)\right)\left(P - \omega Q\right)^{-1}F.$$

The convergence evaluate. If the tolerance error $\left\| y^{(k+1)} - y^{(k)} \right\| \le \sigma$=$10^{-10}$ is met, proceed to step c. Otherwise go back to step b.

c. Display the numerical solution.

d. Stop.

3. Numerical Experiments

The second section investigated the new method for solving the problem (1). This section conducted three numerical simulations to validate the applicability of the newly constructed HSRSOR iterative method based on the 3HSLRFD-HSCT discretization schemes.

Example 1 [47] Consider the second-order linear FIDEs

$$y''(x) = 32x + \int_{-1}^{1}(1 - xu)y(u)du, \tag{30}$$

$-1 \le x \le 1$, with boundary conditions at two points

$$y(-1) = -\frac{5}{2}, \ y(1) = \frac{15}{2},$$

and the exact solution of (30) is

$$y(x) = 5x^3 + \frac{3}{2}x^2 + 1.$$

Example 2 [1] Consider the second-order linear FIDEs

$$y''(x) = 2 - \frac{16}{15}x - \frac{16}{15}x^2 + \int_{-1}^{1}(xu^2 - x^2u)y(u)du, \tag{31}$$

$-1 \le x \le 1$, with boundary conditions at two points

$$y(-1) = -1, \ y(1) = 3,$$

and the exact solution of (31) is

$$y(t) = x^2 + x + 1.$$

Example 3 [1] Consider the second-order linear FIDEs

$$y''(x) = e^x - x + \int_0^1 xuy(u)du, \tag{32}$$

$0 \le x \le 1$, with boundary conditions at two points

$$y(0) = 1, \ \ y(1) = e,$$

and the exact solution of (32) is

$$y(x) = e^x.$$

In order to completely demonstrate the merits of the HSRSOR method, not only the HSRSOR method but also the classical FSGS and FSRSOR methods are implemented in the above three examples, where the latter two methods played the controlling roles. Whereby, the three criterias for the number of iterations (***Iterations***), the elapsed time (***Time***) in seconds, and the maximum absolute errors (***Error***) at five different numbers of subintervals $N = 32, 64, 128, 256, 512$ are compared, respectively. In the meantime, with the help of MATLAB software, the numerical results are quickly and reliably obtained. The corresponding results are shown in Tables 3-5.

As can be seen from Tables 3-5, when the same value of N is taken, among the three methods, HSRSOR method based on the 3HSLRFD-HSCT discretization schemes has the fewest ***Iterations*** and the fastest ***Time***, while FSGS method has the most ***Iterations*** and ***Time***. For FSRSOR and HSRSOR methods, when the value of N of the latter is twice that of the former, the results of their corresponding three parameters are almost the same, also see Figures. 2-4. Furthermore, we also calculated the percentage of decrement of the values of the first two parameters obtained using the FSRSOR and HSRSOR methods compared with the values obtained using the FSGS method, which are as high as 99%, as shown in Table 6. All these indicate that the HSRSOR method based on the 3HSLRFD-HSCT discretization schemes is the most meaningful and effective method for solving problem (1) among the three methods.

4. Conclusions

This paper successfully investigated the high performance of the newly developed HSRSOR iterative method based on the 3HSLRFD-HSCT discretization schemes for getting numerical solutions to the problem (1). The benefits of the proposed method are attributed not only to the 3HSLRFD-HSCT discretization scheme's good approximation properties but also to the fast convergence of the HSRSOR iterative method. In the future, we will extend our method to solve other types of equations, such as VIDEs and PDEs, as well as fractional and nonlinear cases [2,5-10,48]. It can also be developed to solve practical problems like population forecasting, stock investment, epidemic disease prevention [1,2,5,49].

List of the Abbreviations

3FSLRFD	Three-point full-sweep linear rational finite difference
3LRFD	Three-point linear rational finite difference
3HSLRFD	Three-point half-sweep linear rational finite difference
CT	Composite trapezoidal
FD	Finite difference
FDTD	Finite-difference time-domain
FIDEs	Fredholm integro-difference equations
FS	Full-sweep
FSCT	Full-sweep composite trapezoidal
FSGS	Full-sweep Gauss-Seidel
FSLBRI	Full-sweep linear barycentric rational interpolation
FSLRFD	Full-sweep linear rational finite difference
FSRGS	Full-sweep refinement of Gauss-Seidel
FSRSOR	Full-sweep refinement of successive over-relaxation
FSSOR	Full-sweep successive over-relaxation
GS	Gauss-Seidel
HS	Half-sweep
HSAGE	half-sweep alternating group explicit
HSAM	Half-Sweep Arithmetic Mean method

HSCT	Half-sweep composite trapezoidal
HSGM	Half-sweep geometric mean
HSGS	Half-sweep Gauss-Seidel
HSIADE	Half-Sweep Iterative Alternating Decomposition Explicit
HSLBRI	Half-sweep linear barycentric rational interpolation
HSLRFD	Half-sweep linear rational finite difference
HSRGS	Half-sweep refinement of Gauss-Seidel
HSRSOR	Half-sweep refinement of successive over-relaxation
HSSOR	Half-sweep successive over-relaxation
LBRI	Linear barycentric rational interpolation
LRFD	Linear rational finite difference
ODEs	Ordinary differential equations
PDEs	Partial differential equations
RGS	Refinement of Gauss-Seidel
RSOR	Refinement of successive over-relaxation
SOR	Successive over-relaxation
VIDEs	Volterra integro-differential equations

Table 3. The comparison of results for three different techniques on Example 1

Parameters	**Method**	***N***				
		32	**64**	**128**	**256**	**512**
Iterations	FSGS-3LRFD	185224	2492458	32429703	400325235	4513359199
	FSRSOR-3LRFD (ω)	4247 (1.894120000)	31012 (1.9468360000)	226096 (1.973643000)	1637449 (1.987084256)	11345038 (1.992892590)
	HSRSOR-3LRFD (ω)	589 (1.791600000)	4247 (1.894120000)	31012 (1.9468360000)	226096 (1.973643000)	1637449 (1.987084256)
***Time* (seconds)**	FSGS-3LRFD	0.4624	7.6251	232.2066	6634.2979	190132.3872
	FSRSOR-3LRFD	0.0105	0.1209	1.5437	26.3720	551.8140
	HSRSOR-3LRFD	0.0013	0.0143	0.1406	1.7230	27.1865
Error	FSGS-3LRFD	1.2910E-03	3.2510E-04	1.0914E-04	4.5008E-04	4.9842E-03
	FSRSOR-3LRFD	1.2908E-03	3.2279E-04	8.0914E-05	2.1434E-05	1.1088E-05
	HSRSOR-3LRFD	5.1588E-03	1.2908E-03	3.2279E-04	8.0914E-05	2.1434E-05

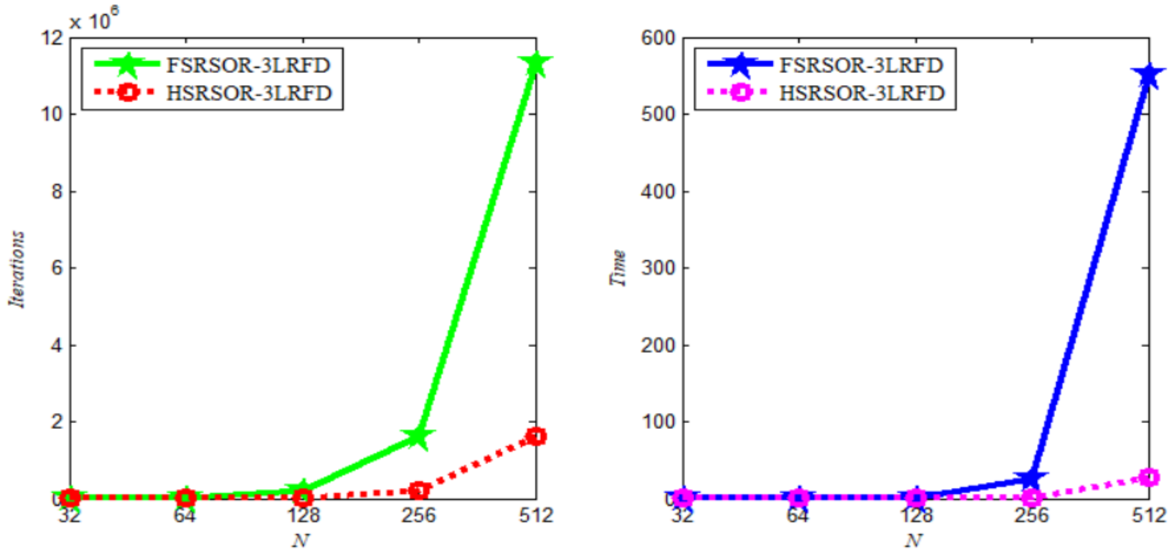

Figure 2. ***Iterations*** **and** ***Time*** (seconds) versus *N* of two different techniques for Example 1

Table 4. The comparison of results for three different techniques on Example 2

Parameters	Method	*N*				
		32	**64**	**128**	**256**	**512**
Iterations	FSGS-3LRFD	448234	5959234	76098613	910442625	9658341997
	FSRSOR-3LRFD (ω)	4996 (1.9440800000)	36507 (1.9722130000)	253489 (1.9853204000)	1936709 (1.9932668000)	13965124 (1.9967335111)
	HSRSOR-3LRFD (ω)	728 (1.7626000000)	4996 (1.9440800000)	36507 (1.9722130000)	253489 (1.9853204000)	1936709 (1.9932668000)
***Time* (seconds)**	FSGS-3LRFD	1.0256	27.3561	566.8762	14873.0139	414576.0779
	FSRSOR-3LRFD	0.0122	0.1201	4.8374	28.7981	594.5617
	HSRSOR-3LRFD	0.0023	0.0122	0.1153	1.8056	31.4941
Error	FSGS-3LRFD	5.3204E-04	1.3904E-04	1.2114E-04	1.1261E-03	1.2686E-02
	FSRSOR-3LRFD	5.3152E-04	1.3314E-04	3.2759E-05	1.1373E-05	2.3281E-05
	HSRSOR-3LRFD	2.1151E-03	5.3152E-04	1.3314E-04	3.2759E-05	1.1373E-05

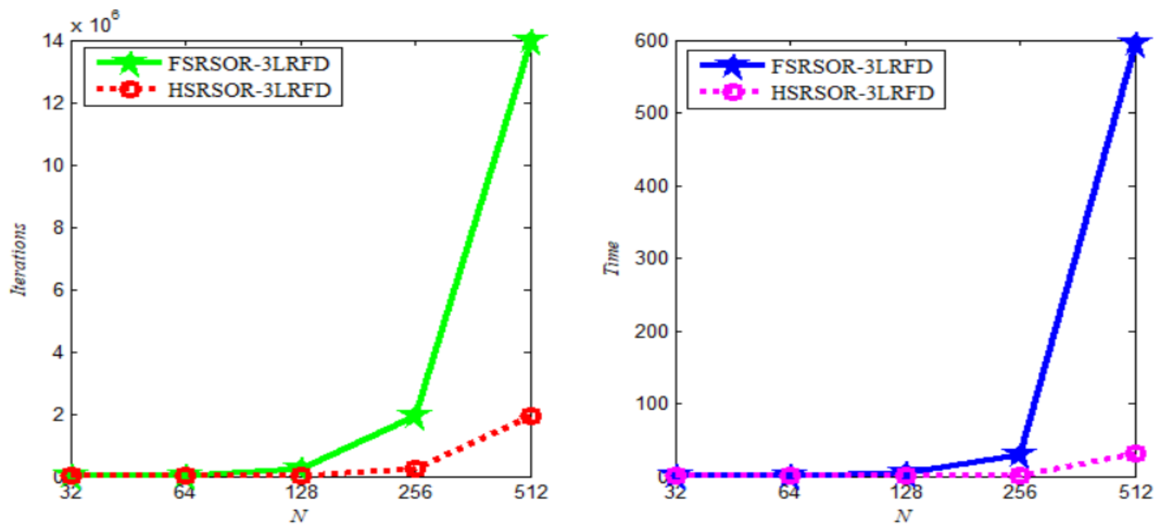

Figure 3. ***Iterations*** **and** ***Time*** (seconds) versus *N* of two different techniques for Example 2

Table 5. The comparison of results for three different techniques on Example 3

Parameters	**Method**	***N***				
		32	**64**	**128**	**256**	**512**
Iterations	FSGS-3LRFD	461118	6163295	79290805	960096418	10428247925
	FSRSOR-3LRFD (ω)	9239 (1.94229380)	66863 (1.97106190)	482119 (1.98553030)	3685270 (1.99341000)	24259791 (1.99643221)
	HSRSOR-3LRFD (ω)	1295 (1.88417900)	9239 (1.94229380)	66863 (1.97106190)	482119 (1.98553030)	3685270 (1.99341000)
***Time* (seconds)**	FSGS-3LRFD	1.0987	23.7031	622.6473	15178.6218	441013.7995
	FSRSOR-3LRFD	0.0305	0.3315	3.6553	60.0946	992.9554
	HSRSOR-3LRFD	0.0043	0.0348	0.3415	3.9125	61.6715
Error	FSGS-3LRFD	6.7632E-06	1.0129E-05	9.9643E-05	1.1235E-03	1.1207E-02
	FSRSOR-3LRFD	6.0313E-06	1.2856E-06	1.2249E-06	8.4393E-06	5.3243E-05
	HSRSOR-3LRFD	2.4195E-05	6.0313E-06	1.2856E-06	1.2249E-06	8.4394E-06

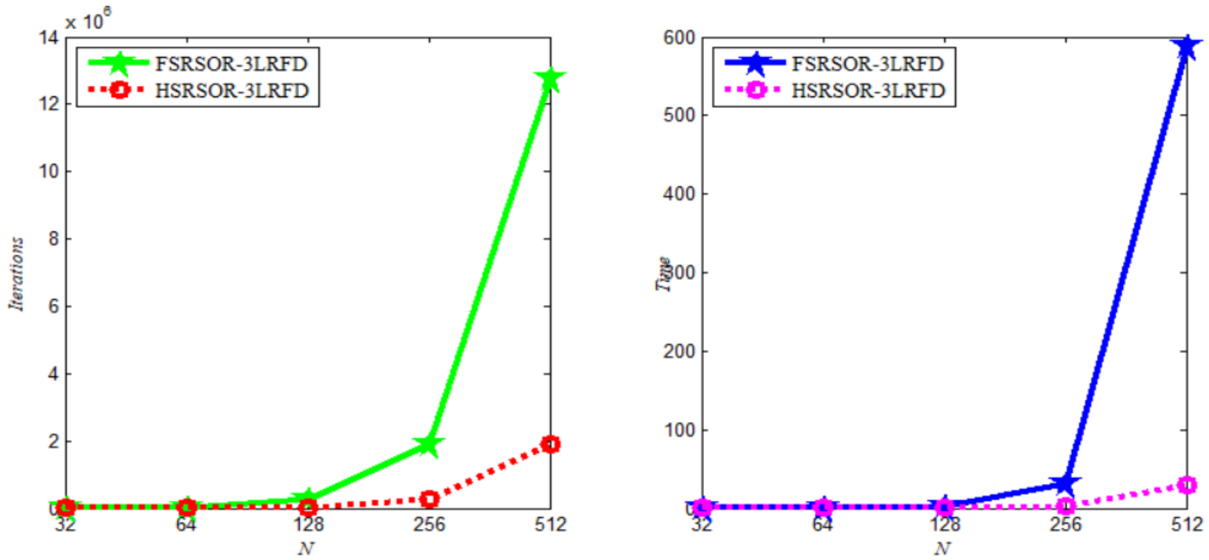

Figure 4. ***Iterations*** and ***Time*** (seconds) versus *N* of two different techniques for Example 3

Table 6. The percentage of decrement of ***Iterations*** and ***Time*** of HSRSOR and FSRSOR methods relative to FSGS method on Example 1 to Example 3

Example	**Method**	***Iterations***	***Time***
1	FSRSOR-3LRFD	97.71%-99.75%	97.73%-99.71%
	HSRSOR-3LRFD	99.68%-99.96%	99.72%-99.99%
2	FSRSOR-3LRFD	98.89%-99.86%	98.81%-99.86%
	HSRSOR-3LRFD	99.84%-99.98%	99.78%-99.99%
3	FSRSOR-3LRFD	98.96%-99.88%	98.69%-99.87%
	HSRSOR-3LRFD	99.84%-99.98%	99.80%-99.99%

REFERENCES

[1] A. M. Wazwaz, A first course in integral equations, 2th ed., World Scientific, USA, 2015.

[2] Babaei, H. Jafari and S. Banihashemi, Numerical solution of variable order fractional nonlinear quadratic integro-differential equations based on the sixth-kind Chebyshev collocation method, Journal of Computational and Applied Mathematics, Vol. 377, No. 112908, 2020.

[3] M. Efendiev and V. Vougalter, Existence of solutions for some non-Fredholm integro-differential equations with mixed diffusion, Journal of Differential Equations, Vol. 284, 83-101, 2021.

[4] P. Linz, Analytic and numerical methods for Volterra equations, SIAM, Philadelphia, Pa., USA, 1985.

[5] Z. Taheri, S. Javadi, and E. Babolian, Numerical solution of stochastic fractional integro-differential equation by the spectral collocation method, Journal of Computational and Applied Mathematics, Vol.321, 336-347, 2017.

[6] Chen, X. Zhang, G. Zhang and Y. Zhang, A two-grid finite element method for nonlinear parabolic integro-differential equations. International Journal of Computer Mathematics, Vol.96, No.10, 2010-2023, 2019.

[7] A. Arqub and B. Maayah, Fitted fractional reproducing kernel algorithm for the numerical solutions of ABC–Fractional Volterra integro-differential equations, Chaos, Solitons & Fractals, Vol.126, 394-402, 2019.

[8] H. Dehestani, Y. Ordokhani and M. Razzaghi, Pseudo-operational matrix method for the solution of variable-order fractional partial integro-differential equations, Engineering with Computers, Vol.37, 1791–1806, 2021.

[9] S. Behera and S. S. Ray, Euler wavelets method for solving fractional-order linear Volterra–Fredholm integro-differential equations with weakly singular kernels, Computational and Applied Mathematics, Vol.40, No.6, 1-30, 2021.

[10] N. Bildik, A. Konuralp and S. Yalçınbaş, Comparison of Legendre polynomial approximation and variational iteration method for the solutions of general linear Fredholm integro-differential equations. Computers & Mathematics with Applications, Vol. 59, No. 6, 1909-1917, 2010.

[11] A. Arqub, M. Al-Smadi and N. Shawagfeh. Solving Fredholm integro–differential equations using reproducing kernel Hilbert space method. Applied Mathematics and Computation, Vol. 219, No. 17, 8938-8948, 2013.

[12] H. Safdari, Y. E. Aghdam, Numerical Solution of Second-Order Linear Fredholm Integro-Differetial Equations by Trigonometric Scaling Functions, Journal of Applied Sciences, Vol. 5, No. 4, 2015

[13] P. K. Pandey, Non-standard finite difference method for numerical solution of second order linear Fredholm integro-differential equations. International Journal of Mathematical Modelling and Computations, 5(3 (SUMMER)), 259-266, 2015.

[14] M. F. Karim, M. Mohamad, M. S. Rusiman, N. Che-Him, R. Roslan and K. Khalid, ADM for solving linear second-order Fredholm integro-differential equations, In Journal of Physics: Conference Series (Vol. 995, No. 1, 012009). IOP Publishing, 2018.

[15] J. Chen, M. He, and T. Zeng, A multiscale Galerkin method for second-order boundary value problems of Fredholm integro-differential equation II: Efficient algorithm for the discrete linear system. Journal of Visual Communication and Image Representation, Vol. 58, 112-118, 2019.

[16] T. Tahernezhad and R. Jalilian, Exponential spline for the numerical solutions of linear Fredholm integro-differential equations. Advances in Difference Equations, Vol. 1, 1-15, 2021

[17] M. F. Karaaslan, A performance assessment of an HDG method for second-order Fredholm integro-differential equation: existence-uniqueness and approximation. Turkish Journal of Mathematics, Vol. 45, No. 5, 2021.

[18] R. Abdullah, The Four Point Explicit Decoupled Group (EDG) Method, A Fast Poisson Solver, International Journal of Computer Mathematics, Vol. 38, 61–70, 1991.

[19] E. Aruchunan and J. Sulaiman, Half-sweep conjugate gradient method for solving first order linear Fredholm integro-differential equations, Australian Journal of Basic and Applied Sciences, Vol. 5, No. 3, 38-43, 2011.

[20] T. K. Enyew, G. Awgichew, E. Haile and G. D. Abie, Second-refinement of Gauss-Seidel iterative method for solving linear system of equations, Ethiopian Journal of Science and Technology, Vol. 13, No. 1, 1-15, 2020.

[21] V. B. K. Vatti, S. Dominic and Sahanica, A refinement of Successive Over Relaxation (RSOR) method for solving of linear system of equations, International Journal of Advanced Information Science and Technology, Vol. 40, No. 40, 1-4, 2015.

[22] R. Abdullah and N. H. M. Ali, A comparative study of parallel strategies for the solution of elliptic pde's, International Journal of Parallel, Emergent And Distributed Systems, Vol. 10, No. 1-2, 93-103, 1996.

[23] Ibrahim and A. R. Abdullah, Solving the two dimension diffusion equation by the Four Point Explicit Decoupled Group (EDG) iterative method, International Journal of Computer Mathematics, Vol.58, No.3-4, 253-263, 1995.

[24] W. S. Yousif and D. J. Evans, Explicit de-coupled group iterative methods and their parallel implementations, Parallel Algorithms and Applications, Vol.7, No.1-2, 53-71, 1995.

[25] J. Sulaiman, M. K. Hasan and M. Othman, Red-Black Half-Sweep Iterative Method Using Triangle Finite Element Approximation for 2D Poisson Equations, In. Y. Shi et al. (Eds), Computational Science 2007, Lecture Notes in Computer Science (LNCS 4487), 326-333, 2007.

[26] J. Sulaiman, M. K. Hasan, M. Othman, M, Red-Black EDGSOR Iterative Method Using Triangle Element Approximation for 2D Poisson Equations, In. O. Gervasi & M. Gavrilova (Eds), Computational Science and Its Application 2007, Lecture Notes in Computer Science (LNCS 4707) , 298-308, 2007.

[27] J. Sulaiman, M. K. Hasan, and M. Othman, The Half-Sweep Iterative Alternating Decomposition Explicit (HSIADE) method for diffusion equations, In. J. Zhang, J.-H. He & Y. Fu (Eds), Computational and Information Science 2004, Lecture Note on Computer Science (LNCS 3314), 57-63, 2004.

[28] A. Dahalan, M. S. Muthuvalu and J. Sulaiman, Numerical solutions of two-point fuzzy boundary value problem using half-sweep alternating group explicit method, In AIP Conference Proceedings, Vol.1557, No.1, 103-107, 2013.

[29] A. Dahalan, M. S. Muthuvalu and J. Sulaiman, Performance of HSAGE method with Seikkala derivative for 2-D fuzzy Poisson equation, Applied Mathematical Sciences, Vol.8, No.17-20, 885-899, 2014.

[30] M. S. Muthuvalu and J. Sulaiman, Half-Sweep Arithmetic Mean method with composite trapezoidal scheme for solving linear Fredholm integral equations, Applied Mathematics and Computation, Vol.217, No.12, 5442-5448, 2011.

[31] M. S. Muthuvalu and J. Sulaiman, Half-sweep geometric mean iterative method for the repeated Simpson solution of second kind linear Fredholm integral equations, Proyecciones (Antofagasta), Vol.31, No.1, 65-79, 2012.

[32] M. Othman, J. Sulaiman and A. R. Abdullah, A parallel halfsweep multigrid algorithm on the shared memory multiprocessors, Malaysian Journal of Computer Science, Vol. 13, No.2, 1-6, 2000.

[33] J. Sulaiman, M. Othman, and M. K. Hasan, Half-Sweep Algebraic Multigrid (HSAMG) method applied to diffusion equations, 2008. In. H.G. Bock et al. (Eds), Modeling, Simulation and Optimization of Complex Processes, 547-556, 2008.

[34] M. K. Hasan, M. Othman, Z. Abbas, J. Sulaiman and F. Ahmad, Parallel Solution of High-Speed Low Order FDTD on 2d Free Space Wave Propagation, In. O. Gervasi & M. Gavrilova (Eds). Computational Science and Its Application 2007. Lecture Notes in Computer Science (LNCS 4706), 13-24, 2007.

[35] M. K. Hasan, M. Othman, R. Johari, Z. Abbas and J. Sulaiman, The HSLO (3)-FDTD with direct-domain and temporary-domain approaches on infinite space wave propagation, in 2005 13th IEEE International Conference on Networks Jointly held with the 2005 IEEE 7th Malaysia International Conf on Communic, Vol.2, No.6, 2005.

[36] Saudi and J. Sulaiman, Robot path planning using four point-explicit group via nine-point laplacian (4EG9L) iterative method, Procedia Engineering. Vol.41, 182-188, 2012.

[37] Saudi and J. Sulaiman, Red-black strategy for mobile robot path planning, In World Congress on Engineering. 2012. July 4-6, 2012. London, UK, 2182, 2215-2219, 2010.

[38] G. Klein and J. P. Berrut, Linear rational finite differences from derivatives of barycentric rational interpolants, SIAM J. Numer. Anal. Vol.50, No.2, 643-656, 2012.

[39] Abdi, S. A. Hosseini, The barycentric rational difference-quadrature scheme for systems of Volterra integro-differential equations, SIAM J. Sci. Comput., Vol.40, No.3, A1936–A1960, 2018.

[40] Abdi, J. P. Berrut and S. A. Hosseini, The linear barycentric rational method for a class of delay Volterra integro-differential equations, J. Sci. Comput, Vol.75, No.3, 1757–1775, 2018.

[41] Abdi, S. A. Hosseini and H. Podhaisky, Adaptive linear barycentric rational finite differences method for stiff ODEs, Journal of Computational and Applied Mathematics. Vol.357, 204–214, 2019.

[42] M. M. Xu, J. Sulaiman and L. H. Ali, Rational Finite Difference Solution of First-Order Fredholm Integro-differential Equations via SOR Iteration, Lecture Notes in Electrical Engineering. Vol.724, 463-474, 2021.

[43] M. M. Xu, J, Sulaiman and L. H. Ali, Refinement of SOR method for the rational finite difference solution of first-order Fredholm integro-differential equations, AIP Conference Proceedings, 2423, 020014, 2021.

[44] M. M. Xu, J, Sulaiman and L. H. Ali, Half-Sweep SOR Iterative Method Using Linear Rational Finite Difference Approximation for First-Order Fredholm Integro-Differential Equations, International Journal of Mathematics and Computer Science. Vol.16, No.4, 1555-1570, 2021.

[45] M. S. Floater and K. Hormann, Barycentric rational interpolation with no poles and high rates of approximation, Numerische Mathematik, Vol.107, No.2, 315-331, 2007.

[46] M. M. Xu, J. Sulaiman and L. H. Ali, Linear rational finite difference approximation for second-order linear fredholm integro-differential equations using the half-sweep SOR iterative method, International Journal of Engineering Trends and Technology, Vol.69, No.6, 136-143, 2021.

[47] J. Sunday, On exact finite difference scheme for the computation of second-order fredholm integro-differential equations, Fulafia Journal of Science and Technology. Vol.5, No.1, 113-119, 2019.

[48] L. H. Ali, J. Sulaiman, A. Saudi and M. M. Xu, Newton-SOR with Quadrature Scheme for Solving Nonlinear Fredholm Integral Equations, Lecture Notes in Electrical Engineering. Vol.724, 325-337, 2021.

[49] D. Polyanin and A. V. Manzhirov: Handbook of Integral Equations. Chapman and Hall/CRC, Boca Raton (2008)