Journal of Quality Measurement and Analysis *JQMA* 22(2) 2026, 175-188
e-ISSN: 2600-8602 *http://www.ukm.my/jqma*
https://doi.org/10.17576/jqma.2202.2026.10

QUARTER-SWEEP ITERATIVE METHODS FOR FUZZY FREDHOLM INTEGRAL EQUATIONS OF THE SECOND KIND

(*Kaedah Lelaran Sapuan Suku Bagi Persamaan Kamiran Fredholm Kabur Jenis Kedua*)

LABIYANA HANIF ALI*, NUR TASNEM JAAFFAR, HARLIZA MOHD HANIF & JUMAT SULAIMAN

ABSTRACT

This study presents a numerical solution of fuzzy Fredholm integral equations (FIEs) using the quarter-sweep Jacobi (QSJ) and Gauss-Seidel (QSGS) iterative methods. The primary goal is to apply the quarter-sweep concept in the discretisation process to reduce computational complexity in solving such equations. The methodology of this study consists of two main parts. First, we implement a quadrature scheme that incorporates with quarter-sweep concept to discretise the fuzzy FIEs, resulting in corresponding linear equations. This approach then generates a system of linear equations with respect to the proposed problem. Subsequently, QSJ and QSGS method are utilised to address the resulting system of linear equations. To demonstrate the effectiveness of this approach, several examples are tested. The results obtained highlight the effectiveness of proposed methods in addressing fuzzy FIEs.

Keywords: fuzzy Fredholm integral equations; half-sweep Gauss-Seidel; quarter-sweep Gauss-Seidel; quadrature scheme; iterative method

ABSTRAK

Kertas kerja ini membentangkan penyelesaian berangka bagi persamaan kamiran Fredholm (FIE) kabur menggunakan kaedah lelaran sapuan-suku Jacobi (QSJ) dan Gauss-Seidel (QSGS). Matlamat utama adalah untuk menggunakan konsep sapuan-suku dalam proses pendiskritan untuk mengurangkan kerumitan pengiraan dalam menyelesaikan persamaan tersebut. Metodologi kajian ini adalah terdiri daripada dua bahagian utama. Pertama, kami melaksanakan skema kuadratur yang menggabungkan konsep sapuan-suku untuk mendiskritkan FIE kabur, menghasilkan persamaan linear sepadan. Pendekatan ini kemudian menjana satu sistem persamaan linear terhadap permasalahan yang dicadangkan. Seterusnya, kaedah QSJ dan QSGS digunakan untuk menyelesaikan sistem persamaan linear yang terhasil. Untuk menunjukkan keberkesanan pendekatan ini, beberapa contoh diuji. Keputusan yang diperoleh menegaskan keberkesanan kaedah yang dicadangkan dalam menangani FIE kabur.

Kata kunci: persamaan kamiran Fredholm kabur; Gauss-Gauss-Seidel sapuan separuh; Gauss-Seidel sepuan suku; skema kuadratur; kaedah lelaran

1. Introduction

Fuzzy Fredholm integral equations (FIEs) offer a diverse range of applications in various fields, including physics, biological models, optimal control theory, mathematical economics, and engineering (Ma *et al*. 2020). Due to its diverse range of applications, fuzzy logic has been extensively developed by researchers across various fields especially engineering and medicine. For example, Bingül and Yıldız (2023) applied fuzzy logic to the suspension system of a 4 × 4 in-wheel motor-driven electric vehicles, while Suzuki and Negishi (2024) introduced a fuzzy logic–based system that incorporates approximate reasoning into medical diagnostic frameworks for diseases such as breast cancer, prostate cancer, and lung cancer. In the context of integral equations, there has been growing interest in fuzzy metric spaces in recent years. Mani *et al.* (2023) established new fixed-point results in fuzzy b-metric spaces and applied them to fuzzy integral equations (FIEs) and dynamic programming problems. Similarly, Saleem *et al.* (2023) investigated the existence and uniqueness of solutions to integral equations using intuitionistic extended fuzzy b-metric spaces. With respect to real-world applications, Kadham *et al.* (2024) demonstrated that fuzzy integrals can be employed in liquid tank systems to enhance sand-dune image detection, thereby supporting sustainable groundwater management. Meanwhile, Rasheed *et al.* (2025) proposed a mathematical model integrating fuzzy integrals to analyse solar cell behavior under varying temperature conditions.

Over the years, numerous researchers have contributed to the solutions of fuzzy FIEs using various analytical and numerical methods. Notable approaches includes residual minimisation method (Jahantigh *et al*. 2008), artificial neural networks (Fadravi *et al*. 2014), triangular and delta basis functions (Mirzaee *et al*. 2014), hybrid of block-pulse functions, as well as Taylor series method (Mirzaee 2014), hybrid of Bernstein and block-pulse functions (Mirzaee *et al*. 2015), Fuzzy quasi-interpolation (Amirfakhrian *et al*. 2017), Adomian decomposition method (Amawi & Qatanani 2016), Successive approximation method with Simpson quadrature rule (Gholam & Ezzati 2017), successive approximations method with midpoint quadrature formula (Mahaleh & Ezzati 2017), Successive Over-Relaxation (SOR) method (Ali *et al*. 2018), Sinc-collocation method (Noeiaghdam *et al*. 2020), and Homotopy analysis method (Molabahrami *et al*. 2020) and Gauss-Legendre type quadrature formulas (Bica & Ziari 2024). Among recent notable approaches, Golshan and Ezzati (2024) employed Newton–Cotes rules and successive approximation sequences to approximate nonlinear fuzzy integral equations of the second kind. More recently, Deng *et al.* (2025) introduced a fuzzy barycentric Lagrange interpolation method to solve two-dimensional fuzzy fractional Volterra integral equations.

In the recent study, Ali *et al*. (2020) studied the numerical solution concerning fuzzy FIEs of the second kind utilising half-sweep Gauss-Seidel (HSGS) method. Typically, the standard iterative methods require extensive operations as they process every node point in each iteration. However, by implementing the half-sweep method, values are computed at every alternate node point which reduces the complexity of the calculations required in resolving the fuzzy FIEs. This half-sweep iteration concept was first proposed by Abdullah (1991) to solve the Poisson equation. Following that, Othman and Abdullah (2000) extended the idea by introducing the quarter-sweep iteration concept to address the two-dimensional (2D) Poisson equation with Modified Explicit Group. This quarter-sweep method further reduces the number of grid points by computing values at every fourth node point, resulting in less computational effort compared to both full- and half-sweep methods. This approach not only lowers computational cost but also maintains convergence, making it particularly valuable in practical applications. For example, in large-scale fuzzy systems or real-time simulations, rapid approximate solutions are often more important than achieving exact results. Therefore, the quarter-sweep method enhances both the efficiency and practical applicability of fuzzy FIE solvers.

The concept regarding quarter-sweep iteration has evolved through the contributions of various researchers. This concept has been effectively applied in addressing diverse scientific problems. For example, Koh *et al*. (2010) utilised the quarter-sweep projected modified GS method to solve linear complementarity issues. Muthuvalu and Sulaiman (2011) explored the solution with regard to linear FIEs employing the quarter-sweep Gauss-Seidel (QSGS) method. Hoe and Hasan (2013) applied quarter-sweep schemes to address steady-state problems. Eng *et* *al.* (2019) implemented the quarter-sweep method in the Poisson image blending problem. Muhiddin *et al*. (2020) solved time-fractional parabolic equations utilising the quarter-sweep KSOR approach. Sunarto *et al*. (2021a) studied the solution of fractional mathematical equations using the quarter-sweep preconditioned relaxation method. Sunarto *et al*. (2021b) proposed the quarter-sweep preconditioned GS technique to solve time-fractional diffusion equations. Dahalan *et al*. (2022) investigated pathfinding solutions for mobile robot navigation by employing the quarter-sweep modified AOR method. Chew *et al.* (2022) applied the quarter-sweep SOR method to solve porous medium equations. In the recent study, Ali *et al*. (2023) addressed nonlinear FIEs utilising the quarter-sweep Newton-PKSOR approach. These studies highlight the versatility and efficiency of the quarter-sweep technique in tackling diverse scientific challenges. Motivated by the advancements of the quarter-sweep method in addressing various problems, this study proposes the QSJ and QSGS iterative methods for solving fuzzy FIEs, aiming to balance computational efficiency with solution accuracy.

The research is structured with Section 2 discussing the methodology employed in this research to address fuzzy FIEs. It includes a detailed formulation of the quadrature discretisation scheme and the quarter-sweep iterative methods to address fuzzy FIEs. Then, the numerical examples are presented in Section 3 and provide a comprehensive discussion of the results obtained. This part aims to illustrate the efficiency concerning the suggested approaches via various examples. In the final section, concluding remarks are given, summarising the key findings of this study.

2. Methodology

This section presents the formulation of fuzzy Fredholm integral equations using a first-order quadrature scheme, chosen for its numerical stability (Nouriani & Ezzati 2017) as well as its lower computational cost compared to higher-order schemes, which typically require greater computational effort (Lloret-Cabot & Sheng 2022).

2.1. *Formulation of Fuzzy FIEs using Quadrature Scheme*

Consider the fuzzy FIEs of the second kind in the following form

$$\tilde{x}(s,r) = \tilde{g}(s,r) + \lambda \int_a^b K(s,t)\tilde{x}(t,r)dt \tag{1}$$

with $\tilde{x}(s,r) = \left(\underline{x(s,r)}, \overline{x(s,r)}\right)$ and $\tilde{g}(s,r) = \left(\underline{g(s,r)}, \overline{g(s,r)}\right)$ on $0 \leq r \leq 1$. This study focusses on solving the following equation

$$\tilde{x}(s,r) - \int_a^b K(s,t)\tilde{x}(t,r)dt = \tilde{g}(s,r) \tag{2}$$

where $\lambda = 1$. Suppose $\tilde{f}(t,r) = \left(\underline{f(t,r)}, \overline{f(t,r)}\right)$, the general formulation of quadrature scheme for fuzzy function $\tilde{f}(t,r)$ may be expressed as

$$\underline{\left(\int_a^b f(t,r)dt\right)} = \sum_{j=0}^{n} A_j \underline{f\left(t_j,r\right)} + \underline{\varepsilon_n(t)} \tag{3}$$

$$\overline{\left(\int_a^b f(t,r)dt\right)} = \sum_{j=0}^{n} A_j \overline{f\left(t_j,r\right)} + \overline{\varepsilon_n(t)} \tag{4}$$

with A_j is numerical coefficient, t_j refers to the abscissa of the integrations partition points on the interval $[a, b]$ while $\tilde{\varepsilon}_n(t) = \left(\underline{\varepsilon_n(t)}, \overline{\varepsilon_n(t)}\right)$ denotes the truncation error. In general, Eq. (3) and (4) then may be expressed as follows

$$\int_a^b \tilde{f}(t, r)dt = \sum_{j=0}^{n} A_j \tilde{f}(t_j, r) + \tilde{\varepsilon}_n(t) \tag{5}$$

where coefficient A_j is given by

$$A_j = \begin{cases} \frac{1}{2}ph, & j = 0, n \\ ph, & \text{otherwise} \end{cases} \tag{6}$$

The value p takes value of 1, 2, and 4 for the full-sweep, half-sweep, as well as quarter-sweep cases, respectively. This parameter p determines the placement of node point of type ●. Meanwhile, h represents a constant step size, described as follows

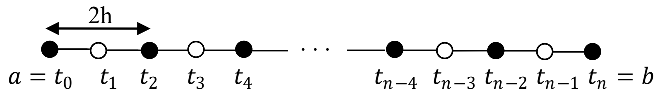
$$h = \frac{b-a}{n} \tag{7}$$

Consider the definite fuzzy integral of a function $\tilde{f}(t, r)$ on an interval $[a, b]$ as follows

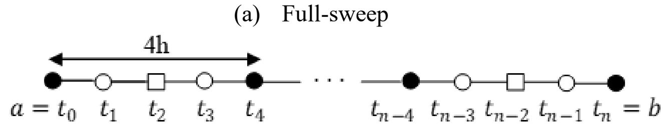
$$\int_a^b \tilde{f}(t, r)dt = \int_{t_0}^{t_n} \tilde{f}(t, r)dt \tag{8}$$

The interval $[a, b]$ can be divided into n equidistant subintervals $\{t_0, t_1, t_2, \ldots, t_n\}$ as $a = t_0 < t_1 < t_2 < \ldots < t_{n-1} < t_n = b$.

The grid points are distributed over a solution domain using half-sweep along with quarter-sweep approaches is illustrated in Fig. 1. These approaches employ different strategies for placing node points within the interval $[a, b]$. The node points demonstrate the calculation method for obtaining the approximate value at each point. In the full-sweep approach, grid points are distributed uniformly across the entire interval $[a, b]$ using node points of type ●. Then, the half-sweep method decreases the number of grid points by roughly half using node points of type ○ as the midpoints. The quarter-sweep method further decreases the grid points number by computing values at every fourth node point, placing node points of type ○ and □ in between node points of type ●. The approximate value at type ● node points is obtained through an iterative process, while the values at type ○ and □ node points are obtained using a direct method.



(a) Full-sweep



(b) Half-sweep

Figure 1: The arrangement concerning grid points within the solution domain utilising half- as well as quarter-sweep approaches

Let a and b on the interval $[a,b]$ as $a=t_0$, $b=t_n$ with t_i, $i=0,p,2p,\dots,n-p,n$, thus integration part by part on Eq. (8) is given as follows

$$\int_{t_0}^{t_n}\tilde{f}(t,r)dt=\int_{t_0}^{t_p}\tilde{f}(t,r)dt+\int_{t_p}^{t_{2p}}\tilde{f}(t,r)dt+\cdots+\int_{t_{n-p}}^{t_n}\tilde{f}(t,r)dt \tag{9}$$

Next, implement the Trapezoidal scheme on Eq. (6) towards Eq. (9) to obtain the equation as follows

$$\int_{t_0}^{t_n}\tilde{f}(t,r)dt=\frac{ph}{2}\left(\tilde{f}_{0,r}+\tilde{f}_{p,r}\right)+\frac{ph}{2}\left(\tilde{f}_{p,r}+\tilde{f}_{2p,r}\right)+\cdots+\frac{ph}{2}\left(\tilde{f}_{n-p,r}+\tilde{f}_{n,r}\right) \tag{10}$$

with $\tilde{f}(t_i,r)=\tilde{f}_{i,r}$, $i=0,p,2p,\dots,n-p,n$. Consider Eq. (11) towards Eq. (3) to get the following approximation equation

$$\tilde{x}(s,r)-\int_{t_0}^{t_p}K(s,t)\tilde{x}(t,r)dt-\int_{t_p}^{t_{2p}}K(s,t)\tilde{x}(t,r)dt-\cdots-\int_{t_{n-p}}^{t_n}K(s,t)\tilde{x}(t,r)dt$$
$$=\tilde{g}(s,r) \tag{11}$$

This leads to the following equations

$$\tilde{x}(s,r)-\frac{ph}{2}\left[K(s,t_0)\tilde{x}(t_0,r)+K(s,t_p)\tilde{x}(t_p,r)\right]-\frac{ph}{2}\left[K(s,t_p)\tilde{x}(t_p,r)+\right.$$
$$\left.K(s,t_{2p})\tilde{x}(t_{2p},r)\right]-\cdots-\frac{ph}{2}\left[K(s,t_{n-p})\tilde{x}(t_{n-p},r)+K(s,t_n)\tilde{x}(t_n,r)\right]=\tilde{g}(s,r) \tag{12}$$

and

$$\tilde{x}(s,r)-\frac{ph}{2}\left(K(s,t_0)\tilde{x}(t_0,r)\right)-ph\left(K(s,t_p)\tilde{x}(t_p,r)\right)-ph\left(K(s,t_{2p})\tilde{x}(t_{2p},r)\right)-$$
$$\cdots-\frac{ph}{2}\left(K(s,t_n)\tilde{x}(t_n,r)\right)=\tilde{g}(s,r) \tag{13}$$

Now, assuming that $\tilde{x}(s_i,r)=\tilde{x}_{i,r}$, $\tilde{g}(s_i,r)=\tilde{g}_{i,r}$, $k(s_i,t_j)=k_{i,j}$, and $\tilde{x}(t_{\mathrm{i}},r)=\tilde{x}_{i,r}$, Eq. (13) simplified to

$$\tilde{x}_{i,r}-\frac{1}{2}(ph)K_{i,0}\tilde{x}_{0,r}-(ph)K_{i,p}\tilde{x}_{p,r}-(ph)K_{i,2p}\tilde{x}_{2p,r}-\cdots-\frac{1}{2}(ph)K_{i,n}\tilde{x}_{n,r}=\tilde{g}_{i,r} \tag{14}$$

with $i=0,p,2p,3p,\dots,n-p,n$. Following that, a system with regards to linear fuzzy FIEs of the second kind can be generated

$$A\underline{\tilde{x}}=\underline{\tilde{g}} \tag{15}$$

where

$$A=\begin{bmatrix}1-A_0(K_{0,0}) & -A_p(K_{0,p}) & -A_{2p}(K_{0,2p}) & \cdots & -A_n(K_{0,n})\\ -A_0(K_{p,0}) & 1-A_p(K_{p,p}) & -A_{2p}(K_{p,2p}) & \cdots & -A_n(K_{p,n})\\ -A_0(K_{2p,0}) & -A_p(K_{2p,p}) & 1-A_{2p}(K_{2p,2p}) & \cdots & -A_n(K_{2p,n})\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ -A_0(K_{n,0}) & -A_p(K_{n,p}) & -A_{2p}(K_{n,2p}) & \cdots & 1-A_n(K_{n,n})\end{bmatrix}_{\left(\frac{n}{p}+1\right)\times\left(\frac{n}{p}+1\right)}$$

$$\underline{\tilde{x}} = \begin{bmatrix} \tilde{x}_{0,r} & \tilde{x}_{p,r} & \tilde{x}_{2p,r} & \cdots & \tilde{x}_{n,r} \end{bmatrix}^T$$
$$\underline{\tilde{g}} = \begin{bmatrix} \tilde{g}_{0,r} & \tilde{g}_{p,r} & \tilde{g}_{2p,r} & \cdots & \tilde{g}_{n,r} \end{bmatrix}^T$$

2.2. *Formulation of Quarter-Sweep Gauss-Seidel Method*

The subsequent decomposition derives the coefficient matrix A

$$A = D - L - U \tag{16}$$

in which D denotes a diagonal matrix, while L and U represent a strictly lower and upper triangular matrix, respective. The GS approach to compute the linear system is defined by

$$\underline{\tilde{x}}^{(k+1)} = (D - L)^{-1} U \underline{\tilde{x}}^{(k)} + (D - L)^{-1} \underline{\tilde{g}} \tag{17}$$

Subsequently, the algorithm for the QSGS method solve fuzzy FIEs can be outlined as follows:

Step 1: Set the initial guess $\underline{\tilde{x}}^{(0)} = 0$ and convergence tolerance $\varepsilon = 10^{-10}$.
Step 2: Calculate the coefficient matrix A and vector $\underline{\tilde{g}}$.
Step 3: For each iteration $k = 1,2,3, \ldots$, calculate

$$\tilde{x}_i^{(k+1)} \leftarrow \begin{cases} \frac{1}{A_i K_{i,i}} \left(\tilde{g}_i - \sum_{j=4}^{n} A_j K_{i,j} \tilde{x}_j^{(k)} \right) & i = 0 \\ \frac{1}{A_i K_{i,i}} \left(\tilde{g}_i - \sum_{j=0}^{n-4} A_j K_{i,j} \tilde{x}_j^{(k+1)} \right), & i = n \\ \frac{1}{A_i K_{i,i}} \left(\tilde{g}_i - \sum_{j=0}^{i-4} A_j K_{i,j} \tilde{x}_j^{(k+1)} - \sum_{j=i+4}^{n} A_j K_{i,j} \tilde{x}_j^{(k)} \right), & otherwise \end{cases}$$

where $i = 0,4,8, \ldots, n-8, n-4, n$ and $j = 0,4,8, \ldots, n-8, n-4, n$.
Step 4: Perform convergence test, $\left| \tilde{x}_i^{(k+1)} - \tilde{x}_i^{(k)} \right| \leq \varepsilon = 10^{-10}$. Provided that the convergence criterion is met, proceed to step 4. If not, return to step 3.
Step 5: Calculate the estimated values for node points ○ and □ using the direct method.
Step 6: Present the approximate solution.

All numerical algorithms were implemented in C programming and executed on a standard laptop computing environment with an Intel-based processor and sufficient memory, and no specialised hardware acceleration was used. MATLAB was used exclusively for visualisation of the numerical results. Based on the proposed algorithm, the total arithmetic operations involved per iteration can be analyse for the QSGS method. Table 1 compares QSGS with the full- (FSGS) as well as HSGS approaches. In terms of computational complexity, it is shown that the QSGS method significantly reduces the total number of arithmetic operations required per iteration. Specifically, the quarter-sweep method can decrease the total computational workload to approximately one-fourth of that required by the FSGS method.

Table 1: Total arithmetic operations involved per iteration for full-, half-, and quarter-sweep Gauss-Seidel

Method	Arithmetic Operation	
	$+$ and $-$	$\times$ and $\div$
FSGS	$2(n+1)(n+1)$	$2(n+2)(n+1)$
HSGS	$2\left(\frac{n}{2}+1\right)\left(\frac{n}{2}+1\right)$	$2\left(\frac{n}{2}+2\right)\left(\frac{n}{2}+1\right)$
QSGS	$2\left(\frac{n}{4}+1\right)\left(\frac{n}{4}+1\right)$	$2\left(\frac{n}{4}+2\right)\left(\frac{n}{4}+1\right)$

3. Results and Discussion

This section presents the performance of QSJ and QSGS methods to solve various test problems. The results are examined based on the iteration number, iteration time, and in terms of accuracy using the Hausdorff distance (HD). To demonstrate the effectiveness of these iterative methods, comparative results are presented by comparing with the HSGS method studied by Ali *et al.* (2020) as well as the half-sweep Jacobi (HSJ). These methods were tested using Examples 1, 2, and 3 across grid points, $m = 2^g$ with $g = 8,9,10,11,12$. These choices allow us to observe the trade-off between computational efficiency and solution accuracy by covering from smaller to larger grid points. The tolerance level for the iterative methods was set to 10^{-10} to ensure sufficient accuracy, a common practice in similar studies (Ali *et al.* 2020). This study also considers three different fuzzy parameter values, r of 1.0, 0.6, and 0.3 which are chosen for comparison with the HSJ and HSGS.

Example 1. Consider the following problem (Gholam & Ezzati 2017)

$$\underline{f}(t,r) = -\frac{1}{52}r(5 - 52t + 2t^2) \tag{18}$$

$$\bar{f}(t,r) = \frac{1}{52}(r - 2)(5 - 52t + 2t^2) \tag{19}$$

and Kernel

$$k(s,t) = \frac{s^2+t^2+2}{13}, 0 \leq s,t \leq 1 \text{ and } \lambda = 1 \tag{20}$$

with $a = 0, b = 1$. The following are the exact solutions

$$\underline{u}(t,r) = rt \text{ and } \overline{u}(t,r) = (2 - r)t \tag{21}$$

Example 2. The second example is given as follows (Gholam & Ezzati 2017)

$$\underline{f}(t,r) = \frac{2}{3}(2 + r)t \tag{22}$$

$$\bar{f}(t,r) = -\frac{2}{3}(r - 4)t \tag{23}$$

and Kernel

$$k(s,t) = st, 0 \leq s,t \leq 1 \text{ and } \lambda = 1 \tag{24}$$

where the exact solutions can be written as

$$\underline{u}(t,r) = (2 + r)t \text{ and } \overline{u}(t,r) = (4 - r)t \tag{25}$$

Example 3. Lastly, consider the following problem (Mirzaee *et al.* 2014)

$$\underline{f}(t,r) = rt(r^4 + 2)(3cos(1 - t) + 5sin(1 - t) - 6cos(t) + t^2) \tag{26}$$

$$\bar{f}(t,r) = -3t(r^3 - 2)(3cos(1 - t) + 5sin(1 - t) - 6cos(t) + t^2) \tag{27}$$

and Kernel

$$k(s,t) = s\cos(t-s), 0 \leq s, t \leq 1 \text{ and } \lambda = 1 \quad (28)$$

with $a = 0, b = 1$. For which the exact solutions are expressed as follows

$$\underline{u}(t,r) = t^3(r^5 + 2r) \text{ and } \overline{u}(t,r) = t^3(6 - 3r^3) \quad (29)$$

The tabulated results given in the Tables 2 to 4 present the numerical findings of all methods tested to solve fuzzy FIEs. The results indicate that the QSGS method consistently achieves the lowest iteration time, demonstrating the computational advantage of the quarter-sweep approach. For instance, in Examples 1 and 2, the QSGS method requires less than one second to solve a problem with a grid size of 256, highlighting its efficiency in handling fuzzy FIEs. However, this speed comes at a cost: the QSGS method initially exhibits lower accuracy compared to HSJ and HSGS, as measured by the Hausdorff distance. This observation provides a clear trade-off between computational efficiency and solution accuracy, which is a common consideration in iterative methods for fuzzy integral equations. Meanwhile, Fig. 2 and 3 illustrate the convergence behavior of the HSJ, QSJ, HSGS and QSGS, methods by plotting the Hausdorff distance (HD) against the mesh size m for different values of the parameter r across Examples 1–3. For all test problems and parameter settings, the HD decreases monotonically as the mesh size increases, indicating that proposed methods converge as the grid size becomes larger.

Notably, the accuracy of QSGS improves substantially with increasing grid sizes, achieving results comparable to HSJ and HSGS at a grid size of 4096. This behavior reflects the convergence characteristics of the chosen quadrature scheme and quarter-sweep iterative approach: while using fewer iterations and smaller grids makes the method faster, using larger grids is needed to get more accurate solutions. It is important to note that at small grid sizes, the lower accuracy may limit solution reliability for some applications, representing an inherent limitation of the method. Regarding scalability, the quarter-sweep framework reduces the number of operations per iteration, suggesting that it can be extended to larger-scale systems or time-sensitive applications where approximate solutions are acceptable, with grid refinement used when higher accuracy is required. These findings provide useful guidance for applying QSGS and QSJ methods in real-world fuzzy systems, where computational efficiency and accuracy requirements must be balanced.

4. Conclusions

In this study, we present the QSJ and QSGS iterative methods for approximating solution with regard to fuzzy FIEs. The presented methods extend the solution of the HSGS approach for addressing fuzzy FIEs studies by Ali *et al.* (2020). By computing values at every alternate node point, the HSGS method significantly reduces computational effort while still maintaining a reasonable level of accuracy in the iterative method. The QSGS method advances this approach further by computing values at every fourth node point. This approach significantly decreases computational effort but resulting in less accuracy compared to the HSGS method. However, the high accuracy can be recorded at a grid size 4096. Thus, this study concludes that the QSGS method proves to be efficient in addressing fuzzy FIEs of the second kind.

Table 2: Numerical results of iterative methods for Example 1.

m	Method	$r = 1.0$			$r = 0.6$			$r = 0.3$		
		k	t	HD	k	t	HD	k	t	HD
256	HSJ	32	0.02	1.55e-06	31	0.02	2.17e-06	31	0.02	2.64e-06
	QSJ	32	0.01	6.21e-06	31	0.01	8.70e-06	31	0.01	1.06e-05
	HSGS	22	0.02	1.55e-06	22	0.02	2.17e-06	21	0.02	2.64e-06
	QSGS	22	0.00	6.21e-06	22	0.00	8.70e-06	21	0.00	1.06e-05
512	HSJ	32	0.07	3.88e-07	31	0.07	5.44e-07	31	0.07	6.60e-07
	QSJ	32	0.02	1.55e-06	31	0.02	2.17e-06	31	0.02	2.64e-06
	HSGS	22	0.05	3.88e-07	22	0.05	5.44e-07	21	0.05	6.60e-07
	QSGS	22	0.01	1.55e-06	22	0.02	2.17e-06	21	0.02	2.64e-06
1024	HSJ	32	0.26	9.71e-08	31	0.24	1.36e-07	31	0.25	1.65e-07
	QSJ	32	0.07	3.88e-07	31	0.07	5.44e-07	31	0.07	6.60e-07
	HSGS	22	0.18	9.71e-08	22	0.18	1.36e-07	21	0.17	1.65e-07
	QSGS	22	0.04	3.88e-07	22	0.05	5.44e-07	21	0.06	6.60e-07
2048	HSJ	32	0.99	2.43e-08	31	0.98	3.40e-08	31	0.96	4.12e-08
	QSJ	32	0.26	9.71e-08	31	0.25	1.36e-07	31	0.24	1.65e-07
	HSGS	22	0.69	2.43e-08	22	0.68	3.40e-08	21	0.66	4.13e-08
	QSGS	22	0.17	9.71e-08	22	0.19	1.36e-07	21	0.17	1.65e-07
4096	HSJ	32	3.93	6.06e-09	31	3.70	8.48e-09	31	3.77	1.03e-08
	QSJ	32	1.00	2.43e-08	31	0.97	3.40e-08	31	0.97	4.12e-08
	HSGS	22	2.74	6.07e-09	22	2.67	8.50e-09	21	2.58	1.03e-08
	QSGS	22	0.69	2.43e-08	22	0.70	3.40e-08	21	0.66	4.13e-08

[a]number of iteration (k), iteration time (t) in seconds, and Hausdorff distance (HD).

Table 3: Numerical results of iterative methods for Example 2.

m	Method	$r = 1.0$			$r = 0.6$			$r = 0.3$		
		k	t	HD	k	t	HD	k	t	HD
256	HSJ	46	0.02	4.58e-05	46	0.03	5.19e-05	46	0.03	5.65e-05
	QSJ	46	0.00	1.83e-04	46	0.00	2.08e-04	46	0.00	2.26e-04
	HSGS	28	0.01	4.58e-05	28	0.02	5.19e-05	28	0.02	5.65e-05
	QSGS	28	0.00	1.83e-04	28	0.00	2.08e-04	28	0.00	2.26e-04
512	HSJ	46	0.08	1.14e-05	46	0.09	1.30e-05	46	0.08	1.41e-05
	QSJ	46	0.03	4.58e-05	46	0.03	5.19e-05	46	0.03	5.65e-05
	HSGS	28	0.05	1.14e-05	28	0.06	1.30e-05	28	0.05	1.41e-05
	QSGS	28	0.01	4.58e-05	28	0.02	5.19e-05	28	0.02	5.65e-05
1024	HSJ	46	0.30	2.86e-06	46	0.30	3.24e-06	46	0.31	3.53e-06
	QSJ	46	0.09	1.14e-05	46	0.08	1.30e-05	46	0.08	1.41e-05
	HSGS	28	0.19	2.86e-06	28	0.19	3.24e-06	28	0.18	3.53e-06
	QSGS	28	0.06	1.14e-05	28	0.06	1.30e-05	28	0.06	1.41e-05
2048	HSJ	46	1.16	7.15e-07	46	1.16	8.11e-07	46	1.17	8.82e-07
	QSJ	46	0.30	2.86e-06	46	0.29	3.24e-06	46	0.30	3.53e-06
	HSGS	28	0.72	7.15e-07	28	0.73	8.11e-07	28	0.73	8.82e-07
	QSGS	28	0.18	2.86e-06	28	0.19	3.24e-06	28	0.18	3.53e-06
4096	HSJ	46	4.59	1.79e-07	46	4.57	2.03e-07	46	4.60	2.20e-07
	QSJ	46	1.18	7.15e-07	46	1.18	8.11e-07	46	1.17	8.82e-07
	HSGS	28	2.85	1.79e-07	28	2.82	2.03e-07	28	2.85	2.21e-07
	QSGS	28	0.72	7.15e-07	28	0.72	8.11e-07	28	0.72	8.82e-07

Table 4: Numerical results of iterative methods for Example 3.

m	Method	$r = 1.0$			$r = 0.6$			$r = 0.3$		
		k	t	HD	k	t	HD	k	t	HD
256	HSJ	64	0.30	7.89e-05	63	0.28	1.41e-04	63	0.28	1.56e-04
	QSJ	62	0.08	3.17e-04	62	0.07	5.66e-04	61	0.07	6.26e-04
	HSGS	38	0.18	7.89e-05	37	0.16	1.41e-04	37	0.17	1.56e-04
	QSGS	38	0.05	3.17e-04	37	0.05	5.66e-04	36	0.04	6.26e-04
512	HSJ	64	1.12	1.97e-05	64	1.11	3.52e-05	63	1.10	3.89e-05
	QSJ	64	0.28	7.90e-05	63	0.28	1.41e-04	63	0.27	1.56e-04
	HSGS	38	0.67	1.97e-05	37	0.64	3.52e-05	37	0.64	3.89e-05
	QSGS	38	0.17	7.90e-05	37	0.16	1.41e-04	37	0.16	1.56e-04
1024	HSJ	64	4.42	4.92e-06	64	4.44	8.78e-06	63	4.45	9.71e-06
	QSJ	64	1.12	1.97e-05	64	1.10	3.52e-05	63	1.09	3.89e-05
	HSGS	38	2.62	4.92e-06	37	2.59	8.78e-06	37	2.58	9.71e-06
	QSGS	38	0.68	1.97e-05	37	0.65	3.52e-05	37	0.65	3.89e-05
2048	HSJ	64	16.81	1.23e-06	64	17.23	2.20e-06	63	15.99	2.43e-06
	QSJ	64	4.72	4.93e-06	64	4.36	8.79e-06	63	4.30	9.72e-06
	HSGS	38	10.29	1.23e-06	37	10.02	2.20e-06	37	10.17	2.43e-06
	QSGS	38	2.63	4.93e-06	37	2.54	8.79e-06	37	2.55	9.72e-06
4096	HSJ	64	48.81	3.08e-07	64	59.32	5.49e-07	63	51.57	6.07e-07
	QSJ	64	17.33	1.23e-06	64	17.16	2.20e-06	63	17.11	2.43e-06
	HSGS	38	38.54	3.08e-07	37	37.73	5.49e-07	37	35.32	6.07e-07
	QSGS	38	10.40	1.23e-06	37	9.26	2.20e-06	37	9.95	2.43e-06

References

Abdullah A.R. 1991. The four point-Explicit Decoupled Group (EDG) method: A fast poisson solver. *International Journal of Computer Mathematics* **38**(1-2): 61-70.

Ali L.H., Sulaiman J. & Hashim S.R.M. 2018. Numerical solution of SOR iterative method for fuzzy Fredholm integral equations of second kind. *AIP Conference Proceedings*, pp. 1-8.

Ali L.H., Sulaiman J. & Hashim S. R. M. 2020. Numerical solution of fuzzy Fredholm integral equations of second kind using half-sweep Gauss-Seidel iteration. *Journal of Engineering Science and Technology* **15**(5): 3303 – 3313.

Ali L.H., Sulaiman J. & Saudi A. 2023. Iterative Method for Solving Nonlinear Fredholm Integral Equations Using Quarter-Sweep Newton-PKSOR Method. *Proceedings of the 9th International Conference on Computational Science and Technology*, pp. 33–46.

Amawi M. & Qatanani N. 2016. Analytical methods for solving fuzzy Fredholm integral equation of the second kind. *Research Journal of Applied Sciences* **11**(12): 1559-1568.

Amirfakhrian M., Shakibi K. & Rodríguez López R. 2017. Fuzzy quasi-interpolation solution for Fredholm fuzzy integral equations of second kind. *Soft Computing* **21**: 4323-4333..

Bica A.M. & Ziari S. 2024. Gauss-Legendre type quadrature iterative method for fuzzy Fredholm integral equations. *Fuzzy Sets and Systems* **477**: 108823.

Bingül Ö. & Yıldız A. 2023. Fuzzy logic and proportional integral derivative based multi-objective optimization of active suspension system of a 4× 4 in-wheel motor driven electrical vehicle. *Journal of Vibration and Control* **29**(5-6): 1366-1386.

Chew J.V.L., Aruchunan E., Sunarto A. & Sulaiman J. 2022. Quarter-Sweep Successive Over-relaxation Approximation to the Solution of Porous Medium Equations. *IAENG International Journal of Applied Mathematics* **52**(2): 1-7.

Dahalan A.A., Saudi A. & Sulaiman J. 2022. Pathfinding for mobile robot navigation by exerting the quarter-sweep modified accelerated overrelaxation (QSMAOR) iterative approach via the Laplacian operator. *Modelling and Simulation in Engineering* **2022**: 9388146.

Deng T., Huang J., Wang Y. & Li H. 2025. A generalized fuzzy barycentric Lagrange interpolation method for solving two-dimensional fuzzy fractional Volterra integral equations. *Numerical Algorithms* **98**: 743-766.

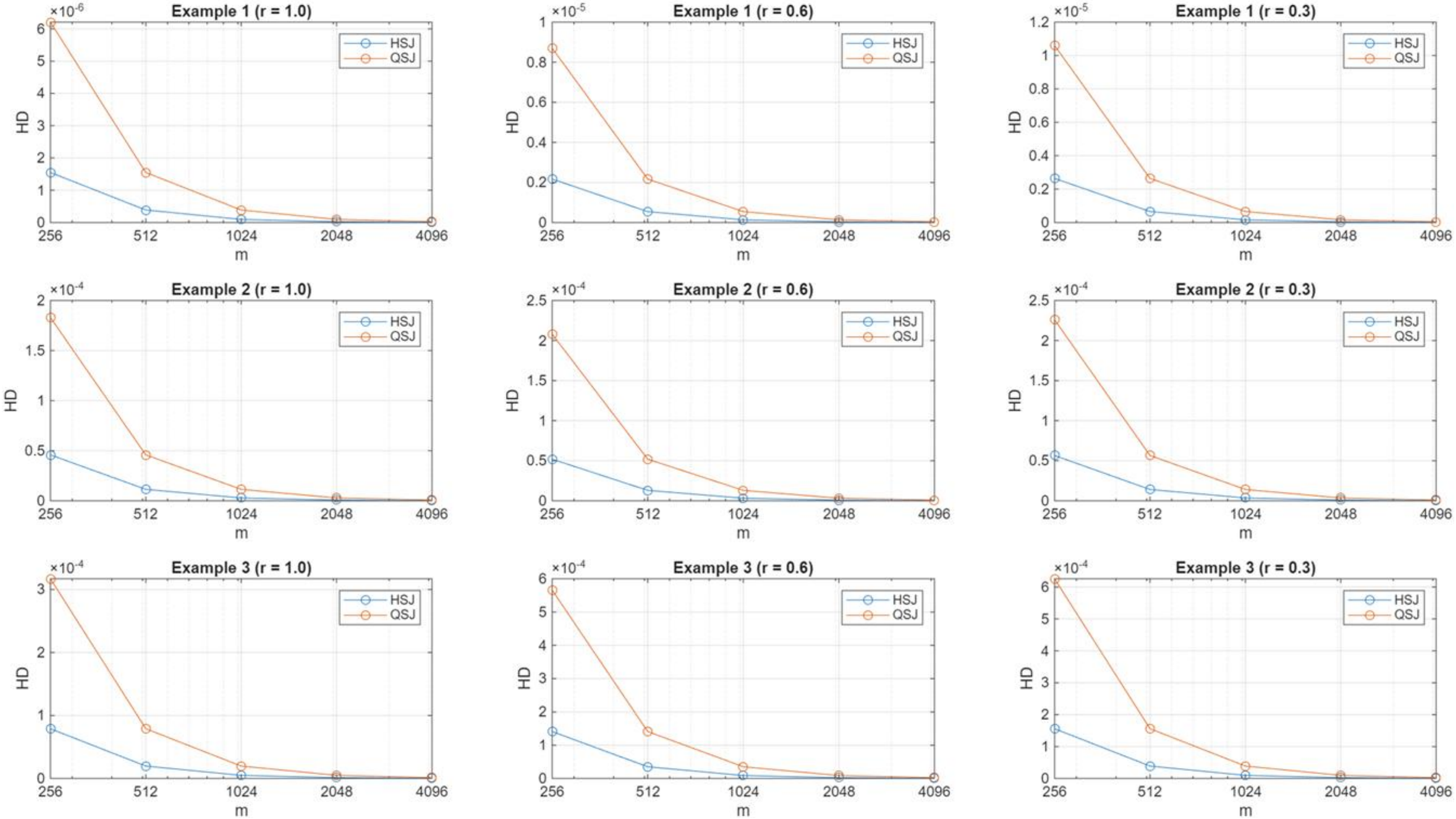


Figure 2: The graph of Hausdorff Distance (HD) vs mesh size (m) for HSJ and QSJ

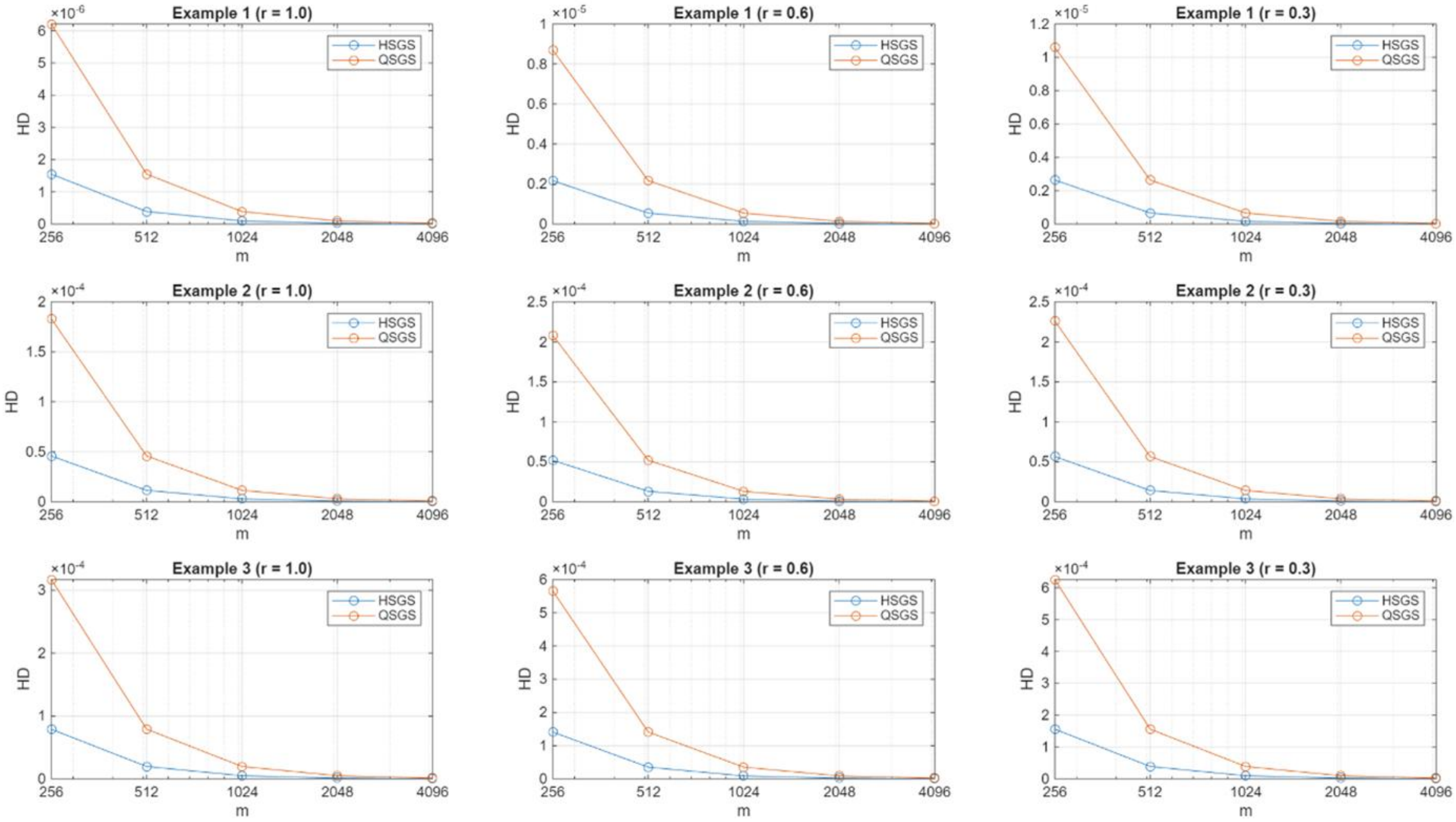


Figure 3: The graph of Hausdorff Distance (HD) vs mesh size (m) for HSGS and QSGS

Eng J.H., Saudi A. & Sulaiman J. 2019. Implementation of quarter-sweep approach in poisson image blending problem. *Computational Science and Technology. Lecture Notes in Electrical Engineering*: 127–136. Springer: Singapore.

Fadravi H.H., Buzhabadi R. & Nik H.S. 2014. Solving linear Fredholm fuzzy integral equations of the second kind by artificial neural networks. *Alexandria Engineering Journal* **53**: 249–257.

Gholam A.M. & Ezzati R. 2017. Solving linear fuzzy Fredholm integral equations of the second kind via iterative method and Simpson quadrature rule: A review. *TWMS Journal of Pure and Applied Mathematics* **8**(2): 121-147.

Golshan H.M. & Ezzati R. 2024. Successive approximation method to solve nonlinear fuzzy Fredholm integral equations using NC rules. *Soft Computing* **28**: 7649–7657.

Hoe N.Y. & Hasan M.K. 2013. Investigation of steady state problems via quarter-sweep schemes. *Sains Malaysiana* **42**(6): 837-844.

Jahantigh M., Allahviranloo T. & Otadi M. 2008. Numerical solution of fuzzy integral equations. *Applied Mathematical Sciences* **2**(1): 33-46.

Kadham S. M., Mustafa M.A., Abbass N.K. & Karupusamy S. 2024. IoT and artificial intelligence–based fuzzy-integral N-transform for sustainable groundwater management. *Applied Geomatics* **16**: 1-8.

Koh W.S., Sulaiman J. & Mail R. 2010. Quarter-sweep projected modified gauss-seidel algorithm applied to linear complementarity problem. *American Journal of Applied Sciences* **7**(6): 790-794.

Lloret-Cabot M. & Sheng D. 2022. Assessing the accuracy and efficiency of different order implicit and explicit integration schemes. *Computers and Geotechnics* **141**: 104531.

Ma Y., Li, H. & Zhang S. 2020. Solving two-dimensional fuzzy Fredholm integral equations via sinc collocation method. *Advances in Difference Equations* **2020**: 1-19.

Mahaleh V.S.K. & Ezzati R. 2017. Numerical solution of linear fuzzy Fredholm integral equations of second kind using iterative method and midpoint quadrature formula. *Journal of Intelligent and Fuzzy Systems* **33**(3): 1293-1302.

Mani G., Gnanaprakasam A.J., Guran L., George R. & Mitrović, Z. D. 2023. Some Results in Fuzzy *b*-Metric Space with *b*-Triangular Property and Applications to Fredholm Integral Equations and Dynamic Programming. *Mathematics* **11**(19): 4101.

Mirzaee F. (2014). Numerical solution of Fredholm fuzzy integral equations of the second kind using hybrid of block-pulse functions and Taylor series. *Ain Shams Engineering Journal* **5**(2): 631–636.

Mirzaee F., Paripour M. & Yari M.K. 2014. Application of triangular and delta basis functions to solve linear Fredholm fuzzy integral equation of the second kind. *Arabian Journal for Science and Engineering* **39**: 3969-3978.

Mirzaee F., Yari M.K. & Hoseini S.F. 2015. A computational method based on hybrid of Bernstein and block-pulse functions for solving linear fuzzy Fredholm integral equations system. *Journal of Taibah University for Science* **9**(2): 252–263.

Molabahrami A., Shidfar S. & Ghyasi A. 2011. An analytical method for solving linear Fredholm fuzzy integral equations of the second kind. *Computer and Mathematics with Applications* **61**(9): 2754-2761.

Muhiddin F. A., Sulaiman J. & Sunarto A. 2020. Numerical evaluation of quarter-sweep KSOR method to solve time-fractional parabolic equations. *International Journal of Engineering Trends and Technology*: 63-69.

Muthuvalu M.S. & Sulaiman J. 2011. Numerical solution of second kind linear Fredholm integral equations using QSGS iterative method with high-order Newton-Cotes quadrature schemes. *Malaysian Journal of Mathematical Sciences* **5**(1): 85-100.

Noeiaghdam S., Araghi M.A.F. & Abbasbandy S. 2020. Valid implementation of Sinc-collocation method to solve the fuzzy Fredholm integral equation. *Journal of Computational and Applied Mathematics* **370**: 1-19.

Nouriani H. & Ezzati R. 2017. Quadrature iterative method for numerical solution of two-dimensional linear fuzzy Fredholm integral equations. *Mathematical Sciences* **11**: 63–72.

Othman M. & Abdullah A.R. 2000. An efficient four points modified explicit group Poisson solver. *International Journal of Computer Mathematics* **76**(2): 203–217.

Rasheed M., Mohammedali, M.N., Sadiq, F.A., Sarhan, M.A. & Saidani, T. 2025. Application of innovative fuzzy integral techniques in solar cell systems. *Journal of Optics* **54:** 3490-3504.

Saleem N., Javed, K., Uddin, F., Ishtiaq, U., Khalil, A., Abdeljawad, T. & Alqudah, M. 2023. Unique solution of integral equations via intuitionistic extended fuzzy b-metric-like spaces. *Computer Modeling in Engineering & Sciences* **135**(1): 1-23.

Sunarto A., Agarwal P., Sulaiman J., Chew J.V.L. & Momani S. 2021a. Quarter-sweep preconditioned relaxation method, algorithm and efficiency analysis for fractional mathematical equation. *Fractal and Fractional* **5**(3): 1-14.

Sunarto A., Sulaiman J. & Lung J.C.V. 2021b. Numerical solution of the time-fractional diffusion equations via quarter-sweep preconditioned Gauss-Seidel method. *International Journal of Applied Mathematics* **34**(1): 111-125.

Suzuki A. & Negishi E. 2024. Fuzzy logic systems for healthcare applications. *Journal of Biomedical and Sustainable Healthcare Applications* **4**(1): 1-9.

School of Quantitative Sciences
College of Arts and Sciences
Universiti Utara Malaysia
06010 Sintok
Kedah, MALAYSIA
*E-mail: labiyana@uum.edu.my**

Faculty of Computer and Mathematical Sciences
Universiti Teknologi MARA
40450 Shah Alam
Selangor, MALAYSIA
E-mail: tasnemjaaffar@tmsk.uitm.edu.my, harliza@tmsk.uitm.edu.my

Faculty of Science and Natural Resources
Universiti Malaysia Sabah
88400 Kota Kinabalu
Sabah, MALAYSIA
E-mail: jumat@ums.edu.my

Received: 2 November 2025
Accepted: 9 February 2026

*Corresponding author