Mathematics and Statistics 10(1): 246-250, 2022 http://www.hrpub.org
DOI: 10.13189/ms.2022.100123

Newton-PKSOR with Quadrature Scheme in Solving Nonlinear Fredholm Integral Equations

Labiyana Hanif Ali[1,*], **Jumat Sulaiman**[1], **Azali Saudi**[2]

[1]Faculty of Science and Natural Resources, Universiti Malaysia Sabah, Kota Kinabalu, Sabah - 88400, MALAYSIA
[2]Faculty of Computing and Informatics, Universiti Malaysia Sabah, Kota Kinabalu, Sabah - 88400, MALAYSIA

Received September 21, 2021; Revised December 29, 2021; Accepted January 16, 2022

Cite This Paper in the following Citation Styles

***(a)**: [1] Labiyana Hanif Ali, Jumat Sulaiman, Azali Saudi, "Newton-PKSOR with Quadrature Scheme in Solving Nonlinear Fredholm Integral Equations," Mathematics and Statistics, Vol.10, No.1, pp. 246-250, 2022. DOI: 10.13189/ms.2022.100123*

***(b)**: Labiyana Hanif Ali, Jumat Sulaiman, Azali Saudi, (2022). Newton-PKSOR with Quadrature Scheme in Solving Nonlinear Fredholm Integral Equations. Mathematics and Statistics, 10(1), 246-250. DOI: 10.13189/ms.2022.100123*

Abstract In this study, we applied Newton method with a new version of KSOR, called PKSOR to form NPKSOR in solving nonlinear second kind Fredholm integral equations. A new version of KSOR is an update to the KSOR method with two relaxation parameters. The properties of KSOR helps in enlargement of the solution domain so the relaxation parameter can take the value $\omega^* \in \mathbb{R} - [-2, 0]$. With PKSOR, the relaxation parameter in KSOR ω^* is treated into two different relaxation paramaters as ω_1^* and ω_2^* which resulting lower number of iteration compared to the KSOR method. By combining the Newton method with PKSOR, we intend to from more efficient method to solve the nonlinear Fredholm integral equations. The discretization part of this study is done using first-order quadrature scheme to develop a nonlinear system. We formulate the solution of the nonlinear system using the given approach by reducing it to a linear system and then solving it using iterative methods to obtain an approximate solution. Furthermore, we compare the results of the proposed methods with NKSOR and NGS methods on three examples. Based on our findings, the NPKSOR method is more efficient than NKSOR and NGS methods. By implementing the NPKSOR method, we can boost the convergence rate of the iteration by considering two relaxation parameters, resulting in a lower number of iteration and computational time.

Keywords PKSOR, NPKSOR, Nonlinear Fredholm Integral Equations, Newton's Method, Quadrature Scheme

1 Introduction

In this study we consider the following nonlinear second kind Fredholm integral equations in Urysohn form [1]

$$y(x) = f(x) + \int_a^b k(x, t, y(t))dt, x \in [a, b]. \qquad (1)$$

where $k(x, t, y(t))$ is continuous on $[a, b]$, $f(x)$ is the given function and $y(x)$ is the unknown function to be determined. The nonlinearity of (1) is defined by function $y(t)dt$. Whenever the function is nonlinear, the equation is called nonlinear integral equations. These equations are one of the most useful integral equations and its applications in various research fields has been highlighted in many literatures, see [2,3]. A survey and study about the methods in solving (1) can be found in [4] and [5,6]. Although with various existing methods, we still can see the development of studies on this problem in many recent literatures such as in [7-11]. The studies on this topic has been conducted widely which covers various types of analytical and numerical methods to find more accurate and efficient methods in solving the problem.

Since analytical methods are hard to be develop, we choose to study numerical methods in solving (1). This study considers Newton method with a new version of KSOR, called NPKSOR to solve (1). At first, the discretization part of this study is done using first-order quadrature scheme to develop a nonlinear system. By converting the nonlinear system into a linear form with Newton method, we can solve it with iterative methods. SOR method is one of the most used iterative methods. It has been introduced in [12], which has become an important tool to accelerate the convergence rate of the iteration. The theory of the SOR method has been used widely and its applications in solving many mathematical and engineering problems is undeniable, see [13-17]. KSOR method is known as a new variant of SOR method [18]. The properties of KSOR helps in enlargement of the solution domain so the relaxation parameter can take the value $\omega^* \in \mathbb{R} - [-2, 0]$ compared to SOR which limited to $\omega^* \in \mathbb{R}(0, 2)$. Also, KSOR method can be used with more relaxation parameters, therefore we can find the development of iterative methods based on KSOR method in MKSOR, KAOR, and PKSOR methods, see [19-21].

Recently, a new version of KSOR, called PKSOR has been introduced in [21]. The new version of KSOR is shown to have better performance than the KSOR method. The idea of PSKOR is that the ω^* in KSOR is treated into ω_1^* and ω_2^*. Subsequently by finding the different coefficients, the convergence rate can be increased. The idea is quite similar to the MSOR and MKSOR which were proposed in [19]. In MSOR and MKSOR methods, the formulation of SOR and KSOR has been used in forward and backward formulation with ω^* is treated as two different relaxation parameters ω_1^* and ω_2^*, see [18,19]. Interestingly, PSKOR can have two relaxation parameters with fewer steps than the MSOR and MKSOR methods. On the other hand, as SOR and KSOR shares the same structure, the PKSOR also shares the same structure with the KSOR method with only a small modification on its relaxation parameter. By adding other relaxation parameters to the equations, the implementation of PKSOR is not difficult to be conducted. This gives the advantage to use PKSOR in saving the operational cost to get the solution for the problem more effectively.

2 Methodology

In this section, we discuss the discretization and the formulation of iterative methods in solving (1). First, (1) is discretized using first-order quadrature scheme by replacing the integration part of (1) with a sequence of finite dimensional nonlinear approximating problems. Following that, the finite dimensional problem is solved by some Newton iterative methods. The idea of Newton iterative method is that we solve the nonlinear system by converting it to a linear system before we apply iterative methods to find the approximate solution.

2.1 Discretization part of nonlinear Fredholm integral equations

We begin the discretization by applying the first-order quadrature scheme into (1) to obtain [13]

$$y(x) - \sum_{j=0}^{n} k(x, t, y(t))dt = f(x). \tag{2}$$

Then, by expending the sequence of (2), we can form the following nonlinear approximation equations

$$\begin{aligned} y_i - \frac{1}{2}hk(x, t_0, y_0)dt - hk(x, t_1, y_1)dt - hk(x, t_2, y_2)dt - \\ \ldots - \frac{1}{2}hk(x, t_n, y_n)dt = f_i, \end{aligned} \tag{3}$$

for $i = 0, 1, \ldots, n$. Now, we consider the nonlinear function of (3) which yileds into

$$\begin{aligned} G_i(y_0, y_1, y_2, \ldots, y_n) = y_i - \frac{1}{2}hk(x, t_0, y_0)dt - \\ hk(x, t_1, y_1)dt - \ldots - \frac{1}{2}hk(x, t_n, y_n)dt - f_i. \end{aligned} \tag{4}$$

We have system of nonlinear equations based on (6) in following form

$$G_i(y_0, y_1, y_2, \ldots, y_n) = 0. \tag{5}$$

By implementing the Newton method, we have [22]

$$J(\underline{y}^{(k)})\Delta y^{(k)} = -G(\underline{y}^{(k)}), \tag{6}$$

where

$$J(\underline{y}^{(k)}) = \begin{bmatrix} \frac{df_0}{dy_0} & \frac{df_0}{dy_1} & \frac{df_0}{dy_2} & \cdots & \frac{df_0}{dy_n} \\ \frac{df_1}{dy_0} & \frac{df_1}{dy_1} & \frac{df_1}{dy_2} & \cdots & \frac{df_1}{dy_n} \\ \frac{df_2}{dy_0} & \frac{df_2}{dy_1} & \frac{df_2}{dy_2} & \cdots & \frac{df_2}{dy_n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{df_n}{dy_0} & \frac{df_n}{dy_1} & \frac{df_n}{dy_2} & \cdots & \frac{df_n}{dy_n} \end{bmatrix}_{(n+1)\times(n+1)}, \tag{7}$$

$$\Delta y^{(k)} = \begin{bmatrix} \Delta y_0 & \Delta y_1 & \Delta y_2 & \cdots & \Delta y_n \end{bmatrix}^T, \tag{8}$$

and solution vector for linear system in Eq. (8) can be determined by

$$y_i^{(k+1)} = y_i^{(k)} + \Delta y_i. \tag{9}$$

2.2 Formulation of Newton's iterative method

Following the discretization scheme, we now discuss the formulation of iterative methods to solve the linear system in (6). Suppose the linear system in (6) is represented as follows

$$\sum_{j=1}^{m} a_{ij}x_j = b_i, i = 0, 1, ..., m. \tag{10}$$

The formulation of NGS is defined by [18]

$$x_i^{[n+1]} = x_i^{(n)} + \frac{1}{a_{ii}}(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{[n+1]} - \sum_{j=i+1}^{m} a_{ij}x_j^{[n]}), \tag{11}$$

the formulation of NKSOR is defined by

$$x_i^{[n+1]} = \frac{1}{(1+\omega^*)}(x_i^{(n)} + \frac{\omega^*}{a_{ii}}(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{[n+1]} - \sum_{j=i+1}^{m} a_{ij}x_j^{[n]})), \tag{12}$$

with $\omega^* \in \mathbb{R} - [-2, 0]$ and where $n = 0, 1, 2, ..$ represents the iteration.

Based on the formulation of NKSOR, the NPKSOR can be defined as follows [21]

$$x_i^{[n+1]} = \frac{1}{(1+\omega_2^*)}(x_i^{(n)} + \frac{\omega_1^*}{a_{ii}}(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{[n+1]} - \sum_{j=i+1}^{m} a_{ij}x_j^{[n]})), \tag{13}$$

where $\omega_1^* \in \mathbb{R}$, $\omega_2^* \in \mathbb{R} - [-2, 0]$ and where $n = 0, 1, 2, ...$

Table 1. Results of NGS, NKSOR, and NPKSOR for Example 1.

	Methods	Grid Size				
		256	512	1024	2048	4096
k	NGS	182	183	183	183	184
	NKSOR	119	119	119	120	120
		$(\omega^* = -4.0731)$	$(\omega^* = -4.0685)$	$(\omega^* = -4.0662)$	$(\omega^* = -4.0525)$	$(\omega^* = -4.0525)$
	NPKSOR	115	115	114	109	107
		$(\omega_1^* = -4.0769)$	$(\omega_1^* = -4.0723)$	$(\omega_1^* = -4.0700)$	$(\omega_1^* = -4.0563)$	$(\omega_1^* = -4.0563)$
		$(\omega_2^* = -4.0731)$	$(\omega_2^* = -4.0685)$	$(\omega_2^* = -4.0662)$	$(\omega_2^* = -4.0525)$	$(\omega_2^* = -4.0525)$
$Time$	NGS	0.26	0.94	3.76	14.97	60.23
	NKSOR	0.19	0.66	2.48	9.90	35.61
	NPKSOR	0.17	0.61	2.37	9.00	35.34
$Error$	NGS	$1.27164E-05$	$3.17884E-06$	$7.94608E-07$	$1.98558E-07$	$4.95456E-08$
	NKSOR	$1.27165E-05$	$3.17896E-06$	$7.94723E-07$	$1.98674E-07$	$4.96624E-08$
	NPKSOR	$1.27165E-05$	$3.17897E-06$	$7.94747E-07$	$1.98698E-07$	$4.96621E-08$

Table 2. Results of NGS, NKSOR, and NPKSOR for Example 2.

	Methods	Grid Size				
		256	512	1024	2048	4096
k	NGS	117	118	118	118	118
	NKSOR	87	87	88	88	88
		$(\omega^* = -4.9355)$	$(\omega^* = -4.9268)$	$(\omega^* = -4.9104)$	$(\omega^* = -4.9077)$	$(\omega^* = -4.9055)$
	NPKSOR	85	85	85	85	85
		$(\omega_1^* = -4.9366)$	$(\omega_1^* = -4.9279)$	$(\omega_1^* = -4.9115)$	$(\omega_1^* = -4.9088)$	$(\omega_1^* = -4.9066)$
		$(\omega_2^* = -4.9355)$	$(\omega_2^* = -4.9268)$	$(\omega_2^* = -4.9104)$	$(\omega_2^* = -4.9077)$	$(\omega_2^* = -4.9055)$
$Time$	NGS	0.19	0.71	2.86	10.79	43.20
	NKSOR	0.17	0.54	2.09	8.22	33.17
	NPKSOR	0.14	0.52	2.04	8.03	31.93
$Error$	NGS	$1.01732E-05$	$2.54313E-06$	$6.35738E-07$	$1.58898E-07$	$3.96883E-08$
	NKSOR	$1.01732E-05$	$2.54317E-06$	$6.35784E-07$	$1.58944E-07$	$3.97342E-08$
	NPKSOR	$1.01732E-05$	$2.54317E-06$	$6.35783E-07$	$1.58943E-07$	$3.97335E-08$

Table 3. Results of NGS, NKSOR, and NPKSOR for Example 3.

	Methods	Grid Size				
		256	512	1024	2048	4096
k	NGS	1068	1096	1110	1118	1122
	NKSOR	635	641	644	645	646
		$(\omega^* = 1.1518)$	$(\omega^* = 0.9557)$	$(\omega^* = 0.9209)$	$(\omega^* = 0.9175)$	$(\omega^* = 0.9182)$
	NPKSOR	459	492	497	497	498
		$(\omega_1^* = 0.8812)$	$(\omega_1^* = 0.7433)$	$(\omega_1^* = 0.7511)$	$(\omega_1^* = 0.7481)$	$(\omega_1^* = 0.7491)$
		$(\omega_2^* = 1.1518)$	$(\omega_2^* = 0.9557)$	$(\omega_2^* = 0.9209)$	$(\omega_2^* = 0.9175)$	$(\omega_2^* = 0.9182)$
$Time$	NGS	1.63	5.88	23.59	93.83	376.50
	NKSOR	0.93	3.44	13.64	54.61	218.65
	NPKSOR	0.74	2.70	10.68	42.49	172.08
$Error$	NGS	$7.03898E-07$	$1.75994E-07$	$4.40177E-08$	$1.10236E-08$	$2.77505E-09$
	NKSOR	$7.03908E-07$	$1.76078E-07$	$4.40987E-08$	$1.11044E-08$	$2.85597E-09$
	NPKSOR	$7.03927E-07$	$1.76056E-07$	$4.40509E-08$	$1.10567E-08$	$2.80573E-09$

Suppose the linear system represented in the matrix form $Ax = b$, so the matriks form of NPSKOR is

$$[(1+\omega_2^*)D - \omega_1^* L]x^{[n+1]} = (D+\omega_1^* U)x^{[n]} + \omega_1^* b, \quad (14)$$

where $A = D - L - U$, in which $A \leftarrow A - dD$ where $\frac{\omega_2^* - \omega_1^*}{\omega_2^*}$. D represents diagonal matrix, L and U are stickly lower and upper triangular matrix of A, respectively. The algorithm of NPKSOR method to solve the second kind Fredholm integral equations can be presented as follows

Algorithm 1: Formulation of NPKSOR i. Set the initial value $\Delta y^{(k)} = 0$, $k = 0$, and $\xi = 10^{-10}$.
ii. Set $q = 0$ and compute matrix $J(\underline{y}^{(k)})$ and $G(\underline{y}^{(k)})$.
iii. Compute the current value, $y_i^{(k+1)}$
a. For $i = 0, 1, \ldots, n$ solve Eq. (6) using Eq. (14).
b. Conduct the convergence test, $\left|\Delta y_i^{(k+1)} - \Delta y_i^{(k)}\right| \leq \xi$. If satisfied, continue to step iv, otherwise repeat step iii(a).
iv. Conduct the convergence test, $\left|G(\underline{y}^{(k+1)}) - G(\underline{y}^{(k)})\right| \leq \xi$. If satisfied, display the approximate solution, and otherwise repeat step iii.
v. Stop.

3 Numerical Examples and Discussion

Based on the methodology of this study, we test the NGS, NKSOR, and NPKSOR methods using the Borland C++ tool on three examples. At first, we run the data to find the relaxation parameters which gives minimum iteration of the methods. The idea of finding the optimum relaxation parameters NPKSOR is to first find the optimum relaxation parameter of the NKSOR method. By running the program on $\omega^* \in \mathbb{R}\backslash[-2, 0]$ with zero decimal places, we find the relaxation parameter which gives the lowest iteration number. Then, we go thoroughly to find the optimum relaxation parameter with the lowest number of iteration by increasing the number of decimal places until 4 decimal places. Following that, we set the ω_*^2 to be as in ω^* of NKSOR and we find ω_*^1 using the same technique. Finally, we run the data again to find the number of iteration, computational time, and maximum absolute error. On the other hand, we start the initial approximation with $\Delta y^{(k)} = 0$, then we run the iterative methods to get the approximate solutions of the problems to the nearest exact solution. The data is collected and we compare the results of NGS, NKSOR, and NPKSOR methods in Table 1 to Table 3.

Example 1. Consider the following equation [23]

$$y(x) = 1 - \frac{5}{12}x + \int_0^1 xt[y(t)]^2 dt,$$

where exact solution is y $y(x) = 1 + \frac{1}{3}t$.

Example 2. Consider the following equation [24]

$$y(x) = cos(x) + \frac{5}{2} - x + \frac{1}{2}cos(1)sin(1) + 2sin(1) + \int_0^1 (x - y(t)^2)dt,$$

where exact solution is $y(x) = cos(x) + 1$.

Example 3. Consider the following equation [25]

$$y(x) = 1 - \frac{1}{3}x + xt^2 y^3(t)dt,$$

where exact solution is $y(x) = 1$.

Based on the data obtained, we found that the NPKSOR to have a lowest number of iteration and computational time compared to NGS and NKSOR for all the tested examples. The application of NPKSOR is proven to be more effective in increasing the convergence rate of the iteration. The accuracy of the data obtained becomes more accurate as the grid size increases.

4 Conclusions

In this study, we applied a new iterative method called NPKSOR with first-order quadrature scheme to solve nonlinear second kind Fredholm integral equations. Based on our results, we found that NPKSOR to be more efficient than NKSOR and NGS. By considering two relaxation parameters, we can increase the convergence rate of the iteration with NPKSOR method where the lower number of iteration and computation time can be obtained. As a new version of KSOR shares the same structure to the KSOR method, the implementation of NPKSOR is not difficult to be done and only requires low operational cost to implement the method. By considering the feature of a new version of KSOR, the similar technique can be applied in solving many types of equations. Also, the further research and discussion can be made to see the significant contribution of a new version of KSOR to overcome the shortcomings of the existing iterative methods. Moreover, further studies of solving nonlinear second kind integral equations can be considered by implementing the half- and quarter-sweep technique in reducing the operational cost of the iteration process, see [26-31].

Acknowledgements

The authors acknowledge the funding of this research by Universiti Malaysia Sabah under research grant code: GUG0485-1/2020

REFERENCES

[1] I. I. Maglevanny, T. I. Karyakina, Numerical solution of Urysohn type nonlinear second kind integral equations by successive quadratures using embedded Dormand and Prince scheme 5(4), Computer Research and Modeling, Vol. 12, 275-300, 2020

[2] J.A. Ezquerro, M.A. Hernández-Verón, On the Existence of Solutions of Nonlinear Fredholm Integral Equations from Kantorovich's Technique, Algorithms, Vol. 10, 2-11, 2020.

[3] M. Kaur, I. Singh, Hermite Wavelets Collocation Method for Solving Nonlinear Fredholm Integral Equations, SAMRIDDHI, Vol. 12, 82-88, 2020.

[4] K.E. Atkinson, A survey of numerical methods for solving nonlinear integral equations, Journal of Integral Equations, Vol. 4, 15-46, 1992.

[5] K.E. Atkinson, The numerical solution of integral equations of the second kind, Cambridge University Press: Cambridge, UK (1997).

[6] L.M. Delves, J.L. Mohamed, Computational methods for integral equations, Cambridge University Press: Cambridge, UK (1985).

[7] S. Amiri, M. Hajipour, D. Baleanu, On accurate solution of the Fredholm integral equations of the second kind, Applied Numerical Mathematics, Vol. 150, 478-490, 2020.

[8] F. Awawdeh, L. Smail, Convergence analysis of a highly accurate Nyström scheme for Fredholm integral equations, Applied Numerical Mathematics, Vol.152, 231-242, 2020.

[9] K. Nedaiasl, Approximation of weakly sigular integral eqautions by sinc projection methods, Electronic Transactions on Numerical Analysis, Vol. 52, 416-430, 2020.

[10] M. Ramadan, H.S. Osheba, A new hybrid orthonormal Bernstein and improved block-pulse functions method for solving mathematical physics and engineering problems, Electronic Transactions on Numerical Analysis, Vol. 59, 3643-3652, 2020.

[11] K. Parand, A.A. Aghaei, M. Jani, A. Ghodsi, A new approach to the numerical solution of Fredholm integral equations using least squares-support vector regression, Mathematics and Computers in Simulation, Vol. 180, 114-128, 2021.

[12] D. M. Young, Iterative Solution of Large Linear System, Academic Press, New York and London, 1971.

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

[14] N.F.A. Zainal, J. Sulaiman, A. Saudi, N.A.M. Ali, Semi-approximate Solution for Burgers' Equation Using SOR Iteration, Lecture Notes in Electrical Engineering, Vol. 724, 474-486, 2021.

[15] N.S. Mohamad, J. Sulaiman, A. Saudi, N.F.A. Zainal, First Order Piecewise Collocation Solution of Fredholm Integral Equation Second Type Using SOR Iteration, Lecture Notes in Electrical Engineering, Vol. 724, 395-406, 2021.

[16] N.A.M. Ali, J. Sulaiman, A. Saudi, N.S Mohamad, The Similarity Finite Difference Solutions for Two-Dimensional Parabolic Partial Differential Equations via SOR Iteration, Lecture Notes in Electrical Engineering, Vol. 724, 515-526, 2021.

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

[18] I.K. Youssef, On the Successive Overrelaxation Method, Journal of Mathematics and Statistics, Vol. 8, 176-184, 2012.

[19] I.K. Youssef, A.A. Taha, On the modified successive overrelaxation method, Applied Mathematics and Computation, Vol. 219, 4601-4613, 2013.

[20] I.K. Youssef, M.M. Farid, On the Accelerated Overrelaxation Method, Pure and Applied Mathematics Journal, Vol. 219, 26-31, 2015.

[21] R. Constantinescu, R. C. Poenaru, F. Pop, P. G. Popescu, A new version of KSOR method with lower number of iterations and lower spectral radius, Soft Computing, 2019.

[22] J. Sulaiman, M.K. Hasan, Newton-EGMSOR Methods for Solution of Second Order Two-Point Nonlinear Boundary Value Problems, Journal of Mathematics and System Science, Vol. 2, 185-190, 2012.

[23] K. Maleknejad, K. Nedaiasl, Application of Sinc-collocation method for solving a class of nonlinear Fredholm integral equations, Computers and Mathematics with Applications, Vol. 62, 3292-3303, 2011.

[24] M.M. Mustafa, K. A. AL-Zubaidy, Use of Bernstein Polynomial in Numerical Solution of Nonlinear Fred Holm Integral Equation, Eng. and Tech. Journal, Vol. 29, 110-115, 2011.

[25] A.M. Wazwaz, A first course in integral equations, World Scientific Publishing Co. Pte. Ltd., Singapore (2015).

[26] M.N. Suardi, J. Sulaiman, On redlich-kister finite difference solution of two-point boundary value problems using half-sweep kaudd successive over relaxation iteration, International Journal of Engineering Trends and Technology, Vol. 69, 77-82 2021.

[27] L.H. Ali, J. Sulaiman, S.R.M. Hashim, Numerical solution of fuzzy fredholm integral equations of second kind using half-sweep gauss-seidel iteration, Journal of Engineering Science and Technology, Vol.15, 3303-3313, 2020.

[28] N.F.A. Zainal, J. Sulaiman, M.U. Alibubin, Application of half-sweep sor iteration with nonlocal arithmetic discretization scheme for solving Burger's equation, ARPN Journal of Engineering and Applied Sciences, Vol. 14, 616-621, 2019.

[29] J. V. L. Chew, J. Sulaiman, A. Sunarto. Solving One-Dimensional Porous Medium Equation Using Unconditionally Stable Half-Sweep Finite Difference and SOR Method, Mathematics and Statistics, Vol.9, 166-171, 2021.

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

[31] J.H. Eng, A. Saudi, J. Sulaiman. Implementation of quarter-sweep approach in poisson image blending problem, Lecture Notes in Electrical Engineering, Vol.481, 127-136, 2019.