Menemui Matematik (Discovering Mathematics)
Vol. 33, No. 1: 49– 57 (2011)

Numerical Solutions of First Kind Linear Fredholm Integral Equations Using Quarter-Sweep Successive Over-Relaxation (QSSOR) Iterative Method

[1] Muthuvalu, M.S. and [2,] * Sulaiman, J.
[1,2] School of Science and Technology, Universiti Malaysia Sabah,
Locked Bag 2073, 88999 Kota Kinabalu, Sabah, Malaysia
[1] sundaram_at2@yahoo.com, [2] jumat@ums.edu.my

ABSTRACT

In this paper, an experimental study is conducted to show the efficiency of the Quarter-Sweep Successive Over-Relaxation (QSSOR) iterative method by using the quadrature approximation equations to obtain numerical solutions of the first kind linear Fredholm integral equations. Furthermore, the derivation and implementation of the QSSOR method in solving first kind linear Fredholm integral equations are also presented. Numerical examples and comparisons with other existing methods are given to illustrate the effectiveness of the proposed method.

Keywords: **First kind linear Fredholm equations, Quadrature, Quarter-sweep iteration, Successive Over-Relaxation**

INTRODUCTION

Presently, the theory and application of integral equations is an important subject within applied mathematics. Integral equations are used as mathematical models for many and varied physical circumstances and also occur as reformulations of other mathematical models. Particularly, linear Fredholm integral equations of the first kind appear in the mathematical formulation of various and important inverse problems such as seismology, gravity surveying, computerized tomography and image deblurring (Băutu *et al*., 2005).

The above-mentioned inverse problems, as well as others, can be formulated as a first kind linear integral equations, which has the generic form as follows

$$\int_{\Gamma} K(x,t)y(t)dt = f(x), \quad \Gamma = [a,b] \tag{1}$$

where the kernel function $K \in L^2(\Gamma \times \Gamma)$ and the function $f \in L(\Gamma)$ are given, and $y \in L(\Gamma)$ is the unknown function to be determined. $K(x, t)$ is called Fredholm kernel if the kernel in Eq. (1) is continuous on the square $S = \{a \le x \le b, a \le t \le b\}$ or at least square integrable on this square. Then, Eq. (1) with constant integration limits and Fredholm kernel are called first kind linear Fredholm integral equations (Polyanin & Manzhirov, 1998). Meanwhile, Eq. (1) also can be rewritten in the operator form as follows

$$\kappa: S \to T\ \kappa(y(t)) = \int_a^b K(x,t)y(t)dt. \tag{2}$$

Definition (Maleknejad *et al*., 2006)

Let $\kappa : S \to T$ be an operator from normed space S into a normed space T, the equation $\kappa y = f$ is called well-posed if κ is onto, one to one and $\kappa^{-1} : T \to S$ is continuous. Otherwise the equation is called ill-posed.

In many application areas, numerical approaches were used widely to solve Fredholm integral equations. To solve Eq. (2) numerically, we either seek to determine an approximation solution in a chosen finite dimensional space by using projection method (Hsiao, 1980; Shang & Han, 2007; Maleknejad *et al*., 2006; Oladejo *et al*., 2008) or the quadrature method (Boland, 1972; Muthuvalu & Sulaiman, 2008; 2009). Such discretizations of integral equations lead to dense linear systems and can be prohibitively expensive to solve as n , the order of the linear systems increases. Thus, iterative methods are the natural options for efficient solutions of the linear system.

Consequently, the concept of the half-sweep iteration method has been inspired by Abdullah (1991) via the Explicit Decoupled Group (EDG) method to solve two-dimensional Poisson equations. Half-sweep iteration is also known as the complexity reduction approach (Hasan *et al*., 2007). Following to that, applications of the half-sweep iteration iterative methods have been reviewed in Yousif and Evans (1995), Abdullah and Ali (1996), Othman *et al*. (2000), Sulaiman *et al*. (2004; 2007; 2008) and Abdullah *et al.* (2006). In 2000, Othman and Abdullah extended this concept by introducing quarter-sweep iterative method via the Modified Explicit Group (MEG) iterative method to solve two-dimensional Poisson equations. Further studies to verify the effectiveness of the quarter-sweep iterative methods have been carried out by Othman and Abdullah (2001), Hasan *et al*. (2005), Sulaiman *et al*. (2004), Hasan *et al*. (2008) and Sulaiman *et al*. (2008). The basic idea of the half- and quarter-sweep iterative methods is to reduce the computational complexities during iteration process. Since the implementation of the half- and quarter-sweep iterations will only consider nearly half and quarter of all interior node points in a solution domain respectively. In this paper, we examined the applications of the half- and quarter-sweep iteration concepts with Successive Over-Relaxation (SOR) iterative method by using approximation equation based on quadrature scheme for solving problem (1). The standard SOR iterative method is also called as the Full-Sweep Successive Over-Relaxation (FSSOR) method. Meanwhile, combinations of the SOR method with half- and quarter-sweep iterations are called as Half-Sweep Successive Over-Relaxation (HSSOR) and Quarter-Sweep Successive Over-Relaxation (QSSOR) methods respectively.

The remainder of this paper is organized in following way. In next section, the formulation of the full-, half- and quarter-sweep quadrature approximation equations will be elaborated. The latter section of this paper will discuss the formulations of the FSSOR, HSSOR and QSSOR iterative methods in solving linear systems generated from discretization of the Eq. (1) and then some numerical results will be shown to assert the effectiveness of the proposed method. Besides that, analysis on computational complexity is also given and the concluding remarks are given in final section.

FULL-, HALF- AND QUARTER-SWEEP QUADRATURE APPROXIMATION EQUATIONS

As afore-mentioned, a discretization scheme based on method of quadrature was used to construct approximation equations for problem (1) by replacing the integral to finite sums. Generally, quadrature method can be defined as follows

$$\int_a^b y(t)dt = \sum_{j=0}^{n} A_j y(t_j) + \varepsilon_n(y) \tag{3}$$

where t_j (j = 0, 1, 2, ... n) is the abscissas of the partition points of the integration interval [a, b], A_j (j=0, 1, 2, ... n), is numerical coefficients that do not depend on the function y(t) and $\varepsilon_n(y)$ is the truncation error of Eq. (3). Meanwhile, Fig. 1 shows the finite grid networks in order to form the full- and quarter-sweep quadrature approximation equations.

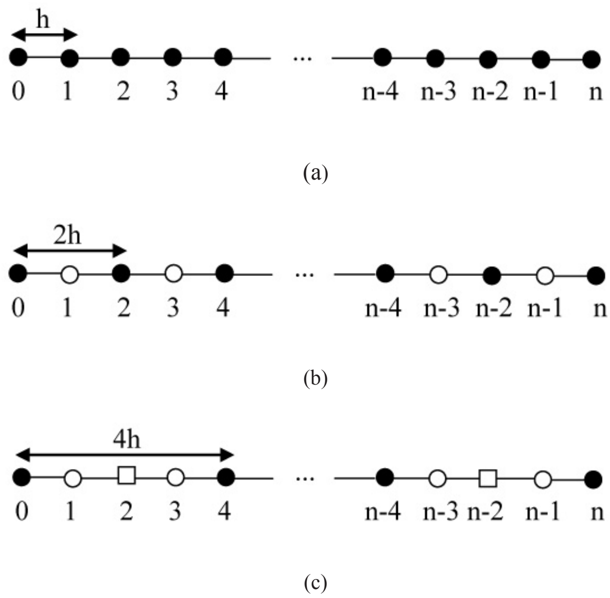

Figure 1 a), b) and c) show distribution of uniform node points for the full-, half- and quarter-sweep cases respectively.

Based on Fig. 1, the full-, half- and quarter-sweep iterative methods will compute approximate values onto node points of type ● only until the convergence criterion is reached. Then, other approximate solutions at remaining points (points of the different type) can be computed using the direct method (Abdullah, 1991; Othman & Abdullah, 2000).

By applying Eq. (3) into Eq. (1) and neglecting the error, $\varepsilon_n(y)$, a system of linear equations can be formed for approximation values of $y(t)$. The following linear system generated using quadrature method can be easily shown in matrix form as follows

$$My = f \tag{4}$$

where

$$M = \begin{bmatrix} A_0K_{0,0} & A_pK_{0,p} & A_{2p}K_{0,2p} & \cdots & A_nK_{0,n} \\ A_0K_{p,0} & A_pK_{p,p} & A_{2p}K_{p,2p} & \cdots & A_nK_{p,n} \\ A_0K_{2p,0} & A_pK_{2p,p} & A_{2p}K_{2p,2p} & \cdots & A_nK_{2p,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_0K_{n,0} & A_pK_{n,p} & A_{2p}K_{n,2p} & \cdots & A_nK_{n,n} \end{bmatrix}_{\left(\left(\frac{n}{p}\right)+1\right)x\left(\left(\frac{n}{p}\right)+1\right)},$$

$$\underset{\sim}{y} = [y_0 \quad y_p \quad y_{2p} \quad \cdots \quad y_{n-2p} \quad y_{n-p} \quad y_n]^T,$$

and

$$\underset{\sim}{f} = [f_0 \quad f_p \quad f_{2p} \quad \cdots \quad f_{n-2p} \quad f_{n-p} \quad f_n]^T.$$

In order to facilitate the formulation of the full-, half- and quarter-sweep quadrature approximation equations for problem (1), further discussion will be restricted onto repeated trapezoidal (RT) scheme, which is based on linear interpolation formula with equally spaced data. Based on RT scheme, numerical coefficient will satisfy the following relation

$$A_j = \begin{cases} \frac{1}{2}ph, & j = 0, n \\ ph, & otherwise \end{cases} \tag{5}$$

where the constant step size, h is defined as follows

$$h = \frac{b-a}{n} \tag{6}$$

and n is the number of subintervals in the interval $[a,b]$. Meanwhile, the value of p, which corresponds to 1, 2 and 4, represents the full-, half- and quarter-sweep cases respectively.

FORMULATION OF THE SUCCESSIVE OVER-RELAXATION METHODS

As mentioned in Section 1, FSSOR, HSSOR and QSSOR iterative methods will be applied to solve linear system generated from the discretization of the problem (1), as shown in Eq. (4). Let matrix M be decomposed into

$$M=D-L-U \tag{7}$$

where D, $-L$ and U are diagonal, strictly lower triangular and strictly upper triangular matrices respectively. Thus, the general scheme for FSSOR, HSSOR and QSSOR iterative method can be written as

$$D\underset{\sim}{y}^{(k+1)} = \omega L\underset{\sim}{y}^{(k+1)} + \omega U\underset{\sim}{y}^{(k)} + \omega\underset{\sim}{f} + (1-\omega)D\underset{\sim}{y}^{(k)} \tag{8}$$

where ω is a weighted parameter.

Actually, the iterative methods attempt to find a solution to the system of linear equations by repeatedly solving the linear system using approximations to the vector y. Iterations for iterative methods continue until the solution is within a predetermined acceptable bound on the error. By determining values of matrices D, $-L$ and U as stated in Eq. (7), the general algorithm for FSSOR, HSSOR and QSSOR iterative methods to solve problem (1) would be generally described in Algorithm 1.

Algorithm 1: FSSOR, HSSOR and QSSOR iterative methods

For $i = 0, p, 2p, \cdots, n-2p, n-p, n$ and $j = 0, p, 2p, \cdots, n-2p, n-p, n$
Calculate

$$y_i^{(k+1)} \leftarrow \begin{cases} (1-\omega)y_i^{(k)} + \left(\omega \sum_{j=p}^{n} A_j K_{i,j} y_j^{(k)} + \omega f_i\right) \Big/ A_i K_{i,i}, & i = 0 \\ (1-\omega)y_i^{(k)} + \left(\omega \sum_{j=0}^{n-p} A_j K_{i,j} y_j^{(k+1)} + \omega f_i\right) \Big/ A_i K_{i,i}, & i = n \\ (1-\omega)y_i^{(k)} + \left(\omega \sum_{j=0}^{i-p} A_j K_{i,j} y_j^{(k+1)} + \omega \sum_{j=i+p}^{n} A_j K_{i,j} y_j^{(k)} + \omega f_i\right) \Big/ A_i K_{i,i}, & \text{others} \end{cases}$$

NUMERICAL EXPERIMENTS

In order to compare the performances of the iterative methods described in the previous section, several experiments were carried out on the following two Fredholm integral equations problems. In this paper, we will only consider well-posed equations and the case where a=0 and b=1.

Example 1 (Rashed, 2003)

$$\int_0^1 K(x,t)y(t)dt = \frac{1}{6}(x^3 - x),\ 0 < x < 1 \tag{9}$$

with kernel

$$K(x,t) = \begin{cases} t(x-1), & t < x \\ x(t-1), & x \le t \end{cases}.$$

The exact solution of the problem is

$$y(x) = x.$$

Example 2 (Rashed, 2003)

$$\int_0^1 K(x,t)y(t)dt = e^x + (1-e)x - 1, 0 < x < 1, \tag{10}$$

with kernel

$$K(x,t) = \begin{cases} t(x-1), & t < x \\ x(t-1), & x \le t \end{cases}.$$

The exact solution of the problem is

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

There are there parameters considered in numerical comparison such as number of iterations, execution time and maximum absolute error. As comparisons, the Full-Sweep Gauss-Seidel (FSGS) method acts as the control of comparison of numerical results. Throughout the simulations, the convergence test considered the tolerance error of $\varepsilon = 10^{-10}$. Meanwhile, the experimental values of ω were obtained by running the program for different values of ω and choosing the one(s) that gave the minimum number of iterations. The simulations were carried out on several mesh sizes, 511, 1023, 2047, 4095 and 8191.

Results of numerical simulations, which were obtained from implementations of the FSGS, FSSOR, HSSOR and QSSOR iterative methods for Examples 1 and 2, have been recorded in Tables 1 and 2 respectively. Meanwhile, reduction percentage of the number of iterations and execution time for the FSSOR, HSSOR and QSSOR methods compared with FSGS method have been summarized in Table 3.

Table 1 Comparison of a number of iterations, execution time and maximum absolute error for the iterative methods at optimum value of ω (Example 1)

Methods	Number of iterations				
	Mesh size				
	511	**1023**	**2047**	**4095**	**8191**
FSGS	381	461	550	646	753
FSSOR	273	312	347	371	450
HSSOR	252	273	312	347	371
QSSOR	226	252	273	312	347

Methods	Execution time (seconds)				
	Mesh size				
	511	**1023**	**2047**	**4095**	**8191**
FSGS	6.36	30.70	143.14	665.59	3177.57
FSSOR	4.57	20.15	64.86	258.37	1273.59
HSSOR	1.66	5.02	28.87	70.33	343.71
QSSOR	0.56	1.06	3.96	15.08	66.25

Methods	Maximum absolute error				
	Mesh size				
	511	**1023**	**2047**	**4095**	**8191**
FSGS	6.8225 E-10	8.3429 E-10	8.4449 E-10	9.7143 E-10	9.7966 E-10
FSSOR	6.4063 E-10	6.9672 E-10	7.3476 E-10	7.9103 E-10	8.3959 E-10
HSSOR	6.3826 E-10	6.4063 E-10	6.9672 E-10	7.3476 E-10	7.9103 E-10
QSSOR	6.3849 E-10	6.3826 E-10	6.4063 E-10	6.9672 E-10	7.3476 E-10

Table 2 Comparison of a number of iterations, execution time and maximum absolute error for the iterative methods at optimum value of ω (Example 2)

Methods	Number of iterations				
	Mesh size				
	511	**1023**	**2047**	**4095**	**8191**
FSGS	394	479	568	667	728
FSSOR	284	325	361	386	469
HSSOR	243	284	325	361	386
QSSOR	202	243	284	325	361

Methods	Execution time (seconds)				
	Mesh size				
	511	**1023**	**2047**	**4095**	**8191**
FSGS	4.77	20.36	91.35	423.66	2034.36
FSSOR	4.43	17.37	67.21	270.31	1159.89
HSSOR	1.58	5.45	23.66	78.11	329.60
QSSOR	0.29	1.13	4.82	21.17	89.45

Methods	Maximum absolute error				
	Mesh size				
	511	**1023**	**2047**	**4095**	**8191**
FSGS	8.6244 E-7	2.1571 E-7	5.5889 E-8	2.5713 E-8	4.2551 E-8
FSSOR	8.6244 E-7	2.1571 E-7	5.5217 E-8	2.2807 E-8	3.7588 E-8
HSSOR	8.5907 E-6	2.1540 E-6	5.3914 E-7	1.3544 E-7	3.7003 E-8
QSSOR	3.4162 E-5	8.5907 E-6	2.1540 E-6	5.3914 E-7	1.3544 E-7

Table 3 Reduction percentage of the number of iterations and execution time for the FSSOR, HSSOR and QSSOR methods compared with FSGS method

Methods	**Example 1**	
	Number of iterations	**Execution time**
FSSOR	28.34 - 42.57%	28.14 - 61.19%
HSSOR	33.85 - 50.74%	73.89 - 89.44%
QSSOR	40.68 - 53.92%	91.19 - 97.92%
Methods	**Example 2**	
	Number of iterations	**Execution time**
FSSOR	27.91 - 42.13%	7.12 - 42.99%
HSSOR	38.32 - 46.98%	66.87 - 83.80%
QSSOR	48.73 - 51.28%	93.92 - 95.61%

COMPUTATIONAL COMPLEXITY ANALYSIS

In order to measure the computational complexity of the FSSOR, HSSOR and QSSOR iterative methods, an estimation amount of the computational work required for iterative methods have been conducted. The computational work is estimated by considering the arithmetic operations performed per iteration. Based on Algorithm 1, it can be observed that there are additions/subtractions (ADD/SUB) and multiplication/divisions (MUL/DIV) in computing a value for each node point in the solution domain for first kind linear Fredholm integral equations. From the order of the coefficient matrix, in Eq. (4), the total number of arithmetic operations per iteration for the FSSOR, HSSOR and QSSOR iterative methods in solving problem (1) has been summarized in Table 4.

Table 4 Total number of arithmetic operations per iteration for FSSOR, HSSOR and QSSOR methods

Methods	**Arithmetic Operation**	
	ADD/SUB	**MUL/DIV**
FSSOR	n^2+3n+2	$2n^2+7n+5$
HSSOR	$\frac{n^2}{4}+\frac{3n}{2}+2$	$\frac{n^2}{2}+\frac{7n}{2}+5$
QSSOR	$\frac{n^2}{16}+\frac{3n}{4}+2$	$\frac{n^2}{8}+\frac{7n}{4}+5$

CONCLUSIONS

In Section 2, it has shown that the formulation of full-, half- and quarter-sweep quadrature approximation equations based on RT scheme can easily generate a system of linear equations. Through numerical solutions obtained in Tables 1 and 2, it clearly shows that half- and quarter-sweep iteration concept reduces number of iterations and computational time of the iterative method (refer Table 3). Meanwhile, the accuracy of all the iterative methods is in good agreement. It can be summarized that the QSSOR method is the most superior among the iterative methods in the sense of number of iterations and execution time as the mesh sizes getting larger. This is mainly because of computational complexity of the QSSOR which is approximately 75% and 50% less than FSSOR and HSSOR methods respectively, see Table 4.

ACKNOWLEDGEMENT

The authors acknowledge the Postgraduate Research Grant, Universiti Malaysia Sabah (GPS0003-SG-1/2009) for the completion of this article.

REFERENCES

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

Abdullah, A.R. & Ali, N.H.M. (1996). A comparative study of parallel strategies for the solution of elliptic pde's. *Parallel Algorithms and Applications*, *10*, 93-103.

Abdullah, M.H., Sulaiman, J. & Othman, A. (2006). A numerical assessment on water quality model using the Half-Sweep Explicit Group methods. *Gading*, *10*, 99-110.

Băutu, E., Băutu, A. & Luchian, H. (2005). A GEP-based approach for solving Fredholm first kind integral equations. In D. Zaharie *et al*. (Eds.), *Seventh International Symposium on Symbolic and Numeric Algorithms for Scientific Computing* (p. 325-328). Timisoara, Romania: IEEE.

Boland, W.R. (1972). The numerical solution of Fredholm integral equations using product type quadrature formulas. *BIT Numerical Mathematics*, *12*(1), 5-16.

Hasan, M.K., Othman, M., Abbas, Z., Sulaiman, J. & Ahmad, F. (2007). Parallel solution of high speed low order FDTD on 2D free space wave propagation. In O. Gervasi and M. Gavrilova (Eds.), *Lecture Notes in Computer Science* (p. 13-24). Berlin: Springer-Verlag.

Hasan, M.K., Othman, M., Johari, R., Abbas, Z. & Sulaiman, J. (2005). The HSLO(3)-FDTD with direct-domain and temporary-domain approaches on infinite space wave propagation. In B.M. Ali *et al*. (Eds.), *13th IEEE International Conference on Network* (p. 1002-1007). Kuala Lumpur, Malaysia: IEEE.

Hasan, M.K., Sulaiman, J. & Othman, M. (2008). Implementation of red black strategy to quarter-sweep iteration for solving first order hyperbolic equations. In H.B. Zaman *et al*. (Eds.), *International Symposium on Information Technology* (p. 1864-1869). Kuala Lumpur, Malaysia: IEEE.

Hsiao, G.C., Kopp, P. & Wendland, W.L. (1980). A Galerkin Collocation method for some integral equations of the first kind. *Computing*, *25*, 89-130.

Maleknejad, K., Mollapourasl, R. & Nouri, K. (2006). Convergence of numerical solution of the Fredholm integral equation of the first kind with degenerate kernel. *Applied Mathematics and Computation*, *181*, 1000-1007.

Muthuvalu, M.S. & Sulaiman, J. (2008). Half-Sweep Geometric Mean method for solution of linear Fredholm equations. *Matematika*, *24*(1), 75-84.

Muthuvalu, M.S. & Sulaiman, J. (2009). Half-Sweep Arithmetic Mean method with high-order Newton-Cotes quadrature schemes to solve linear second kind Fredholm equations. *Journal of Fundamental Sciences*, *5*(1), 7-16.

Oladejo, S.O., Mojeed, T.A. & Olurode, K.A. (2008). The application of cubic spline collocation to the solution of integral equations. *Journal of Applied Sciences Research*, *4*(6), 748-753.

Othman, M. & Abdullah, A.R. (2000). An efficient Four Points Modified Explicit Group Poisson solver. *International Journal of Computer Mathematics*, *76*, 203-217.

Othman, M. & Abdullah, A.R. (2001). Implementation of the Parallel Four Points Modified Explicit Group Iterative Algorithm on Shared Memory Parallel Computer. In V. Malyshkin (Ed.), *Lectures Notes in Computer Science* (p. 480-489). Berlin: Springer-Verlag.

Othman, M., Sulaiman, J. & Abdullah, A.R. (2000). A parallel halfsweep multigrid algorithm on the shared memory multiprocessors. *Malaysian Journal of Computer Science*, *13*(2), 1-6.

Polyanin, A.D. & Manzhirov, A.V. (1998). *Handbook of Integral Equations*. CRC Press LLC.

Rashed, M.T. (2003). An expansion method to treat integral equations. *Applied Mathematics and Computation*, *135*, 65-72.

Shang, X., Han, D. (2007). Numerical solution of Fredholm integral equations of the first kind by using linear Legendre multi-wavelets. *Applied Mathematics and Computation*, *191*, 440-444.

Sulaiman, J., Hasan, M.K., Othman, M. (2004). The Half-Sweep Iterative Alternating Decomposition Explicit (HSIADE) method for diffusion equation. In J. Zhang *et al*. (Eds.), *Lectures Notes in Computer Science* (p. 57-63). Berlin: Springer-Verlag.

Sulaiman, J., Hasan, M.K., Othman, M. (2007). Red-Black Half-Sweep iterative method using triangle finite element approximation for 2D Poisson equations. In Y. Shi *et al*. (Eds.), *Lectures Notes in Computer Science* (p. 326-333). Berlin: Springer-Verlag.

Sulaiman, J., Othman, M., Hasan, M.K. (2004). Quarter-Sweep Iterative Alternating Decomposition Explicit algorithm applied to diffusion equations. *International Journal of Computer Mathematics*, *81*(12), 1559-1565.

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

Sulaiman, J., Saudi, A., Abdullah, M.H., Hasan, M.K., Othman, M. (2008). Quarter-Sweep Arithmetic Mean algorithm for water quality model. In H.B. Zaman *et al*. (Eds.), *International Symposium on Information Technology* (p. 1859-1863). Kuala Lumpur, Malaysia: IEEE.

Yousif, W.S., Evans, D.J. (1995). Explicit De-coupled Group iterative methods and their implementations. *Parallel Algorithms and Applications*, *7*, 53-71.