ISSN 1 746-7233, England, UK
World Journal of Modelling and Simulation
Vol. 8 (2012) No. 4, pp. 293-301

An Implementation of the 2-Point Block Arithmetic Mean Iterative Method for First Kind Linear Fredholm Integral Equations

Mohana Sundaram Muthuvalu[1*], Jumat Sulaiman[2]

[1] Department of Fundamental and Applied Sciences, Faculty of Science and Information Technology, Universiti Teknologi PETRONAS, Tronoh 31750, Malaysia
[2] School of Science and Technology, Universiti Malaysia Sabah, Kota Kinabalu 88400, Malaysia

(*Received August 18 2011, Accepted June 6 2012*)

Abstract. In recent decades, many researches involving Arithmetic Mean (AM) iterative methods for solving matrix equations that arise from various scientific problems have been conducted. In this paper, application of the 2-Point Block Arithmetic Mean (2-BLAM) method to solve first kind linear Fredholm integral equations with semi-smooth kernel is investigated. The formulation and implementation of the method are discussed. Furthermore, numerical results of the method on test problems are also included.

Keywords: Fredholm integral equations, semi-smooth kernel, composite Newton-Cotes quadrature, Block Arithmetic Mean method

1 Introduction

It is well known that iterative methods are applied extensively in large scale computations for solving matrix equations problems. Among the existing iterative methods, Arithmetic Mean (AM) iterative methods have been applied widely for solving various types of matrix equations problems. In a series of papers, the effectiveness of the AM method and its variants were studied and tested on linear and nonlinear systems, refer Benzi and Dayar [1], Galligani [2], Galligani and Ruggiero [7], Galligani and Ruggiero [5], Hasan et al. [8], Muthuvalu and Sulaiman [11, 13], Ruggiero and Galligani [15] and Sulaiman et al. [16, 17]. Besides that, the AM method also has been successfully applied as a preconditioner with Conjugate Gradient (CG) method for solving symmetric positive definite linear system[3, 6] and with first degree iterative method to compute the minimal eigenpair of generalized eigen problem[4]. In this paper, application one of the AM iterative methods i.e. 2-Point Block Arithmetic Mean (2-BLAM) method will be utilized in solving large dense linear system associated with the first kind linear Fredholm integral equations with semi-smooth kernel. Previously, the 2-BLAM method has been successfully applied for second kind linear Fredholm integral equations[11].

The rest of this paper is organized as follows. In Section 2, the formulation of the approximation equation based on first order composite closed Newton-Cotes quadrature method will be elaborated. The latter section of this paper will discuss the formulations of the 2-BLAM method, and numerical results from the simulations will be shown in fourth section to evaluate the performance of the 2-BLAM method. Meanwhile, concluding remarks are given in the final section.

2 Newton-cotes quadrature approximation equation

The standard mathematical form of first kind linear Fredholm integral equations can be defined as follows

* Corresponding author. Tel.: +60 16 8239739. *E-mail address*: sundaram_at2@yahoo.com.

$$\int_a^b K(x,t)\,\phi(t)\,dt = f(x), \tag{1}$$

where the kernel K and the function f are given, and ϕ is the unknown function to be determined. If operator κ is denoted by

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

then the following definitions is satisfied.

Definition 1. *(Maleknejad et al. [10]) Let $\kappa : S \to T$ be an operator from normed space S into a normed space T, the equation $\kappa\phi = f$ is called well-posed if κ is onto, one to one and the inverse operator $\kappa^{-1} : T \to S$ is continuous. Otherwise the equation is called ill-posed.*

Definition 2. *(Kang et al. [9]) A kernel $K(x,t)$ is called $q-$semi-smooth if*

$$K(x,t) = \begin{cases} K_1(x,t) & if\ a \le t \le x \\ K_2(x,t) & if\ x \le t \le b \end{cases}$$

where $K_{1,2}(x,t) \in C^q_{[a,b]\,\times\,[a,b]}$ for some $q > 1$.

To formulate an approximation equation based on composite closed Newton-Cotes quadrature method for problem Eq. (1), let integration domain of problem Eq. (1) be divided uniformly into N subintervals and the discrete points of x and t given by $x_i = a + ih$ and $t_j = a + jh$ where the constant step width, h is defined as follows

$$h = \frac{b-a}{N}. \tag{3}$$

Before further explanations, the following notation will be used for simplicity

$$\begin{cases} K_{i,j} = K(x_i, t_j) \\ \hat{\phi}_j = \hat{\phi}(t_j) \\ f_i = f(x_i) \end{cases} \tag{4}$$

As discussed in [12, 13], application of the composite closed Newton-Cotes quadrature method reduces Eq. (1) to

$$\sum_{j=0}^{N} w_j K_{i,j}\,\hat{\phi}_j = f_i \quad , \quad i = 0,\,1,\,2,\,\cdots,N-2,\,N-1,N, \tag{5}$$

where solution $\hat{\phi}$ is an approximation of the exact solution ϕ to Eq. (1) and w_j is the weights of quadrature method. Moreover, Eq. (5) can be represented in matrix form as

$$A\hat{\phi} = f, \tag{6}$$

where

$$A=\begin{bmatrix} w_0K_{0,0} & w_1K_{0,1} & w_2K_{0,2} & \cdots & w_{N-2}K_{0,N-2} & w_{N-1}K_{0,N-1} & w_NK_{0,N} \\ w_0K_{1,0} & w_1K_{1,1} & w_2K_{1,2} & \cdots & w_{N-2}K_{1,N-2} & w_{N-1}K_{1,N-1} & w_NK_{1,N} \\ w_0K_{2,0} & w_1K_{2,1} & w_2K_{2,2} & \cdots & w_{N-2}K_{2,N-2} & w_{N-1}K_{2,N-1} & w_NK_{2,N} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ w_0K_{N-2,0} & w_1K_{N-2,1} & w_2K_{N-2,2} & \cdots & w_{N-2}K_{N-2,N-2} & w_{N-1}K_{N-2,N-1} & w_NK_{N-2,N} \\ w_0K_{N-1,0} & w_1K_{N-1,1} & w_2K_{N-1,2} & \cdots & w_{N-2}K_{N-1,N-2} & w_{N-1}K_{N-1,N-1} & w_NK_{N-1,N} \\ w_0K_{N,0} & w_1K_{N,1} & w_2K_{N,2} & \cdots & w_{N-2}K_{N,N-2} & w_{N-1}K_{N,N-1} & w_NK_{N,N} \end{bmatrix}_{(N+1)\times(N+1)}$$

$$\hat{\phi} = \begin{bmatrix} \hat{\phi}_0 \\ \hat{\phi}_1 \\ \hat{\phi}_2 \\ \vdots \\ \hat{\phi}_{N-2} \\ \hat{\phi}_{N-1} \\ \hat{\phi}_N \end{bmatrix}, \quad f = \begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ \vdots \\ f_{N-2} \\ f_{N-1} \\ f_N \end{bmatrix}$$

In order to facilitate the formulation of the composite closed Newton-Cotes quadrature method, further discussion will be restricted onto first order scheme. Based on the first order composite closed Newton-Cotes quadrature method, quadrature weights, w_j will satisfy the following relation

$$w_j = \begin{cases} \frac{h}{2} & j = 0, N \\ h & \text{otherwise} \end{cases} \tag{7}$$

3 2-point block arithmetic mean iterative method

In this section, the formulation and implementation of the 2-BLAM method to solve generated full linear system Eq. (6) will be discussed. Essentially, each iteration of the 2-BLAM method consists of solving two independent systems i.e. $\hat{\phi}^1$ and $\hat{\phi}^2$. For the 2-BLAM method, the diagonal of matrix A will be grouped into square matrix, where each square is a nonsingular 2×2 matrix. Further discussions of the 2-BLAM method will be restricted for the case N is even which will generate incomplete block with one ungrouped.

Let elements of matrix A Eq. (6) partitioned into two types of submatrices as follow

$$A = \begin{bmatrix} G_{0,0} & G_{0,2} & G_{0,4} & \cdots & G_{0,N-4} & G_{0,N-2} & G_{0,N} \\ G_{2,0} & G_{2,2} & G_{2,4} & \cdots & G_{2,N-4} & G_{2,N-2} & G_{2,N} \\ G_{4,0} & G_{4,2} & G_{4,4} & \cdots & G_{4,N-4} & G_{4,N-2} & G_{4,N} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ G_{N-4,0} & G_{N-4,2} & G_{N-4,4} & \cdots & G_{N-4,N-4} & G_{N-4,N-2} & G_{N-4,N} \\ G_{N-2,0} & G_{N-2,2} & G_{N-2,4} & \cdots & G_{N-2,N-4} & G_{N-2,N-2} & G_{N-2,N} \\ G_{N,0} & G_{N,2} & G_{N,4} & \cdots & G_{N,N-4} & G_{N,N-2} & G_{N,N} \end{bmatrix} \tag{8}$$

$$A = \begin{bmatrix} G^*_{0,0} & G^*_{0,2} & G^*_{0,4} & \cdots & G^*_{0,N-4} & G^*_{0,N-2} & G^*_{0,N} \\ G^*_{2,0} & G^*_{2,2} & G^*_{2,4} & \cdots & G^*_{2,N-4} & G^*_{2,N-2} & G^*_{2,N} \\ G^*_{4,0} & G^*_{4,2} & G^*_{4,4} & \cdots & G^*_{4,N-4} & G^*_{4,N-2} & G^*_{4,N} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ G^*_{N-4,0} & G^*_{N-4,2} & G^*_{N-4,4} & \cdots & G^*_{N-4,N-4} & G^*_{N-4,N-2} & G^*_{N-4,N} \\ G^*_{N-2,0} & G^*_{N-2,2} & G^*_{N-2,4} & \cdots & G^*_{N-2,N-4} & G^*_{N-2,N-2} & G^*_{N-2,N} \\ G^*_{N,0} & G^*_{N,2} & G^*_{N,4} & \cdots & G^*_{N,N-4} & G^*_{N,N-2} & G^*_{N,N} \end{bmatrix} \tag{9}$$

where notations of $G_{i,j}$ and $G^*_{i,j}$ are defined as follows

$$G_{i,j} = \begin{bmatrix} w_j K_{i,j} & w_{j+1} K_{i,j+1} \\ w_j K_{i+1,j} & w_{j+1} K_{i+1,j+1} \end{bmatrix}_{(2\times 2)} \tag{10}$$

and

$$G^*_{i,j} = \begin{bmatrix} w_{j-1} K_{i-1,j-1} & w_j K_{i-1,j} \\ w_{j-1} K_{i,j-1} & w_j K_{i,j} \end{bmatrix}_{(2\times 2)} \tag{11}$$

respectively. Based on matrices Eq. (8) and Eq. (9), it is noticeable that $G_{N,N} = w_N K_{N,N}$ and $G^*_{0,0} = w_0 K_{0,0}$. Meanwhile, $G_{N,j}$ $(j = 0, 2, 4, \cdots, N-4, N-2)$ and $G^*_{0,j}$ $(j = 2, 4, \cdots, N-4, N-2, N)$ are row vector of order two. Whereas, $G_{i,N}$ $(i = 0, 2, 4, \cdots, N-4, N-2)$ and $G^*_{i,0}$ $(j = 2, 4, \cdots, N-4, N-2, N)$ are column vector of order two.

Now, let consider the following two splittings of the matrix A as follows

$$A = D_1 - L_1 - U_1, \tag{12}$$
$$A = D_2 - L_2 - U_2. \tag{13}$$

Based on Eqs. (8) and (12), splittings of D_1, $-L_1$ and $-U_1$ are

$$D_1 = \begin{bmatrix} G_{0,0} & & & & & & \\ & G_{2,2} & & & & & \\ & & G_{4,4} & & 0 & & \\ & & & \ddots & & & \\ & & 0 & & G_{N-4,N-4} & & \\ & & & & & G_{N-2,N-2} & \\ & & & & & & G_{N,N} \end{bmatrix} \tag{14}$$

$$-L_1 = \begin{bmatrix} & & & & & \\ G_{2,0} & & & & & \\ G_{4,0} & G_{4,2} & & & 0 & \\ \vdots & \vdots & \ddots & & & \\ G_{N-4,0} & G_{N-4,2} & \cdots & G_{N-4,N-6} & & \\ G_{N-2,0} & G_{N-2,2} & \cdots & G_{N-2,N-6} & G_{N-2,N-4} & \\ G_{N,0} & G_{N,2} & \cdots & G_{N,N-6} & G_{N,N-4} & G_{N,N-2} \end{bmatrix} \tag{15}$$

$$-U_1 = \begin{bmatrix} G_{0,2} & G_{0,4} & G_{0,6} & \cdots & G_{0,N-2} & G_{0,N} \\ & G_{2,4} & G_{2,6} & \cdots & G_{2,N-2} & G_{2,N} \\ & & G_{4,6} & \cdots & G_{4,N-2} & G_{4,N} \\ & & & \ddots & \vdots & \vdots \\ & 0 & & & G_{N-4,N-2} & G_{N-4,N} \\ & & & & & G_{N-2,N} \\ & & & & & \end{bmatrix} \tag{16}$$

respectively. By using the same approach in first splitting, D_2, $-L_2$ and $-U_2$ based on Eqs. (9) and (13) satisfy the following relations

$$D_2 = \begin{bmatrix} G^*_{0,0} & & & & & & \\ & G^*_{2,2} & & & & & \\ & & G^*_{4,4} & & 0 & & \\ & & & \ddots & & & \\ & & 0 & & G^*_{N-4,N-4} & & \\ & & & & & G^*_{N-2,N-2} & \\ & & & & & & G^*_{N,N} \end{bmatrix} \tag{17}$$

$$-L_2 = \begin{bmatrix} G^*_{2,0} & & & & & \\ G^*_{4,0} & G^*_{4,2} & & & 0 & \\ \vdots & \vdots & \ddots & & & \\ G^*_{N-4,0} & G^*_{N-4,2} & \cdots & G^*_{N-4,N-6} & & \\ G^*_{N-2,0} & G^*_{N-2,2} & \cdots & G^*_{N-2,N-6} & G^*_{N-2,N-4} & \\ G^*_{N,0} & G^*_{N,2} & \cdots & G^*_{N,N-6} & G^*_{N,N-4} & G^*_{N,N-2} \end{bmatrix} \quad (18)$$

$$-U_2 = \begin{bmatrix} G^*_{0,2} & G^*_{0,4} & G^*_{0,6} & \cdots & G^*_{0,N-2} & G^*_{0,N} \\ & G^*_{2,4} & G^*_{2,6} & \cdots & G^*_{2,N-2} & G^*_{2,N} \\ & & G^*_{4,6} & \cdots & G^*_{4,N-2} & G^*_{4,N} \\ & & & \ddots & \vdots & \vdots \\ & 0 & & & G^*_{N-4,N-2} & G^*_{N-4,N} \\ & & & & & G^*_{N-2,N} \end{bmatrix} \quad (19)$$

respectively.

Thus, for nonsingular $(D_1 - \omega L_1)$ and $(D_2 - \omega U_2)$ matrices, the general formulation for 2-BLAM iterative method is defines as

$$\begin{cases} (D_1 - \omega L_1)\,\hat{\phi}^1 = [(1-\omega)\,D_1 + \omega U_1]\,\hat{\phi}^{(k)} + \omega f \\ (D_2 - \omega U_2)\,\hat{\phi}^2 = [(1-\omega)\,D_2 + \omega L_2]\,\hat{\phi}^{(k)} + \omega f \\ \hat{\phi}^{(k+1)} = \frac{\hat{\phi}^1 + \hat{\phi}^2}{2} \end{cases} \quad (20)$$

where ω denotes the acceleration parameter. By considering Eq. (20), algorithm for 2-BLAM method associated with composite closed Newton-Cotes quadrature approximation equation (as derived in Eq. (5)) to solve problem Eq. (1) would be described in Algorithm 1.

During the iteration process, LU decomposition method based on Crout technique will be applied to compute values of $\hat{\phi}^1$ and $\hat{\phi}^2$ at each block ($G_{i,i}$ and $G^*_{i,i}$). By utilizing Crout technique, formulations to calculate $\hat{\phi}^1_{i+1}$, $\hat{\phi}^1_i$, $\hat{\phi}^2_i$ and $\hat{\phi}^2_{i-1}$ are

$$\hat{\phi}^1_{i+1} = \frac{S^1_{i+1} - (w_i K_{i+1,i})\left(\frac{S^1_i}{w_i K_{i,i}}\right)}{(w_{i+1} K_{i+1,i+1}) - (w_i K_{i+1,i})\left(\frac{w_{i+1} K_{i,i+1}}{w_i K_{i,i}}\right)}, \quad (21)$$

$$\hat{\phi}^1_i = \frac{S^1_i}{w_i K_{i,i}} - \frac{w_{i+1} K_{i,i+1}}{w_i K_{i,i}} \hat{\phi}^{i+1}, \quad (22)$$

$$\hat{\phi}^2_i = \frac{S^2_i - (w_{i-1} K_{i,i-1})\left(\frac{S^2_{i-1}}{w_{i-1} K_{i-1,i-1}}\right)}{(w_i K_{i,i}) - (w_{i-1} K_{i,i-1})\left(\frac{w_i K_{i-1,i}}{w_{i-1} K_{i-1,i-1}}\right)}, \quad (23)$$

$$\hat{\phi}^2_{i-1} = \frac{S^2_{i-1}}{w_{i-1} K_{i-1,i-1}} - \frac{w_i K_{i-1,i}}{w_{i-1} K_{i-1,i-1}} \hat{\phi}^2_i, \quad (24)$$

respectively, where

$$s^1_i = f_i - \sum_{j=0}^{i-1} w_j K_{i,j} \hat{\phi}^1_j - \sum_{j=i+2}^{N} w_j K_{i,j} \hat{\phi}^{(k)}_j,$$

$$s^1_{i+1} = f_{i+1} - \sum_{j=0}^{i-1} w_j K_{i+1,j} \hat{\phi}^1_j - \sum_{j=i+2}^{N} w_j K_{i+1,j} \hat{\phi}^{(k)}_j,$$

$$s_{i-1}^2 = f_{i-1} - \sum_{j=0}^{i-2} w_j K_{i-1,j} \hat{\phi}_j^{(k)} - \sum_{j=i+1}^{N} w_j K_{i-1,j} \hat{\phi}_j^2,$$
$$s_i^2 = f_i - \sum_{j=0}^{i-2} w_j K_{i,j} \hat{\phi}_j^{(k)} - \sum_{j=i+1}^{N} w_j K_{i,j} \hat{\phi}_j^2$$
$$S_i^1 = (1-\omega)\left[w_i K_{i,i} \hat{\phi}_i^{(k)} + w_{i+1} K_{i,i+1} \hat{\phi}_{i+1}^{(k)}\right] + \omega\left(s_i^1\right),$$
$$S_{i+1}^1 = (1-\omega)\left[w_i K_{i+1,i} \hat{\phi}_i^{(k)} + w_{i+1} K_{i+1,i+1} \hat{\phi}_{i+1}^{(k)}\right] + \omega\left(s_{i+1}^1\right),$$
$$S_{i-1}^2 = (1-\omega)\left[w_{i-1} K_{i-1,i-1} \hat{\phi}_{i-1}^{(k)} + w_i K_{i-1,i} \hat{\phi}_i^{(k)}\right] + \omega\left(s_{i-1}^2\right),$$
$$S_i^2 = (1-\omega)\left[w_{i-1} K_{i,i-1} \hat{\phi}_{i-1}^{(k)} + w_i K_{i,i} \hat{\phi}_i^{(k)}\right] + \omega\left(s_i^2\right).$$

Meanwhile, for the ungrouped point, standard AM method based on point approach as described in [13, 15] is applied.

3.1 Algorithm 1. 2-blam algorithm

i. Set $\hat{\phi}^0 = \hat{\phi}^1 = \hat{\phi}^2$, ε.
ii. Iteration cycle
a. Stage 1 1. Level 1
for $i = 0,\ 2,\ 4,\ \cdots, N-6, N-4,\ N-2$

Compute $G_{i,i}\begin{bmatrix} \hat{\phi}_i \\ \hat{\phi}_{i+1} \end{bmatrix}^1 \leftarrow (1-\omega)\, G_{i,i}\begin{bmatrix} \hat{\phi}_i \\ \hat{\phi}_{i+1} \end{bmatrix}^{(k)} + \omega\begin{bmatrix} s_i \\ s_{i+1} \end{bmatrix}^1.$

for $i = N$

Compute $\hat{\phi}_N^1 \leftarrow (1-\omega)\,\hat{\phi}_N^{(k)} + \frac{\omega}{w_N K_{N,N}}\left[f_N - \sum_{j=0}^{N-1} w_j K_{N,j} \hat{\phi}_j^1\right].$

b. Stage 2 1. Level 2
for $i = N,\ N-2,\ N-4,\ \cdots, 6,\ 4,\ 2$

Compute $G_{i,i}^*\begin{bmatrix} \hat{\phi}_{i-1} \\ \hat{\phi}_i \end{bmatrix}^2 \leftarrow (1-\omega)\, G_{i,i}^*\begin{bmatrix} \hat{\phi}_{i-1} \\ \hat{\phi}_i \end{bmatrix}^{(k)} + \omega\begin{bmatrix} s_{i-1} \\ s_i \end{bmatrix}^2.$

for $i = 0$

Compute $\hat{\phi}_0^2 \leftarrow (1-\omega)\,\hat{\phi}_0^{(k)} + \frac{\omega}{w_0 K_{0,0}}\left[f_0 - \sum_{j=1}^{N} w_j K_{0,j} \hat{\phi}_j^2\right].$

2.
for $i = 0,\ 1,\ 2,\ \cdots, N-2,\ N-1, N$

Compute $\hat{\phi}_i^{(k+1)} \leftarrow \frac{\hat{\phi}_i^1 + \hat{\phi}_i^2}{2}.$

iii. Check the convergence. If the converge criterion i.e. the maximum norm $\left\|\hat{\phi}^{(k+1)} - \hat{\phi}^{(k)}\right\| \le \varepsilon$ (where ε is the convergence criterion) is satisfied, go to Step (iv), otherwise, repeat the iteration cycle (i.e., go to Step (ii)).
iv. Stop.

4 Numerical tests

In order to investigate the performance of the 2-BLAM iterative method, several numerical tests were carried out on the following two first kind linear Fredholm integral equations with semi-smooth kernel.

Text problem 1 (Rashed [14])

$$\int_0^1 K(x,t)\,\phi(t)\,dt = \frac{(x^3 - x)}{6}, \quad x \in (0,1), \tag{25}$$

Table 1. Numerical results of the GS, AM and 2-BLAM iterative methods for test problem Eq. (1)

Methods	N			
	Number of iterations			
	240	480	960	1920
GS	303	375	451	540
AM	135	136	139	140
2-BLAM	31	31	32	32
	Execution time (seconds)			
	240	480	960	1920
GS	4.29	13.39	52.31	183.39
AM	3.25	10.75	37.19	127.33
2-BLAM	0.64	2.07	13.93	53.71
	Maximum absolute error			
	240	480	960	1920
GS	6.847871 E-10	7.368467 E-10	9.065918 E-10	8.796745 E-10
AM	8.699241 E-10	1.346541 E-09	8.040829 E-10	7.487605 E-10
2-BLAM	7.019740 E-11	6.325473 E-11	4.231797 E-10	7.220884 E-10

with kernel

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

The exact solution of problem Eq. (25) is

$$\varphi(x) = x.$$

Text problem 2 (Rashed [14])

$$\int_0^1 K(x,t)\ \phi(t)\,dt = e^x + (1-e)\,x - 1, \quad x \in (0,1), \tag{26}$$

with kernel

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

and the exact solution is given by

$$\phi(x) = e^x.$$

For the numerical tests, there criteria i.e. number of iterations, execution time and maximum absolute error have been measured. Throughout the tests, the convergence test considered the tolerance error, $\varepsilon = 10^{-10}$ and carried out on several different values of N. Meanwhile, the experimental values of ω were obtained within ± 0.01 by running the 2-BLAM algorithm for different values of ω and choosing the one(s) that produces the minimum number of iterations. In addition, numerical results by applying the standard AM and Gauss-Seidel (GS) methods for solving linear system Eq. (6) are also included. All the tested algorithms were implemented by a computer with processor Intel(R) Core (TM) 2 CPU 1.66GHz and algorithm codes were written in C programming language. The numerical results of the tested methods for Examples 1 and 2 are tabulated in Tabs. 1 and 2 respectively. Meanwhile, reduction percentages in terms of number of iterations and execution time for the AM and 2-BLAM methods compared with GS method have been summarized in Tab. 3.

Table 2. Numerical results of the GS, AM and 2-BLAM iterative methods for test problem 2

Methods	N			
	Number of iterations			
	240	480	960	1920
GS	315	388	470	559
AM	141	144	145	145
2-BLAM	33	33	35	35
	Execution time (seconds)			
	240	480	960	1920
GS	4.44	14.83	58.32	195.66
AM	3.65	12.19	39.74	132.61
2-BLAM	0.69	2.46	15.73	59.63
	Maximum absolute error			
	240	480	960	1920
GS	3.916332 E-06	9.810628 E-07	2.453851 E-07	6.290920 E-08
AM	3.915194 E-06	9.806430 E-07	2.449272 E-07	6.115767 E-08
2-BLAM	3.916333 E-06	9.810656 E-07	2.451143 E-07	6.126630 E-08

Table 3. Reduction percentages of the AM and 2-BLAM methods compared with GS method

Methods	Number of iterations	
	Test Problem 1 (%)	Test Problem 2 (%)
AM	$55.44 \sim 74.08$	$55.23 \sim 74.07$
2-BLAM	$89.76 \sim 94.08$	$89.52 \sim 93.74$
Methods	Execution time	
	Test Problem 1 (%)	Test Problem 2 (%)
AM	$19.71 \sim 30.57$	$17.79 \sim 32.23$
2-BLAM	$70.71 \sim 85.09$	$69.52 \sim 84.46$

5 Conclusions

In this paper, an application of the 2-BLAM iterative method for solving large full matrix arise from the first kind linear Fredholm integral equations with semi-smooth kernel is investigated. Through numerical results presented in Tabs. 1 and 2, it clearly shows that applications of the standard AM and 2-BLAM methods reduce number of iterations and execution time compared to the GS method. Meanwhile, among the tested methods, 2-BLAM method has the least number of iterations and compute with the fastest time. In terms of accuracy, approximate solutions obtained by using 2-BLAM method are in good agreement compared to the GS and AM methods. Finally, it can be concluded that the 2-BLAM method is superior compared to AM and GS methods in the sense of number of iterations and execution time.

References

[1] M. Benzi, T. Dayar. The arithmetic mean method for finding the stationary vector of markov chains. *International Journal of Parallel, Emergent and Distribution Systems*, 1995, **6**: 25–37.

[2] E. Galligani. The arithmetic mean method for solving systems of nonlinear equations in finite differences. *Applied Mathematics and Computation*, 2006, **181**: 579–597.

[3] E. Galligani, V. Ruggiero. The arithmetic mean preconditioner for multivector computers. *International Journal of Computer Mathematics*, 1992, **44**: 207–222.

[4] E. Galligani, V. Ruggiero. Computation of minimal eigenpair in the large sparse generalized eigen-problem using vector computers. *Parallel Computing*, 1992, 193–201.

[5] E. Galligani, V. Ruggiero. The two-stage arithmetic mean method. *Applied Mathematics and Computation*, 1997, **85**: 245–264.

[6] I. Galligani. A splitting method for large sparse linear systems on a multivector computer. *Computer Algorithms for Solving Linear Algebraic Equations: The State of the Art*, 1991, 207–215.

[7] I. Galligani, V. Ruggiero. The arithmetic mean method for solving essentially positive systems on a vector computer. *International Journal of Computer Mathematics*, 1989, **30**: 113–121.

[8] M. Hasan, J. Sulaiman, et al. Approximation of iteration number for gauss-seidel using redlich-kister polynomial. *American Journal of Applied Science*, 2010, **7**: 956–962.

[9] S. Kang, I. Koltracht, G. Rawitscher. Nyström-clenshaw-curtis quadrature for integral equations with discontinuous kernel. *Mathematics of Computation*, 2002, **72**: 729–756.

[10] K. Maleknejad, R. Mollapourasl, K. Nouri. Convergence of numerical solution of the fredholm integral equation of the first kind with degenerate kernel. *Applied Mathematics and Computation*, 2006, **181**: 1000–1007.

[11] M. Muthuvalu, J. Sulaiman. New block arithmetic mean method for solution of second kind linear fredholm integral equations. **in:** *Proceeding of the 16th National Symposium on Mathematical Sciences*, 2008, 87–94.

[12] M. Muthuvalu, J. Sulaiman. Quarter-sweep iteration for first kind linear fredholm integral equations. *International Journal of Open Problems in Computer Science and Mathematics*, 2010, **3**: 357–367.

[13] M. Muthuvalu, J. Sulaiman. Half-sweep arithmetic mean method with composite trapezoidal scheme for solving linear fredholm integral equations. *Applied Mathematics and Computation*, 2011, **217**: 5442–5448.

[14] M. Rashed. An expansion method to treat integral equations. *Applied Mathematics and Computation*, 2003, **135**: 65–72.

[15] V. Ruggiero, E. Galligani. An iterative method for large sparse systems on a vector computer. *Computers & Mathematics with Applications*, 1990, **20**: 25–28.

[16] J. Sulaiman, M. Othman, M. Hasan. A new half-sweep arithmetic mean (hsam) algorithm for two-point boundary value problems. **in:** *Proceeding of the International Conference on Statistics and Mathematics and Its Application in the Development of Science and Technology*, 169–173, 2004.

[17] J. Sulaiman, M. Othman, M. Hasan. A new quarter sweep arithmetic mean (qsam) method to solve diffusion equations. *Chamchuri Journal of Mathematics*, 2009, **1**: 93–103.