Malaya Journal of Matematik S(1)(2013) 77–84

Malaya Journal of Matematik

MJM
an international journal of mathematical sciences with computer applications...

www.malayajournal.org

ISSN : 2319-3786

Numerical solution of systems of linear Volterra integral equations using block-pulse functions

V. Balakumar, K. Murugesan*

Department of Mathematics, National Institute of Technology, Tiruchirappalli-620 015, Tamil Nadu, India.

Abstract:

This paper generalizes Block-Pulse Functions method for solving systems of linear Volterra integral equations of the second kind. This method, using operational matrix associated with Block-Pulse Functions, reduces these types of equations to a linear lower triangular system of algebraic equations. Numerical examples are presented to illustrate the computational efficiency of the method.

Keywords: Block-Pulse Functions, Numerical method, Operational matrix, Volterra integral equations.

2010 MSC: 65R20, 33D45, 45D05, 45F05.

1 Introduction

Volterra equations naturally appear in history-dependent problems such as population dynamics, renewal equations, nuclear reactor dynamics, viscoelasticity, study of epidemics, superfluidity, damped vibrations, heat conduction and diffusion [7]. Systems of Volterra integral equations have wide applications in engineering, physics, chemistry and populations growth models [14]. We consider the following system of linear Volterra integral equations (SLVIEs) of the second kind [7].

$$\mathbf{F}(t) = \mathbf{G}(t) + \int_0^t \mathbf{K}(t,s)\,\mathbf{F}(s)\,\mathrm{d}s, \qquad 0 \le t \le 1 \tag{1.1}$$

$$\begin{aligned} \text{where}\quad \mathbf{F}(t) &= \begin{bmatrix} f_1(t) & f_2(t) & \dots & f_n(t) \end{bmatrix}^T \\ \mathbf{G}(t) &= \begin{bmatrix} g_1(t) & g_2(t) & \dots & g_n(t) \end{bmatrix}^T \\ \mathbf{K}(t,s) &= \begin{bmatrix} k_{pq}(t,s) \end{bmatrix}, \qquad p,q = 1,2,\dots,n. \end{aligned}$$

In this system, $\mathbf{G}, \mathbf{K}$ are given differentiable functions and $\mathbf{F}$ is the solution to be determined. Continuity of $\mathbf{G}$ and $\mathbf{K}$ guarantees the existence and uniqueness of the solution [1].

Several researchers have adopted different techniques for solving this system (1.1). Rabbani *et* al. have used a modified Taylor-series expansion method to reduce the system of integral equations to a linear system of Ordinary Differential Equations, which are solved by constructing appropriate boundary conditions [11]. Tahmasbi and Fard have presented a derivative-free method based on the power series method [13]. Mirzaee have used Rationalized Haar functions with their product operational matrix [10]. Biazar and Eslami have proposed a modified Homotopy Perturbation Method using a simple modification [4].

Block-Pulse functions have been used by many researchers for various problems such as solving differential equations [12], integral equations [6], population balance equations [5]. Recently, Maleknejad have used BPFs for solving Fredholm integral equations system [9]. Babolian presented a direct method to solve Volterra integral equation of the first kind using operational matrix with BPFs [2] . Numerical solution for a system of first kind Volterra integral equations is presented in [8]. In this paper, we present Block-Pulse Functions(BPFs)

*Corresponding author.

E-mail addresses : balakumar.math@gmail.com(V. Balakumar), murugu@nitt.edu(K. Murugesan)

method for the computation of numerical solution of system of linear Volterra integral equations (SLVIEs) of the second kind.

This paper is organized as follows. Section 2 defines BPFs and describes the mathematical preliminaries, which are relevant for the derivation that follows. Section 3 is concerned with the derivation of the proposed method. Section 4 demonstrates the efficiency of the method by numerical examples. Section 5 concludes the paper.

2 Block-Pulse Functions

This section describes the definition and properties of Block Pulse Functions. Function approximation using BPFs and operational matrix associated with BPFs have been discussed briefly.

2.1 Definition

An m-set of BPFs is defined as [12],

$$\phi_i(t) = \begin{cases} 1, & \frac{(i-1)T}{m} \leq t < \frac{iT}{m}, \\ 0, & \text{otherwise}, \end{cases} \tag{2.2}$$

where $t \in [0, T)$, $h = \frac{T}{m}$, $i = 1, 2, \ldots, m$ and ϕ_i is the i^{th} BPF.
In this article, it is assumed that $T = 1$, so the BPFs are defined over $t \in [0, 1)$. This set of functions can be concisely described by an m-vector

$$\Phi(t) = \begin{bmatrix} \phi_1(t) & \phi_2(t) & \ldots & \phi_m(t) \end{bmatrix}^T.$$

2.2 Properties

The most important properties of BPFs are disjointness, orthogonality and completeness [2, 8].
The disjointness property is,

$$\phi_i(t)\phi_j(t) = \delta_{ij}\phi_i(t) \tag{2.3}$$

for $i, j = 1, 2, \ldots, m$ and δ_{ij} is the Kronecker delta.
The orthogonality property is,

$$\int_0^1 \phi_i(t)\ \phi_j(t)\ \mathrm{d}t = h\delta_{ij} \tag{2.4}$$

for $i, j = 1, 2, \ldots, m$.
The completeness property is as follows:
For every $f \in \mathcal{L}^2([0, 1))$ when m approaches to the infinity, Parseval's identity holds:

$$\int_0^1 f^2(t)\ \mathrm{d}t = \sum_{i=1}^{m} f_i^2 \|\phi_i(t)\|^2 \tag{2.5}$$

where

$$f_i = \frac{1}{h}\int_0^1 f(t)\ \phi_i(t)\ \mathrm{d}t = m\int_{\frac{i-1}{m}}^{\frac{i}{m}} f(t)\ \mathrm{d}t. \tag{2.6}$$

It is easy to verify that $\Phi(t)\Phi^T(t) = \operatorname{diag}(\Phi(t))$ and $\Phi^T(t)\Phi(t) = 1$.
Let V be an m-vector and $\tilde{V} = \operatorname{diag}(V)$. Then,

$$\Phi(t)\Phi^T(t)V = \tilde{V}\Phi(t) \tag{2.7}$$

Further, for every $m \times m$ matrix B,

$$\Phi^T(t)B\Phi(t) = \hat{B}^T\Phi(t) \tag{2.8}$$

where $\hat{B}$ is an m-vector with elements equal to the diagonal entries of matrix B.

2.3 Operational matrix

Now, the integral $\int_0^t \phi_i(\tau)\ \mathrm{d}\tau$ is expanded in terms of BPFs. Arranging the coefficients of expansion in a matrix form, we have

$$\int_0^t \Phi(\tau)\ \mathrm{d}\tau \backsimeq P\Phi(t) \tag{2.9}$$

where $P_{m\times m}$ is given by:

$$P = \frac{h}{2}\begin{pmatrix} 1 & 2 & 2 & \cdots & 2 \\ 0 & 1 & 2 & \cdots & 2 \\ 0 & 0 & 1 & \cdots & 2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{pmatrix}.$$

The significance of the matrix P is that the integration of $\Phi(\tau)$ can be approximately (in the least-square error sense) achieved by premultiplying $\Phi(t)$ by the constant matrix P. For this reason, P is called the operational matrix for BPFs [12].

2.4 Function Approximation

The orthogonality property of BPFs is the base of expanding functions into their block pulse series [9]. For every $f \in \mathcal{L}^2([0,1))$,

$$f(t) \backsimeq \sum_{i=1}^{m} f_i\phi_i(t) = F^T\Phi(t) = \Phi^T(t)F,$$

where $F = \begin{bmatrix} f_1 & f_2 & \dots & f_m \end{bmatrix}^T$ and $\Phi(t) = \begin{bmatrix} \phi_1(t) & \phi_2(t) & \dots & \phi_m(t) \end{bmatrix}^T$

Also, any function $k(t,s) \in \mathcal{L}^2([0,1)\times[0,1))$ can similarly be expanded with respect to BPFs such as

$$k(t,s) \backsimeq \Phi^T(t)K\Psi(s),$$

where $\Phi(t)$ and $\Psi(s)$ are m_1 and m_2 dimensional BPF vectors respectively, and K is the $m_1 \times m_2$ block-pulse coefficient matrix with $k_{ij},\ i = 1,2,\dots,m_1,\ \ j = 1,2,\dots,m_2$, as follows:

$$k_{ij} = m_1 m_2 \int_0^1 \int_0^1 k(t,s)\ \phi_i(t)\ \psi_j(s)\ \mathrm{d}t\ \mathrm{d}s$$

In this article, it is assumed that $m_1 = m_2 = m$.

3 BPFs Method for System of Volterra Integral Equations

We consider the system of linear Volterra integral equations (1.1).
The system also can be written as

$$f_p(t) = g_p(t) + \sum_{q=1}^{n} \int_0^t k_{pq}(t,s)\ f_q(s)\ \mathrm{d}s, \qquad 0 \le t \le 1 \tag{3.10}$$

where $p = 1,2,\dots,n$.
BPF expansions for the functions f_p, f_q, g_p, k_{pq} can be written as:

$$\begin{aligned} f_p(t) &\backsimeq F_p^T\Phi(t) = \Phi^T(t)F_p \\ f_q(s) &\backsimeq F_q^T\Phi(s) = \Phi^T(s)F_q \\ g_p(t) &\backsimeq G_p^T\Phi(t) = \Phi^T(t)G_p \\ k_{pq}(t,s) &\backsimeq \Phi^T(t)K_{pq}\Phi(s) \end{aligned}$$

where $F_p = \begin{bmatrix} f_{p1} & f_{p2} & \ldots & f_{pm} \end{bmatrix}^T$ and $K_{pq} = \left[k_{pq}^{(i,j)}\right]$ with $i, j = 1, 2, \ldots, m$.
Now, the p^{th} equation, from the system (3.10), can be expanded in m-terms BPFs expansion as follows,

$$\begin{aligned} F_p^T \Phi(t) &\backsimeq G_p^T \Phi(t) + \sum_{q=1}^{n} \int_0^t \Phi^T(t)\ K_{pq}\ \Phi(s)\ \Phi^T(s)\ F_q\ \mathrm{d}s \\ &= G_p^T \Phi(t) + \Phi^T(t) \sum_{q=1}^{n} K_{pq} \int_0^t \Phi(s)\ \Phi^T(s)\ F_q\ \mathrm{d}s \end{aligned}$$

Using Eq.(2.7) gives,

$$\begin{aligned} F_p^T \Phi(t) &\backsimeq G_p^T \Phi(t) + \Phi^T(t) \sum_{q=1}^{n} K_{pq} \int_0^t \tilde{F}_q\ \Phi(s)\mathrm{d}s \\ &= G_p^T \Phi(t) + \Phi^T(t) \sum_{q=1}^{n} K_{pq} \tilde{F}_q \int_0^t \Phi(s)\mathrm{d}s \end{aligned}$$

Using Eq. (2.9) gives,

$$\begin{aligned} F_p^T \Phi(t) &\backsimeq G_p^T \Phi(t) + \Phi^T(t) \sum_{q=1}^{n} K_{pq} \tilde{F}_q P \Phi(t) \\ &= G_p^T \Phi(t) + \Phi^T(t) \left[K_{p1} \tilde{F}_1 P + K_{p2} \tilde{F}_2 P + \ldots + K_{pn} \tilde{F}_n P\right] \Phi(t) \end{aligned}$$

Using Eq. (2.8) gives,

$$\begin{aligned} F_p^T \Phi(t) &\backsimeq G_p^T \Phi(t) + \hat{F_p}^T \Phi(t) \\ F_p &\backsimeq G_p + \hat{F}_p \end{aligned} \tag{3.11}$$

where $\hat{F}_p$ is an m-vector with components equal to the diagonal entries of the $m \times m$ matrix $\sum_{q=1}^{n} K_{pq} \tilde{F}_q P$. We have $\tilde{F}_q = \mathrm{diag}(F_q)$. Now, $\hat{F}_p$ can be computed as

$$\hat{F}_p = \begin{pmatrix} \frac{h}{2} \sum\limits_{q=1}^{n} k_{pq}^{(1,1)} f_{q1} \\ h \sum\limits_{q=1}^{n} k_{pq}^{(2,1)} f_{q1} + \frac{h}{2} \sum\limits_{q=1}^{n} k_{pq}^{(2,2)} f_{q2} \\ \vdots \\ h \sum\limits_{q=1}^{n} k_{pq}^{(m,1)} f_{q1} + h \sum\limits_{q=1}^{n} k_{pq}^{(m,2)} f_{q2} + \ldots + \frac{h}{2} \sum\limits_{q=1}^{n} k_{pq}^{(m,m)} f_{qm} \end{pmatrix}$$

which can also be written as

$$\hat{F}_p = \begin{pmatrix} \frac{h}{2} \sum\limits_{q=1}^{n} k_{pq}^{(1,1)} & 0 & \cdots & 0 \\ h \sum\limits_{q=1}^{n} k_{pq}^{(2,1)} & \frac{h}{2} \sum\limits_{q=1}^{n} k_{pq}^{(2,2)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ h \sum\limits_{q=1}^{n} k_{pq}^{(m,1)} & h \sum\limits_{q=1}^{n} k_{pq}^{(m,2)} & \cdots & \frac{h}{2} \sum\limits_{q=1}^{n} k_{pq}^{(m,m)} \end{pmatrix} \begin{pmatrix} f_{q1} \\ f_{q2} \\ \vdots \\ f_{qm} \end{pmatrix}$$

Using Eq. (3.11) in Eq. (1.1) and replacing $\backsimeq$ with $=$ yields,

$$\begin{pmatrix} \mathbf{F}^{(1)} \\ \mathbf{F}^{(2)} \\ \vdots \\ \mathbf{F}^{(m)} \end{pmatrix} = \begin{pmatrix} \mathbf{G}^{(1)} \\ \mathbf{G}^{(2)} \\ \vdots \\ \mathbf{G}^{(m)} \end{pmatrix} + \begin{pmatrix} \frac{1}{2m}\mathbf{K}^{(1,1)} & 0 & \cdots & 0 \\ \frac{1}{m}\mathbf{K}^{(2,1)} & \frac{1}{2m}\mathbf{K}^{(2,2)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{m}\mathbf{K}^{(m,1)} & \frac{1}{m}\mathbf{K}^{(m,2)} & \cdots & \frac{1}{2m}\mathbf{K}^{(m,m)} \end{pmatrix} \begin{pmatrix} \mathbf{F}^{(1)} \\ \mathbf{F}^{(2)} \\ \vdots \\ \mathbf{F}^{(m)} \end{pmatrix}$$

This can also be written as

$$\mathbf{F}^{(i)} = \mathbf{G}^{(i)} + \frac{1}{m}\sum_{j=1}^{i-1}\mathbf{K}^{(i,j)}\mathbf{F}^{(j)} + \frac{1}{2m}\mathbf{K}^{(i,i)}\mathbf{F}^{(i)}, \quad i = 1, 2, \ldots, m \tag{3.12}$$

where $\mathbf{F}^{(i)} = \begin{bmatrix} f_{1i} & f_{2i} & \ldots & f_{ni} \end{bmatrix}^T$ and $\mathbf{K}^{(i,j)} = \begin{bmatrix} k_{pq}^{(i,j)} \end{bmatrix}$, $p, q = 1, 2, \ldots, n.$

Eq.(3.12) gives Block-Pulse coefficients recursively. Using these coefficients with $\mathbf{F}(t) = \begin{bmatrix} \mathbf{F}^{(1)} & \mathbf{F}^{(2)} & \ldots & \mathbf{F}^{(m)} \end{bmatrix} \Phi(t)$, numerical solutions can be easily computed.

4 Numerical Examples

Now, we employ the proposed BPF method to compute the numerical solution of some examples and compare it with their exact solutions. For each example, the exact solutions $f_i(t)$, computed numerical solutions $f_i^*(t)$ and maximum absolute errors $e_i(t) = \max\{|f_i^*(t) - f_i(t)|, 0 \le t < 1\}$, $i = 1, 2, \ldots, n$, for different values of m, are presented in Tables 1-6. The computed values, illustrated by these results, agrees well with the exact solutions. For higher m values, the method achieves better accuracy.

Example 1. Consider the system of Volterra integral equations, from [10],

$$\begin{cases} f_1(t) = t + \frac{t^3}{2} + \frac{t^4}{12} - \frac{t^5}{5} + \int\limits_0^t \left(s^2 - t\right)\left(f_1(s) + f_2(s)\right) ds \\ f_2(t) = t^2 - \frac{t^3}{3} - \frac{t^4}{4} + \int\limits_0^t s\left(f_1(s) + f_2(s)\right) ds \end{cases}$$

with the exact solutions $f_1(t) = t$ and $f_2(t) = t^2$. The numerical results are shown in Tables 1-2.

Table 1: Numerical results for Example 1

	$m = 32$		$m = 64$		Exact Solution	
t	$f_1^*(t)$	$f_2^*(t)$	$f_1^*(t)$	$f_2^*(t)$	$f_1(t)$	$f_2(t)$
0.0	0.0156	0.0003	0.0078	0.0001	0.0000	0.0000
0.1	0.1094	0.0121	0.1016	0.0103	0.1000	0.0100
0.2	0.2031	0.0414	0.1953	0.0382	0.2000	0.0400
0.3	0.2969	0.0883	0.3047	0.0929	0.3000	0.0900
0.4	0.3906	0.1527	0.3984	0.1588	0.4000	0.1600
0.5	0.5157	0.2660	0.5078	0.2579	0.5000	0.2500
0.6	0.6094	0.3715	0.6016	0.3619	0.6000	0.3600
0.7	0.7032	0.4946	0.6953	0.4835	0.7000	0.4900
0.8	0.7970	0.6353	0.8047	0.6476	0.8000	0.6400
0.9	0.8908	0.7936	0.8985	0.8073	0.9000	0.8100

Table 2: Maximum absolute errors for Example 1

m	$e_1(t)$	$e_2(t)$
32	1.5667e−02	1.6419e−02
64	7.8226e−03	7.9170e−03
128	3.9087e−03	4.2478e−03
256	1.9537e−03	1.9596e−03

Example 2. For the system, from [4],

$$\begin{cases} f_1(t) = 1 - \frac{t^2}{2} + \int\limits_0^t \left(f_1(s) + se^s f_2(s)\right) ds \\ f_2(t) = 1 + \frac{t^2}{2} + \int\limits_0^t \left(-se^{-s} f_1(s) - f_2(s)\right) ds \end{cases}$$

with the exact solutions $f_1(t) = e^t$ and $f_2(t) = e^{-t}$, the numerical results are given in Tables 3-4.

Table 3: Numerical results for Example 2

	$m = 32$		$m = 64$		Exact Solution	
t	$f_1^*(t)$	$f_2^*(t)$	$f_1^*(t)$	$f_2^*(t)$	$f_1(t)$	$f_2(t)$
0.0	1.0160	0.9845	1.0079	0.9922	1.0000	1.0000
0.1	1.1158	0.8964	1.1070	0.9034	1.1052	0.9048
0.2	1.2255	0.8162	1.2158	0.8226	1.2214	0.8187
0.3	1.3460	0.7432	1.3563	0.7374	1.3499	0.7408
0.4	1.4783	0.6767	1.4896	0.6714	1.4918	0.6703
0.5	1.6752	0.5971	1.6618	0.6018	1.6487	0.6065
0.6	1.8398	0.5437	1.8251	0.5480	1.8221	0.5488
0.7	2.0207	0.4950	2.0045	0.4989	2.0138	0.4966
0.8	2.2193	0.4507	2.2362	0.4472	2.2255	0.4493
0.9	2.4375	0.4103	2.4560	0.4072	2.4596	0.4066

Table 4: Maximum absolute errors for Example 2

m	$e_1(t)$	$e_2(t)$
32	2.6440e−02	1.5464e−02
64	1.3049e−02	7.7720e−03
128	6.4821e−03	3.8961e−03
256	3.2306e−03	1.9506e−03

Example 3. Finally, let us solve the following system of Volterra integral equations, taken from [3],

$$\begin{cases} f_1(t) = 1 + t^2 - \frac{t^3}{3} - \frac{t^4}{3} + \int\limits_0^t \left((t-s)^3 f_1(s) + (t-s)^2 f_2(s)\right) ds \\ f_2(t) = 1 - t - t^3 - \frac{t^4}{4} - \frac{t^5}{4} - \frac{t^7}{420} + \int\limits_0^t \left((t-s)^4 f_1(s) + (t-s)^3 f_2(s)\right) ds \end{cases}$$

with the exact solutions $f_1(t) = 1 + t^2$ and $f_2(t) = 1 + t - t^3$. The numerical results are shown in Table 5-6.

Table 5: Numerical results for Example 3

	$m = 32$		$m = 64$		Exact Solution	
t	$f_1^*(t)$	$f_2^*(t)$	$f_1^*(t)$	$f_2^*(t)$	$f_1(t)$	$f_2(t)$
0.0	1.0003	1.0156	1.0001	1.0078	1.0000	1.0000
0.1	1.0120	1.1080	1.0103	1.1005	1.0100	1.0990
0.2	1.0413	1.1947	1.0382	1.1879	1.0400	1.1920
0.3	1.0882	1.2706	1.0929	1.2764	1.0900	1.2730
0.4	1.1527	1.3309	1.1588	1.3352	1.1600	1.3360
0.5	1.2660	1.3784	1.2579	1.3768	1.2500	1.3750
0.6	1.3714	1.3830	1.3619	1.3838	1.3600	1.3840
0.7	1.4945	1.3554	1.4835	1.3591	1.4900	1.3570
0.8	1.6351	1.2907	1.6476	1.2836	1.6400	1.2880
0.9	1.7934	1.1840	1.8072	1.1732	1.8100	1.1710

Table 6: Maximum absolute errors for Example 3

m	$e_1(t)$	$e_2(t)$
32	1.6640e−02	1.5617e−02
64	7.8991e−03	7.8115e−03
128	4.2336e−03	3.9061e−03
256	1.9585e−03	1.9531e−03

5 Conclusion

Block-Pulse Functions method is presented for the numerical solution of linear system of Volterra integral equations of the second kind. Using operational matrix associated with Block-Pulse functions, the method converts the system of integral equations into a lower triangular system of algebraic equations.

As illustrated by examples, the proposed method gives accurate results. The value of $'m'$ can be selected as large enough to increase the accuracy.

References

[1] H. Aminikhah and J. Biazar, A new analytical method for solving systems of Volterra integral equations *International Journal of Computer Mathematics,* 87 (5) (2010), 1142-1157.

[2] E. Babolian and Z. Masouri, Direct method to solve Volterra integral equation of the first kind using operational matrix with block-pulse functions, *Journal of Computational and Applied Mathematics,* 220 (1) (2008), 51-57.

[3] M. I. Berenguer, D. Gamez, A. I. Garralda-Guillem,M. Ruiz Galan, and M. C. Serrano Perez, Biorthogonal systems for solving Volterra integral equation systems of the second kind *Journal of Computational and Applied Mathematics,* 235 (7) (2011), 1875-1883.

[4] J. Biazar and M. Eslami, Modified HPM for solving systems of Volterra integral equations of the second kind, *Journal of King Saud University-Science,* 23 (1) (2011), 35-39.

[5] H. Chyi and S. Yen-Ping, Solution of population balance equations via block pulse functions *The Chemical Engineering Journal,* 25 (1) (1982), 39-45.

[6] F. C. Kung and S. Y. Chen, Solution of integral equations using a set of block pulse functions, *Journal of the Franklin Institute,* 306 (4) (1978), 283-291.

[7] P. Linz, Analytical and numerical methods for Volterra equations, volume 7, SIAM, 1985.

[8] K. Maleknejad, H. Safdari, and M. Nouri, Numerical solution of an integral equations system of the first kind by using an operational matrix with block pulse functions, *International Journal of Systems Science,* 42 (1) (2011), 195-199.

[9] K. Maleknejad, M. Shahrezaee, and H. Khatami, Numerical solution of integral equations system of the second kind by BlockPulse functions, *Applied Mathematics and Computation,* 166 (1) (2005), 15-24.

[10] F. Mirzaee, Numerical computational solution of the linear Volterra integral equations system via rationalized Haar functions, *Journal of King Saud University-Science,* 22 (4) (2010), 265-268.

[11] M. Rabbani, K. Maleknejad, and N. Aghazadeh, Numerical computational solution of the Volterra integral equations system of the second kind by using an expansion method, *Applied Mathematics and Computation,* 187 (2) (2007), 1143-1146.

[12] P. Sannuti, Analysis and synthesis of dynamic systems via block-pulse functions, *Proceedings of the Institution of Electrical Engineers,* 124 (6) (1977), 569-571.

[13] A. Tahmasbi and O. S. Fard, Numerical solution of linear Volterra integral equations system of the second kind, *Applied Mathematics and Computation,* 201 (1) (2008), 547-552.

[14] A. M. Wazwaz, Linear and Nonlinear Integral Equations: Methods and Applications, Springer, 2011.

Received: August 16, 2013; *Accepted*: September 1, 2013

UNIVERSITY PRESS