Available online at www.sciencedirect.com
SCIENCE DIRECT®
Applied Mathematics and Computation 176 (2006) 324–333
APPLIED MATHEMATICS AND COMPUTATION
www.elsevier.com/locate/amc

Haar wavelet method for nonlinear integro-differential equations

Ülo Lepik

Institute of Applied Mathematics, University of Tartu, Liivi Str. 2, 50409 Tartu, Estonia

Abstract

A numerical method for solving nonlinear integral equations based on the Haar wavelets is presented. The method is applicable for Volterra integral equations and integro-differential equations; it can be used also for solving boundary value problems of ordinary differential equations.

The efficiency of the proposed method is tested with the aid of four examples. High accuracy even for a small number of collocation points is stated.

Keywords: Integro-differential equations; Numerical solution; Nonlinear; Haar wavelets; Collocation method

1. Introduction

Many problems of theoretical physics and other disciplines lead to nonlinear Volterra integral equations or integro-differential equations. For solving these equations several numerical approaches have been proposed, an overview can be found in the monograph [1]. Beginning from 1991 the wavelet method has been applied for solving integral equations, a short survey on these papers can be found in [2]. The solutions are often quite complicated and the advantages of the wavelet method get lost, therefore any kind of simplifications are welcome. One possibility for it is to make use of the Haar wavelets, which are mathematically the most simple wavelets.

The Haar wavelet method for solving linear integral equations of different type was proposed in [2] and for nonlinear Fredholm integral equations in [3].

In the present article the Haar wavelets are applied for solving of nonlinear Volterra integral equations and integro-differential equations. The proposed method is based on the collocation technique; the wavelet coefficients are calculated iteratively making use of the Newton method. The main idea of the method is to double the number of collocation points at each iteration. With minor changes the proposed method is applicable also for solving two-point boundary value problems of ordinal differential equations. The method is tested by the help of four numerical examples, for which the exact solution is known. To make the article more readable in Section 2 a short description on the Haar wavelets is added.

E-mail address: ulo.lepik@ut.ee

0096-3003/$ - see front matter
doi:10.1016/j.amc.2005.09.021

2. Haar wavelet basis

The Haar wavelet family is

$$h_i(t) = \begin{cases} 1 & \text{for } t \in [t^{(1)}, t^{(2)}), \\ -1 & \text{for } t \in [t^{(2)}, t^{(3)}), \\ 0 & \text{elsewhere.} \end{cases} \tag{2.1}$$

Here the notations

$$t^{(1)} = \frac{k}{m}, \quad t^{(2)} = \frac{k+0.5}{m}, \quad t^{(3)} = \frac{k+1}{m} \tag{2.2}$$

are introduced. The integer $m = 2^j$, $j = 0, 1, \ldots, J$ indicates the level of the wavelet; $k = 0, 1, \ldots, m-1$ is the translation parameter. The integer J determines the maximal level of resolution. The index i is calculated from the formula $i = m + k + 1$. Here the minimal value is $i = 2$ (then $m = 1, k = 0$); maximal value is $i = 2M$ where $M = 2^J$. The index $i = 1$ corresponds to the scaling function

$$h_1(t) = \begin{cases} 1 & \text{for } 0 \leqslant t \leqslant 1, \\ 0 & \text{elsewhere.} \end{cases} \tag{2.3}$$

We shall divide the interval $t \in [0, 1]$ into $2M$ parts of equal length $\Delta t = 1/(2M)$; the grid points are

$$\tau_l = (l-1)\Delta t, \quad l = 1, 2, \ldots, 2M+1. \tag{2.4}$$

Since in the subsequent sections the collocation method is used then we define also the collocation points

$$t_l = (l-0.5)\Delta t, \quad l = 1, 2, \ldots, 2M. \tag{2.5}$$

Following Chen and Hsiao [4] the Haar coefficient matrix H is introduced; it is a $2M \times 2M$ matrix with the elements $H(i, l) = h_i(t_l)$. Let us integrate (2.1)

$$q_i = \int_0^t h_i(t)\,\mathrm{d}t. \tag{2.6}$$

In the collocation points (2.6) gets the form $Q(i, l) = q_i(t_l)$, where Q is a $2M \times 2M$ matrix. Chen and Hsiao [4] presented this matrix in the form $Q = PH$, where PH is interpreted as the product of the matrices P and H. Chen and Hsiao called P the operational matrix of integration; they shoved that for calculating this matrix of order μ the following matrix equation holds:

$$P_\mu = \begin{bmatrix} P_{0,5\mu} & -(\frac{1}{2}\mu)H_{0,5\mu} \\ (\frac{1}{2}\mu)H^{-1}_{0,5\mu} & 0 \end{bmatrix}. \tag{2.7}$$

It should be noted that calculations for H_μ, P_μ must be carried out only once; the obtained results are applicable for solving whatever problems.

3. Solving nonlinear integral equations by the Haar wavelet method

Consider integro-differential equation of the type

$$\alpha u'(x) + \beta u(x) = \int_0^x K(x, t, u(t), u'(t))\,\mathrm{d}t + f(x) \tag{3.1}$$

for $0 \leqslant x \leqslant 1$ and $u'(0) = u_0$. Here α, β are constants, the functions K, f are prescribed.

Satisfying (3.1) in the collocation points (2.5) we obtain

$$\alpha u'(x_l) + \beta u(x_l) = \int_0^{x_l} K(x_l, t, u(t), u'(t))\,\mathrm{d}t + f(x_l), \tag{3.2}$$

where $l = 1, 2, \ldots, 2M$.

The function $u'(t)$ is developed into the Haar series

$$u'(t) = \sum_{i=1}^{2M} a_i h_i(t). \tag{3.3}$$

By integrating (3.3) we obtain

$$u(t) = \sum_{i=1}^{2M} a_i q_i(t) + u(0). \tag{3.4}$$

If (3.3) and (3.4) are substituted into (3.2) and the integrations are carried out a system of $2M$ equations a_i is obtained for calculating the wavelet coefficients. Since this system is nonlinear we must use some numerical approach. In the present article for this the Newton method is applied, which brings us to the equation

$$\begin{aligned}\sum_{p=1}^{2M}\left[\alpha h_p(x_l) + \beta q_p(x_l) - \int_0^{x_l}\frac{\partial K}{\partial a_p}\,\mathrm{d}t\right]\Delta a_p \\ = -\alpha u'(x_l) - \beta u(x_l) + \int_0^{x_l} K\,\mathrm{d}t + f(x_l), \quad l = 1,2,\ldots,2M,\end{aligned} \tag{3.5}$$

where

$$\frac{\partial K}{\partial a_p} = \frac{\partial K}{\partial u}\frac{\partial u}{\partial a_p} + \frac{\partial K}{\partial u'}\frac{\partial u'}{\partial a_p} = \frac{\partial K}{\partial u}q_p(t) + \frac{\partial K}{\partial u'}h_p(t). \tag{3.6}$$

The main problem is to evaluate the integrals in (3.5). This can be done in the following way. Let us denote

$$\varphi(l) = \int_0^{x_l} K\,\mathrm{d}t, \quad \psi(p,l) = \int_0^{x_l}\frac{\partial K}{\partial a_p}\,\mathrm{d}t \tag{3.7}$$

and consider the subinterval $t \in [\tau_s, \tau_{s+1}]$, $s = 1,2,\ldots,2M$, where τ_s is the sth grid point defined by (2.4). In each subinterval

$$\begin{aligned} h_p(t) &= h_p(t_s) = \text{const.}, \\ q_p(t) &= q_p(\tau_s) + (t-\tau_s)h_p(t_s). \end{aligned} \tag{3.8}$$

Here t_s denotes the sth collocation point defined by (2.5) and

$$t_s = \tau_s + 0.5\Delta t. \tag{3.9}$$

Since $u'(t) = u'(t_s) = \text{const.}$, we obtain

$$\begin{aligned} u(t) &= u(\tau_s) + (t-\tau_s)u'(t_s), \\ u(t_s) &= u(\tau_s) + 0.5u'(t_s)\Delta t. \end{aligned} \tag{3.10}$$

Next the following notations are introduced:

$$\begin{aligned} \Delta G(x_l, t_s, u(t_s), u'(t_s)) &= \int_{\tau_s}^{\tau_{s+1}} K(x_l, t, u(t), u'(t_s))\,\mathrm{d}t, \\ \Delta\widetilde{G}(x_l, t_l, u(t_l), u'(t_l)) &= \int_{\tau_l}^{t_l} K(x_l, t, u(t), u'(t_l))\,\mathrm{d}t, \\ \Delta G_u &= \int_{\tau_s}^{\tau_{s+1}}\frac{\partial K}{\partial a_p}\,\mathrm{d}t = \int_{\tau_s}^{\tau_{s+1}}\left[\frac{\partial K}{\partial u}q_p(t) + \frac{\partial K}{\partial u'}h_p(t)\right]\mathrm{d}t, \\ \Delta\widetilde{G}_u &= \int_{\tau_l}^{t_l}\frac{\partial K}{\partial a_p}\,\mathrm{d}t = \int_{\tau_l}^{t_l}\left[\frac{\partial K}{\partial u}q_p(t) + \frac{\partial K}{\partial u'}h_p(t)\right]\mathrm{d}t. \end{aligned} \tag{3.11}$$

Evaluating the integrals (3.7) for each subinterval $t \in [\tau_s, \tau_{s+1}]$ and summing up the results we obtain

$$\varphi(l) = \sum_{s=1}^{l-1} \Delta G + \Delta \widetilde{G},$$
$$\psi(p,l) = \sum_{s=1}^{l-1} \Delta G_u + \Delta \widetilde{G}_u. \tag{3.12}$$

It is convenient to put our results into the matrix form. For this purpose we introduce the row vectors $x = [x(l)]$, $a = [a_i]$, $\Delta a = [\Delta a_i]$, $u = [u(t_l)]$, $u' = [u'(t_l)]$, $f = [f(x_l)]$, $\varphi = [\varphi(l)]$ and $2M \times 2M$ matrices $H = [h_p(x_l)]$, $Q = [g_p(x_l)]$. Besides we denote

$$S = \alpha H + \beta Q - \psi,$$
$$F = -\alpha u' - \beta u + \varphi + f. \tag{3.13}$$

The matrix form of the system (3.5) is now

$$\Delta a S = F, \tag{3.14}$$

which has the solution

$$\Delta a = F S^{-1}. \tag{3.15}$$

The Volterra integral equation

$$u(x) = \int_0^x K(x,t,u(t))\,\mathrm{d}t + f(x) \tag{3.16}$$

is a special case of (3.5). Our results remain applicable also for the Volterra equation if out the following changes are carried out:

(i) $\alpha = 0$, $\beta = 1$, $\frac{\partial K}{\partial u'} = 0$;
(ii) Eqs. (3.3) and (3.4) are replaced with $u = aH$.

4. Calculation of the wavelet coefficients

The wavelet coefficients a_i can be calculated in the following way. Let us assume that the problem is solved for the level $v = J - 1$, to which correspond $M = 2^J$ collocation points. The wavelet coefficients for this approximation are labeled as $a_i^{(v)}, i = 1, 2, \ldots, M$. Next the value of J is increased by one, thus the number of collocation points is doubled. The wavelet coefficients at the new level are estimated as

$$\hat{a}_i^{(v+1)} = \begin{cases} a_i^{(v)} & \text{for } i = 1, 2, \ldots, M, \\ 0 & \text{for } i = M+1, \ldots, 2M. \end{cases} \tag{4.1}$$

The estimates $\hat{u}^{(v+1)}, (\hat{u}')^{(v+1)}$ are calculated according to (3.3) and (3.4). The values $\hat{a}_i^{(v+1)}$ are corrected by (3.15) and we obtain

$$a_i^{(v+1)} = \hat{a}_i^{(v+1)} + \Delta a_i^{(v+1)}, \quad i = 1, 2, \ldots, 2M. \tag{4.2}$$

The corrected values for $u^{(v+1)}, (u')^{(v+1)}$ are calculated again from (3.3) and (3.4). This cycle is repeated until the necessary exactness of the results is obtained. Assumption (4.1) is motivated by the fact that higher coefficients of the sequence a_i are usually quite small.

The question how to start this procedure arises. We recommend to take the starting solution in the form $(u')^{(0)} = a_0$, $u(0) = a_0 x + u(0)$ and satisfy (3.1) in the collocation point $x = 0.5$. This leads to the equation

$$\alpha a_0 + \beta(0.5 a_0 + u(0)) = \int_0^{0.5} K(0.5, t, a_0 t + u(0), a_0)\,\mathrm{d}t + f(0.5) \tag{4.3}$$

from which the coefficient a_0 can be evaluated.

Estimates for the next step are $\hat{a}_1^{(1)} = a_0,\ \hat{a}_2^{(1)} = 0$ and

$$\begin{aligned} \hat{u}^{(1)}(t) &= \hat{a}_1^{(1)} h_1(t) + \hat{a}_2^{(1)} h_2(t) = a_0, \\ \hat{u}^{(1)}(t) &= \hat{a}_1^{(1)} q_1(t) + \hat{a}_2^{(1)} q_2(t) + u(0) = a_0 t + u(0). \end{aligned} \tag{4.4}$$

These estimates are corrected by solving (3.15) for $M = 1$.

Our calculations show that this simple method gives good results in many cases, but there may be problems for which the sequence of iterations does not converge (the convergence of the Newton method depends upon the successful choice of the initial approximation). In such cases for getting a convergent solution we could solve (3.1) directly for two or four collocation points. The other possibility is to apply the Newton method for a fixed number of collocation points more than once and only after that double the number of collocation points.

5. Error estimates

With the purpose to demonstrate the applicability and efficiency of the proposed method in the following sections some tutorial examples, for which the exact solution is known, are solved. The error of the vth iteration can be estimated as

$$\varepsilon(v) = \max_{1 \leqslant i \leqslant 2M} |u^{(v)}(x_i) - u_{ex}(x_i)|, \tag{5.1}$$

where $u_{ex}(x)$ is the exact solution. For determining the convergence rate of the solution the quantity

$$\rho(v) = \frac{\varepsilon(v-1)}{\varepsilon(v)}, \quad v = 2, 3, 4, \ldots \tag{5.2}$$

is introduced.

In most cases we do not know the exact solution. Here the error of the results can be estimated in the following way. We introduce the quantity

$$S(v) = \Delta x^{(v)} \cdot \sum_{i=1}^{2M} \left| u^{(v)}(x_i) \right| \tag{5.3}$$

where $\Delta x^{(v)}$ denotes the stepsize of the vth approximation. Since $u^{(v)}(x)$ is in each subinterval $x \in [x_i, x_{i+1}]$ a linear function then $S(v)$ has a distinct geometrical meaning: it is the area which lies in the interval $x \in [0, 1]$ underneath the curve $|u^{(v)}(x)|$. For estimating the exactness of the solution the quantity

$$\Delta(v) = \left| \frac{S(v+1)}{S(v)} - 1 \right|, \tag{5.4}$$

is introduced. Geometrically $\Delta(v)$ denotes the relative increase of the area underneath the curve $|u^{(v)}(x)|$, $x \in [0, 1]$ for one iteration step.

The convergence rate of the process can be estimated with the aid of the function

$$\sigma(v) = \frac{\Delta(v-1)}{\Delta(v)}, \quad v = 2, 3, \ldots$$

According to the criterion (5.1) the error estimate is based on a single collocation point, therefore it is a local criterion of convergence. The estimate (5.4) makes use of all collocation points therefore it is a integral criterion.

6. Examples

Let us consider some test problems.

Example 6.1. Consider the Volterra equation

$$u(x) = \int_0^x \frac{1+u^2(t)}{1+t^2}, \tag{6.1}$$

which has the exact solution $u_{ex} = x$.

The wavelet solution is sought in the form

$$u(t) = \sum_{i=1}^{2M} a_i h_i(t). \tag{6.2}$$

Since $K = \frac{1+u^2}{1+t^2}$, it follows from (3.11) and (3.12) that:

$$\begin{aligned} \varphi(l) &= \sum_{s=1}^{l-1}[1+u^2(t_s)A(s)] + 1 + u^2(t_l)a(l), \\ \psi(p,l) &= 2\sum_{s=1}^{l-1} u(t_s)h_p(t_s)A(s) + 2[1+u(t_l)h_p(t_l)]A(l), \quad p,l = 1,2,\ldots,2M, \end{aligned} \tag{6.3}$$

where

$$\begin{aligned} A(s) &= \arctan\tau_{s+1} - \arctan\tau_s, \\ A(l) &= \arctan t_l - \arctan\tau_l. \end{aligned}$$

We start our solution with $u = a_0 = \text{const}$. Satisfying (4.3) we get the equation

$$a_0^2 - \frac{a_0}{\arctan 0.5} + 1 = 0, \tag{6.4}$$

which has two roots $a_0^{(1)} = 0.6747$ and $a_0^{(2)} = 1.4821$. It follows from the calculations that the value $a_0^{(2)}$ brings to a nonconvergent iteration process and therefore we shall take $a_0 = 0.6747$. Next two collocation points $x_1 = 0.25$ and $x_2 = 0.75$ are taken and the estimates for a and u are $\hat{a} = (a_0, 0), \hat{u} = (a_0, a_0)$. Correcting these values with the aid of (3.15) we obtain $a^{(1)} = (0.523, -0.323)$ and $u^{(1)} = (0.199, 0.846)$; the error function (5.1) is $\varepsilon^{(1)} = 0.096$. Results of the following iterations are shown in Table 1.

Example 6.2. Solve the Volterra equation of second kind

$$u(x) = \frac{1}{2}\int_0^x u(t)u(x-t)\,\mathrm{d}t + \frac{1}{2}\sin x, \quad 0 < x < 1. \tag{6.5}$$

The exact solution of (6.5) is $u(x) = J_1(x)$, where symbol J_1 denotes the Bessel function of order 1.

In the present case

$$\begin{aligned} K &= -\frac{1}{2}u(t)u(x-t), \\ \frac{\partial K}{\partial a_p} &= \frac{1}{2}[u(x-t)h_p(x-t) - u(t)h_p(t)]. \end{aligned} \tag{6.6}$$

Table 1
Error estimates ε, Δ and convergence rates ρ, σ for Eq. (6.1)

J	$2M$	ε	ρ	Δ	σ
1	4	2.7E−2	–	7.7E−2	–
2	8	1.6E−3	15.0	2.8E−2	2.7
3	16	4.7E−4	3.7	8.6E−4	32.5
4	32	1.3E−4	3.7	2.9E−4	2.9
5	64	3.3E−5	3.9	7.9E−5	3.7
6	128	8.4E−6	3.9	2.0E−5	3.9

Table 2
Error estimates ε, Δ and convergence rates ρ, σ for Eq. (6.5)

J	$2M$	ε	ρ	Δ	σ
1	4	1.2E−2	–	2.2E−2	–
2	8	1.2E−3	9.2	3.5E−2	0.6
3	16	1.3E−3	1.0	6.8E−3	51.2
4	32	7.9E−4	1.6	7.0E−4	1.0
5	64	4.3E−4	1.8	5.1E−4	1.3
6	128	2.2E−4	1.9	3.0E−4	1.7

Making use of (3.11) and (3.12) we obtain

$$\varphi(l) = \frac{\Delta t}{2}\sum_{s=1}^{l-1} u(t_s)u(x_l - t_s),$$
$$\psi(p,l) = \frac{\Delta t}{2}\sum_{s=1}^{l-1}[u(t_s)h_p(t_s) - u(x_l - t_s)h_p(x_l - t_s)]. \tag{6.7}$$

Eq. (3.14) get the form $S = H$, $F = -u' + \varphi + f$, where $f(l) = 0.5\sin x_l$.

The starting solution is taken again in the form $u = a_0$. Satisfying (4.3) we get for a_0 the quadratic equation

$$a_0^2 - 4a_0 + 2\sin 0.5 = 0,$$

which has the solutions $a_0^{(1)} = 0.256, a_0^{(2)} = 3.744$. Since the value $a_0^{(2)}$ leads to a nonconverging iteration process we take $a_0 = 0.256$. The estimated solution for two collocation points is $\hat{a}_1 = (a_0, 0), \hat{u} = (a_0, a_0)$. Correcting it with the aid of (3.15) we obtain $a = (0.240, -0.117)$, $u = (0.1237, 0.3572)$ with the error $\varepsilon_0 = 0.0021$. Error estimates of the subsequent approximations are shown in Table 2.

It follows from this table that the convergence rates are smaller as in the case of Example 6.1.

Example 6.3. Consider the integro-differential equation

$$u'(x) = 1 + \int_0^x u(t)u'(t)\,\mathrm{d}t, \quad 0 \leqslant x \leqslant 1; \quad u(0) = 0. \tag{6.8}$$

The exact solution of (6.8) is

$$u_{ex}(x) = \sqrt{2}\tan\frac{x}{\sqrt{2}}. \tag{6.9}$$

We shall seek the wavelet solution in the form (3.3) and (3.4); from (3.11) and (3.12) we find

$$\varphi(l) = \Delta t\sum_{s=1}^{l-1} u(t_s)u'(t_s) + \frac{\Delta t}{2}u(t_l)u'(t_l),$$
$$\psi(p,l) = \Delta t\sum_{s=1}^{l-1}[u(t_s)h_p(t_s) + u'(t_s)q_p(t_s)] + \frac{\Delta t}{2}[u(t_l)h_p(t_l) + u'(t_l)q_k(t_l)]. \tag{6.10}$$

Eqs. (3.13) and (3.14) obtain the form $\Delta a S = F$, where $S = H - \psi$, $F = -u' + \varphi + E$ (symbol E denotes the $2M$ dimensional unit vector).

For starting we take $u' = a_0$, $u = a_0 x$. Satisfying (4.3) for $x = 0.5$ the equation $a_0^2 - 8a_0 + 8 = 0$ is obtained. It has two roots from which we shall take $a_0 = 1.172$ (the other root leads to a nonconverging process). Correcting the estimates $\hat{a} = (a_0, 0)$, $\hat{u}' = (a_0, a_0)$, $\hat{u} = 0.25(a_0, 3a_0)$ with the aid of (3.15) we find $a = (1.237, -0.190)$, $u' = (1.047, 1.426)$, $u = (0.262, 0.880)$ the error of this approximation is $\varepsilon = 0.051$. Results for the next approximations are presented in Table 3.

This example is taken from the article [5] by Avudainayagam and Vani, who solved it by the wavelet-Galerkin method. For computing the integrals the connection coefficient method was used. This makes the solution quite complicated, besides it is applicable only in the case of quadratic nonlinearities of the type u^2,

Table 3
Error estimates ε, Δ and convergence rates ρ, σ for Eq. (6.8)

J	$2M$	ε	ρ	Δ	σ
1	4	4.2E−3	–	4.0E−2	–
2	8	3.3E−3	1.3	1.1E−3	37.0
3	16	1.1E−3	3.0	1.6E−3	0.7
4	32	3.0E−4	3.7	4.3E−4	3.7
5	64	8.9E−5	3.3	1.2E−4	3.5
6	128	2.6E−5	3.4	3.5E−5	3.4

uu' or u'^2. The same example was solved also in [6] with the aid of the Adomian's decomposition method. The authors of [6] denote that their solution is more simple and easy to use. We would like to note that its accuracy of the results may be insufficient. So it follows from Table 3 of the paper [6] that $u = 1.0973$ for $x = 1$ while the exact value is $u = 1.2085$; consequently, the error is 9.2%.

7. Boundary value problems of ODE

The method of solution proposed in this article can be applied also for solving boundary value problems of ordinary differential equations. To illustrate this let us solve the two-point boundary value problem for the differential equation

$$u'' = K(x, u, u') \tag{7.1}$$

with the boundary conditions $u(0) = A$, $u(1) = B$.

By integrating (7.1) we get the integro-differential equation

$$u'(x) = \int_0^x K(t, u(t), u'(t))\,\mathrm{d}t + u'(0), \quad 0 \leqslant x \leqslant 1. \tag{7.2}$$

As before we seek its solution in the form (3.3) and (3.4). Since

$$q_i(1) = \begin{cases} 1 & \text{for } i = 1, \\ 0 & \text{for } i > 1, \end{cases} \quad (i = 1, \ldots, 2M) \tag{7.3}$$

it follows from (3.4) that $a_1 = B - A$, consequently, the first coefficient in (3.4) is specified.

According to (3.3)

$$u'(0) = \sum_{i=1}^{2M} a_i h_i(0), \tag{7.4}$$

where

$$h_i(0) = \begin{cases} 1 & \text{if } i = 1, \\ 1 & \text{if } i = 2^J + 1, \quad J = 0, 1, 2, \ldots \\ 0 & \text{elsewhere.} \end{cases}$$

Due to the fact that a_1 is fixed and $\Delta a_1 = 0$ some changes in the system of Eqs. (3.14) must be executed. Let the symbol $\tilde{a}$ denote the vector a for which the first component is deleted. In similar way the $\widehat{S}$ is a reduced matrix in which the first row and column are deleted. Instead of (3.13)–(3.15) we get now

$$\Delta\tilde{a}\widetilde{S} = \widetilde{F}, \tag{7.5}$$

where

$$\widetilde{S} = \widetilde{H} - \tilde{\psi} - \widetilde{E} \otimes \tilde{h}(0)^{\mathrm{T}},$$
$$\widetilde{F} = -\tilde{u}' + \tilde{u}'(0) + \tilde{\varphi} + \tilde{f}.$$

Here $\widetilde{E}$ is a $2M - 1$ dimensional unit vector, and $\otimes$- denotes the Kronecker tensor product.

The following solution proceeds according to the algorithm presented in Sections 3 and 4: we generate the vector $a = (B - A, a_2, \ldots, a_{2M})$ and evaluate $u(x)$, $u'(x)$ for the next approximation from (3.3) and (3.4).

Example 7.1. Consider the boundary value problem

$$u'' - uu' = \frac{1}{4\sqrt{(1+x)^3}} - \frac{1}{2}, \quad u(0) = 1, \quad u(1) = \sqrt{2}, \tag{7.6}$$

which has the exact solution

$$u = \sqrt{1+x}.$$

By integrating (3.4) we obtain

$$u'(x) = \int_0^x uu' \,\mathrm{d}t + u'(0) - \frac{1}{2}(1+x) + \frac{1}{2\sqrt{1+x}}. \tag{7.7}$$

It follows from boundary conditions that $a_1 = u(1) - u(0) = \sqrt{2} - 1$.

We start our solution with evaluating the quantities $\varphi(l)$ and $\psi(p, l)$. Making use of (3.11) and (3.12) we obtain

$$\varphi(l) = \Delta t \sum_{s=1}^{l-1} u(t_s)u'(t_s) + \frac{\Delta t}{2}u'(t_l)\left[u(t_l) - \frac{\Delta t}{4}u'(t_l)\right], \tag{7.8}$$

$$\psi(p, l) = \Delta t \sum_{s=1}^{l-1}[h_p(t_s)u(t_s) + q_p(t_s)u'(t_s)] + \frac{\Delta t}{2}(h_p(t_l)u(t_l) + q_p(t_l)u'(t_l)) - \left(\frac{\Delta t}{2}\right)^2 h_p(t_l)u'(t_l). \tag{7.9}$$

Since now a_1 is fixed we take the starting solution in the form

$$\begin{aligned} u'(t) &= a_1 + a_2 h_2(t), \\ u(t) &= a_1 t + a_2 q_2(t) + u(0). \end{aligned} \tag{7.10}$$

Replacing (7.10) into (7.7) and satisfying this equation in the point $x = 0.5$ we get the quadratic equation

$$a_2^2 + 2(5 + \sqrt{2})a_2 + 2\sqrt{2} + \frac{8}{\sqrt{6}} - 7 = 0.$$

This equation has two roots from which fits to us the root $a_2 = 0.0702$. The estimates for the first approximation are $\hat{a} = (a_1, a_2) = (0.4142, 0.0702)$, $\hat{u}' = (0.4943, 0.3539)$, $\hat{u} = (1.124, 1.336)$. Correcting the values with the aid of (7.5), (7.8) and (7.9) we find $a^{(1)} = (0.424, 0.056)$, $u^{(1)} = (1.120, 1.332)$, $u'^{(1)} = (0.480, 0.368)$ with the error $\varepsilon^{(0)} = 0.009$. Results of the subsequent approximations are presented in Table 4.

Table 4
Error estimates ε, Δ and convergence rates ρ, σ for Eq. (7.7)

J	$2M$	ε	ρ	Δ	σ
1	4	3.2E−3	–	6.5E−2	–
2	8	1.1E−3	2.9	1.0E−3	62.4
3	16	3.2E−4	3.5	3.9E−4	2.7
4	32	8.6E−5	3.7	1.2E−4	3.4
5	64	2.2E−5	3.9	3.1E−5	3.7
6	128	5.6E−6	3.9	8.2E−6	3.8

8. Conclusions

A new method for numerical solution of Volterra integral-equations and integro-differential equations, which is based on the Haar wavelets, is proposed. Its applicability and efficiency is checked on four test problems.

The benefits of the Haar wavelet approach are sparse matrices of representation, fast transformation and possibility of implementation of fast algorithms. Simplicity of our solution is due to a great extent to the assumption (4.1), based on the fact that usually higher wavelet coefficients a_i are small. It should be mentioned that such an approach is not applicable in the case of the conventional piecewise constant approximations method.

It follows from Tables 1–4 that the accuracy of the obtained solutions is quite high even if the number of collocation points is small (as a rule a four or eight point solution guarantees satisfactory exactness). By increasing the number of collocation points the error of the solution rapidly decreases.

Approximation with the Haar wavelets is equivalent with the approximation for piecewise constant functions. Therefore the convergence rate for piecewise constant functions, which is $O(M^{-2})$ can be transferred to Haar wavelet approach. As to the Newton method then in the case of sufficiently good initial values it has also quadratic convergence. So it could be expected that in the case of our solution by doubling the number of collocation points the error function roughly decreases four times. This theoretical estimation is in general consistent with the data in Tables 1, 3 and 4. Deviations appear only for small values of $J \leqslant 3$, where due to small number of calculation points some "adaption" takes place. For bigger values of J the convergence rates ρ and σ come nearer to the theoretical value 4. Exceptional is Example 6.2, where convergence rates are considerably smaller.

Acknowledgement

Financial support from the Estonian Science Foundation under Grant ETF-5240 is gratefully acknowledged.

References

[1] H. Brunner, Collocation method for Volterra integral and related functional equations, Cambridge Monograph on Applied and Computational Mathematics, Cambridge University Press, Cambridge, MA, 2004.
[2] Ü. Lepik, E. Tamme, Application of the Haar wavelets for solution of linear integral equations, in: Dynamical Systems and Applications, Antalaya, 2004, Proceedings, 2005, pp. 395–407.
[3] Ü. Lepik, E. Tamme, Solution of nonlinear integral equations via Haar wavelet method, Int. J. Wavelets, Multiresol. Inform. Process., submitted for publication.
[4] C.F. Chen, C.H. Hsiao, Haar wavelet method for solving lumped and distributed-parameter systems, IEE Proc. Control Theory Appl. 144 (1997) 87–94.
[5] A. Avudainayagam, C. Vani, Wavelet-Galerkin method for integro-differential equations, Appl. Numer. Math. 32 (2000) 247–254.
[6] S.M. El-Sayed, M.R. Abdel-Azis, A comparison of Adomian's decomposition method and wavelet-Galerkin method for solving integro-differential equations, Appl. Math. Comput. 136 (2003) 151–159.