A Numerical Approach for Solving Linear and Nonlinear Volterra Integral Equations with Controlled Error

A. Vahidian Kamyad, M. Mehrabinezhad, J. Saberi-Nadjafi

***Abstract*—In this paper we propose a new approach for solving linear and nonlinear Volterra integral equations (LVIE, NVIE) of the first and the second kinds. First, we define a new problem in calculus of variations, which is equivalent to this kind of problem. By taking this approach, one can solve a large number of problems in calculus of variations. We use a discretisation method to obtain a nonlinear programming problem and in some cases a linear programming problem. Then by using the optimal solution of the latest (LP or NLP) problem, we obtain an approximate solution with a controllable error for the original solution.**

***Keywords*—Volterra integral equations, Discretisation, Nonlinear programming.**

I. Introduction

Finding the exact solution of the integral equations by classical methods is sometimes too difficult, and it is usually very useful to find a numerical estimation of the exact solution.

Consider the following Volterra integral equation of the second kind:

$$u(x)=f(x)+\lambda\int_0^x k(x,t)\varphi(u(t))dt, \tag{1}$$

where $k(.,.)$:$\boldsymbol{R^2}\to\boldsymbol{R}$ is the kernel which is a known function, and λ is a given real parameter and $f(.)$:$\boldsymbol{R}\to\boldsymbol{R}$ is a given function. We are trying to find an approximate solution of equation (1) where $u(.)$:$\boldsymbol{R}\to\boldsymbol{R}$. $\varphi(u(t))$ is a linear or nonlinear function of $u(t)$. Many different techniques have been presented so far for solving (1) such as Adomian's decomposition method, series solution method and successive substitution method [6],[9]. In recent years many numerical methods are also presented for solving VIEs [2],[5],[7],[8]. Homotopy perturbation method is applied in [1], [4] to solve such problems.

A.V. Kamyad is with the department of Applied Mathematics, Ferdowsi University of Mashhad, Mashhad, Iran. (e-mail: Kamyad@math.um.ac.ir).

M. Mehrabinezhad is a Ph.D. student in Applied Math. at Ferdowsi University of Mashhad, Mashhad, Iran. (corresponding author. Phone and fax: +98 (0) 511 8828606; e-mail: mmehrabinezhad@gmail.com, mehrabinezhad@stu-mail.um.ac.ir).

J. Saberi-Nadjafi is with the department of Applied Mathematics, Ferdowsi University of Mashhad, Mashhad, Iran. (e-mail: najafi141@gmail.com).

In this paper we propose a new numerical approach for solving the above equation, both linear and nonlinear by discretisation and using an interpolation method to find a formula for solution of such integral equation.

The organization of this paper is as follows: in Section II some theorems are presented that will be used in later sections. Our algorithm is illustrated In Section III. In Section IV some examples are provided and the results are compared with the exact solutions. Section V is the conclusion.

II. Preliminaries. Consider the following theorem

Theorem 2.1. *If f(x,t) be a given function and a and b are constants, and let* $\{t_1,t_2,\ldots,t_n\}$ *be a set of support points in* $[a,b]$, *where* $a=t_1<t_2<\ldots<t_n=b$, *then*

$$\int_a^b f(x,t)dt=\lim_{n\to\infty}\sum_{i=1}^{i=n-1} f(x,\xi_i)\Delta t_i, \tag{2}$$

Proof. As we know from calculus

$$\int_a^b f(x,t)dt=\lim_{n\to\infty}\sum_{i=1}^{i=n-1}\int_{t_i}^{t_{i+1}} f(x,t)dt, \tag{3}$$

where $\Delta t_i=t_{i+1}-t_i$ and ξ_i are arbitrary points in the interval $[t_i,t_{i+1}]$, $(i=1,2,\ldots,n-1)$, notice that the subintervals are of equal length. The right-hand components of (3) contain integrals with small intervals (when n→∞ t_i's get close to each other), so we can substitute each integral by

$$\int_{t_i}^{t_{i+1}} f(x,t)dt=f(x,\xi_i)\Delta t_i, \tag{4}$$

where ξ_i $(i=1,2,\ldots,n-1)$ are arbitrary points in the interval $[t_i,t_{i+1}]$. Now we use (4) in the right-hand side of (3) to obtain

$$\lim_{n\to\infty}\sum_{i=1}^{i=n-1}\int_{t_i}^{t_{i+1}} f(x,t)dt=\lim_{n\to\infty}\sum_{i=1}^{i=n-1} f(x,\xi_i)\Delta t_i, \tag{5}$$

By considering (3) and (5) together, the proof is completed, and we have

$$\int_a^b f(x,t)dt=\lim_{n\to\infty}\sum_{i=1}^{i=n-1}\int_{t_i}^{t_{i+1}} f(x,t)dt=\lim_{n\to\infty}\sum_{i=1}^{i=n-1} f(x,\xi_i)\Delta t_i. \blacksquare$$

(Advance online publication: 13 May 2010)

Remark 1: If we choose the same distance between support points, from (3) we obtain the following formula:

$$\int_a^b f(x,t)dt = \lim_{n\to\infty}\sum_{i=0}^{i=n-1}\int_{a+ih}^{a+(i+1)h} f(x,t)dt$$
$$= \lim_{n\to\infty}\sum_{i=0}^{i=n-1} f(x,\xi_i)h,$$

where $h=\dfrac{b-a}{n}$ and ξ_i (i=0,…,n-1) are arbitrary points in the interval $[t_i, t_{i+1}]$.

The next two theorems are about some properties of convex functions.

Theorem 2.2. *If y=f(x) be a convex function on a convex set, then any local minimum of f is a global one.*

Proof. See [3]

Theorem 2.3. *Consider n convex vector functions* $f_1, f_2, \ldots, f_n$*, then* $g(x)=\sum_{i=1}^{n}\alpha_i f_i(x)$ *is also a convex function, for* $\alpha_i \geq 0$*,* $(i=1,2,\ldots,n)$.

Proof. It is easy to see.

III. A new approach for solving NVIE and LVIE

In this section we propose our method to find the numerical solution of nonlinear Volterra integral equation of the form

$$u(x) = f(x) + \lambda\int_0^x k(x,t)\varphi(u(t))dt,\quad 0\leq x\leq 1, \quad (6)$$

where $f(x)$, $k(x,t)$ are given functions and λ is a given parameter and f is a continuous function.

The basis of our method exclusively focuses upon discretisation. We can rewrite (6) as follows:

$$u(x) - f(x) - \lambda\int_0^x k(x,t)\varphi(u(t))dt = 0,\ 0\leq x\leq 1. \quad (7)$$

(Note that we are choosing $0\leq x\leq 1$, since every interval such as $[a,b]$ can be transformed into this interval by a linear transformation.) Let

$$E_u(x) = u(x) - f(x) - \lambda\int_0^x k(x,t)\varphi(u(t))dt,$$

where E_u(x) (an error function) is a functional and depends on the unknown function $u(x)$, so E_u:Pc[0,1]→***R*** (where Pc[0,1] is the set of all piecewise continuous functions on the interval [0,1]). To solve (7), consider the following problem:

$$\underset{u\in Pc[0,1]}{Min}\int_X |E_u(x)|dx \quad ; \quad 0\leq x\leq 1, \quad (8)$$

where $X=[0,1]$, we can even solve the following problem:

$$\underset{u\in Pc[0,1]}{Min}\|E_u\|_p^p \quad ; \quad 0\leq x\leq 1 \quad ,p\geq 1, \quad (9)$$

where $\|E_u\|_p = \left(\int_X |E_u|^p\, d\mu\right)^{\frac{1}{p}}$.

The following theorem is of great importance and is the basis of our method.

Theorem 3.1. *Let $u(x)$ be a continuous function on* $[0,1]$ *and a solution for (7), then $u(x)$ is the optimal solution of (8) with zero objective function and vice versa.*

Proof. Let $u_1(x)$ be a solution for (7), which is continuous on [0,1], so

$$u_1(x) - f(x) - \lambda\int_0^x k(x,t)\varphi(u_1(t))dt = 0,\quad 0\leq x\leq 1.$$

Hence,

$$\left|u_1(x) - f(x) - \lambda\int_0^x k(x,t)\varphi(u_1(t))dt\right| = 0.$$

Since u_1 and f are continuous on their domains, by integrating both sides of the last equation on X=[0,1], we obtain

$$\int_X\left|u_1(x) - f(x) - \lambda\int_0^x k(x,t)\varphi(u_1(t))dt\right|dx = 0.$$

Thus, $\|E_u\|_1 = 0$ and this means that $u_1(x)$ is the optimal solution of (8) with zero objective function.

For the converse part of the proof, we let $u_1(x)$ be the optimal solution of (8) with zero objective function, then

$$\int_x\left|u_1(x) - f(x) - \lambda\int_0^x k(x,t)\varphi(u_1(t))dt\right|dx = 0.$$

Since

$$\left|u_1(x) - f(x) - \lambda\int_0^x k(x,t)\varphi(u_1(t))dt\right|$$

is an absolute function, by using Lebesgue integral theorems, we see that the following equality must be held

$$\left|u_1(x) - f(x) - \lambda\int_0^x k(x,t)\varphi(u_1(t))dt\right| = 0,\ 0\leq x\leq 1,$$

which

$$u_1(x) - f(x) - \lambda\int_0^x k(x,t)\varphi(u_1(t))dt = 0,$$

so $u_1(x)$ is the solution of (7). ■

Note that the same result is held between a solution of (7) and the optimal solution of (9).

Remark 2: It is clear, the optimal solution for (8) is zero, and since zero is a local minimum of (8), by theorem 2.2, it is also a global one. We solve (8) instead of solving (7).

In what follows, we present our numerical method based on discretisation, for solving Eq. (7). By applying the result of theorem 3.1, we solve Eq. (7), by solving an NLP optimization problem (8).

By theorem 2.1 we can substitute (8) by an infinite series

$$\int_0^1 \left| u(x) - f(x) - \lambda \int_0^x k(x,t)\varphi(u(t))dt \right| dx$$

$$= \lim_{n\to\infty} \sum_{i=0}^{i=n-1} \int_{x_i}^{x_{i+1}} \left| u(x) - f(x) - \lambda \int_0^x k(x,t)\varphi(u(t))dt \right| dx$$

$$= \lim_{n\to\infty} \sum_{i=0}^{i=n-1} h \left| u(\xi_i) - f(\xi_i) - \lambda \int_0^x k(\xi_i,t)\varphi(u(t))dt \right|, \tag{10}$$

where ξ_i for $(i = 1, 2, \ldots, n-1)$ are arbitrary points in the interval $[x_i, x_{i+1}]$, $h = \frac{1}{n}$ and $x_i = ih$.

We can choose $\xi_i = x_i$, the lower bound in each interval; thus, (10) changes to the following:

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{i=n-1} \left| u(x_i) - f(x_i) - \lambda \int_0^{x_i} k(x_i,t)\varphi(u(t))dt \right|. \tag{11}$$

If we use the same method as we discussed above to approximate the inner integral in (11), then we have

$$\int_0^{x_i} k(x_i,t)\varphi(u(t))dt = \int_0^{ih} k(x_i,t)\varphi(u(t))dt$$

$$= \int_0^{h} k(x_i,t)\varphi(u(t))dt + \int_h^{2h} k(x_i,t)\varphi(u(t))dt + \ldots$$

$$+ \int_{(i-1)h}^{ih} k(x_i,t)\varphi(u(t))dt = \sum_{j=0}^{j=i-1} \int_{jh}^{(j+1)h} k(x_i,t)\varphi(u(t))dt. \tag{12}$$

Since h is small, we can approximate each integral by

$$\int_{jh}^{(j+1)h} k(x_i,t)u(t)dt = \frac{1}{n} k(x_i,t_j)u(t_j)$$

$$= \frac{1}{n} k(ih, jh)u(jh), \tag{13}$$

where $t_j = jh$. Now from (12) and (13), we can reach the following conclusion:

$$\int_0^{x_i} k(x_i,t)u(t)dt = \frac{1}{n} \sum_{j=0}^{j=i-1} k(x_i,t_j)u(t_j). \tag{14}$$

Finally, from (10),(11) and (14) we have

$$\int_0^1 \left| u(x) - f(x) - \lambda \int_0^x k(x,t)\varphi(u(t))dt \right| dx$$

$$= \lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{i=n-1} \left| u(x_i) - f(x_i) - \lambda \int_0^{x_i} k(x_i,t)\varphi(u(t))dt \right|$$

$$= \lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{i=n-1} \left| u(x_i) - f(x_i) - \lambda h \sum_{j=0}^{j=i-1} k(x_i,t_j)u(t_j) \right| \tag{15}$$

To solve (8), we find $u_i = u(x_i)$ $(i = 1, 2, \ldots, n)$, so that (15) is minimized. We are now dealing with an NLP problem and we can use different software to find a solution for this problem such as *Matlab* or *Mathematica*. By theorem 2.3, (15) is a convex function; hence, if zero is a local minimum of (8), it is also a global minimum of (15).

After finding u_i $(i = 1, 2, \ldots, n)$, we can use an interpolation method to fit a curve to these data and find a relation for $u(x)$.

To make this approach more clear, let us choose n=10, and divide the interval [0,1] into ten equidistance subintervals, so h=0.1, then we have

$$\int_0^1 \left| u(x) - f(x) - \lambda \int_0^x k(x,t)\varphi(u(t))dt \right| dx$$

$$= \int_0^{.1} \left| u(x) - f(x) - \lambda \int_0^x k(x,t)\varphi(u(t))dt \right| dx$$

$$+ \int_{.1}^{.2} \left| u(x) - f(x) - \lambda \int_0^x k(x,t)\varphi(u(t))dt \right| dx$$

$$\int_{.2}^{.3} \left| u(x) - f(x) - \lambda \int_0^x k(x,t)\varphi(u(t))dt \right| dx + \ldots$$

$$+ \int_{.9}^{1} \left| u(x) - f(x) - \lambda \int_0^x k(x,t)\varphi(u(t))dt \right| dx$$

$$= 0.1 \left| u(0) - f(0) - \lambda \int_0^0 k(0,t)\varphi(u(t))dt \right|$$

$$+ 0.1 \left| u(.1) - f(.1) - \lambda \int_0^{.1} k(.1,t)\varphi(u(t))dt \right|$$

$$+ 0.1 \left| u(.2) - f(.2) - \lambda \int_0^{.2} k(.2,t)\varphi(u(t))dt \right| + \ldots$$

$$+ 0.1 \left| u(.9) - f(.9) - \lambda \int_0^{.9} k(.9,t)\varphi(u(t))dt \right|$$

$$= 0.1 \Big[\left| u(0) - f(0) \right| +$$

$$+ \left| u(.1) - f(.1) - 0.1\lambda k(.1,0)\varphi(u(0)) \right|$$

$$+ \Big| u(.2) - f(.2) - \lambda \Big(\int_0^{.1} k(.2,t)\varphi(u(t))dt$$

$$+ \int_{.1}^{.2} k(.2,t)\varphi(u(t))dt \Big) \Big| + \ldots$$

$$+ \Big| u(.9) - f(.9) - \lambda \Big(\int_0^{.1} k(.9,t)\varphi(u(t))dt$$

$$+ \int_{.1}^{.2} k(.9,t)\varphi(u(t))dt + \ldots + \int_{.8}^{.9} k(.9,t)\varphi(u(t))dt \Big) \Big| \Big]$$

$$= 0.1 \Big[\left| u(0) - f(0) \right|$$

$$+ \left| u(.1) - f(.1) - .1\lambda k(.1,0)\varphi(u(0)) \right|$$

$$+ \Big| u(.2) - f(.2) - .1\lambda [k(.2,0)\varphi(u(0))$$

$$+ k(.2,.1)\varphi(u(.1))] \Big| + \ldots$$

$$+ \Big| u(.9) - f(.9) - .1\lambda [k(.9,0)\varphi(u(0))$$

$$+ k(.9,.1)\varphi(u(.1)) + \ldots + k(.9,.8)\varphi(u(.8))] \Big| \Big] \tag{16}$$

Now, by finding the minimum value of (16) and determining approximate values for $u(x_i)$ $(i = 1, 2, \ldots, n)$, and using an interpolation function, we can find a relation for the unknown function $u(x)$.

Remark 3: If in (9), we choose $p>1$, we can change the problem to a nonlinear programming problem and use special software to solve the problem and find more accurate results.

Remark 4: In this method, we can control the accuracy of the results. For example, if we want the total error to be less than a given number ε, it is just needed to add this as a constraint to the problem, and instead of solving (9), solve the following:

$$\underset{u\in Pc[0,1]}{Min}\left\|E_u\right\|_p^p$$
$$\text{s.t}$$
$$\left\|E_u\right\|_p^p<\varepsilon$$

where $0\le x\le 1$, and for any $p\ge 1$.

IV. Some numerical examples of NVIE and LVIE

In this section, we solve some examples by our method, and compare the numerical results with the values of exact solutions.

Example 4.1. Consider the following Volterra integral equation:

$$u(x)=2\cos(x)+3\int_0^x u(t)\sin(t)dt,\quad 0\le x\le 1. \quad (17)$$

We intend to use (9) to solve this problem. Clearly, $u(0)=2.$ The exact solution of this problem is

$$u(x)=(2/3)\left(1+2e^{-3(\cos x-1)}\right).$$

Now, we use our method to solve this equation. Instead of solving (17), we consider the following optimization problem. Let p=2, from (9) we have

$$Min\int_0^1|u(x)-2\cos(x)-3\int_0^x u(t)\sin(t)dt\,|^2\,dx$$
$$\cong Min\left[\int_0^{.01}\alpha^2dx+\int_{.01}^{.02}\alpha^2dx+\cdots+\int_{.99}^{1}\alpha^2dx\right], \quad (18)$$

where $\alpha=u(x)-2\cos(x)-3\int_0^x u(t)\sin(t)dt$.

Here we choose h=.01 and use a finite series instead of an infinite one. Similar to the process which results in (16), we conclude that (18) equals to the following:

$$Min\ .01(u(0)-2\cos(0)-3\int_0^0 u(t)\sin tdt)^2$$
$$+.01(u(.01)-2\cos(.01)-3\int_0^{.01}u(t)\sin tdt)^2$$
$$+.01(u(.02)-2\cos(.02)-3\int_0^{.02}u(t)\sin tdt)^2+\ldots$$
$$+.01(u(.99)-2\cos(.99)-3\int_0^{.99}u(t)\sin tdt)^2.$$

Now by using theorem 2.1 for solving inner integrals, we have

$$Min\ .01[u(0)-2\cos(0)]^2$$
$$+.01[u(.01)-2\cos(.01)-3u(0)\sin 0]^2$$
$$+.01[u(.02)-2\cos(.02)$$
$$-3(\int_0^{.01}u(t)\sin tdt+\int_{.01}^{.02}u(t)\sin tdt)]^2+\ldots$$
$$+.01[u(.99)-2\cos(.99)$$
$$-3(\int_0^{.01}u(t)\sin tdt+\ldots+\int_{.98}^{.99}u(t)\sin tdt)]^2.$$

Finally, we have the following optimization problem to solve

$$Min\ .01[u(0)-2\cos(0)]^2$$
$$+.01[u(.01)-2\cos(.01)]^2$$
$$+.01[u(.02)-2\cos(.02)-.03u(.01)\sin(.01)]^2$$
$$+.01[u(.03)-2\cos(.03)$$
$$-.03(u(.01)\sin(.01)+u(.02)\sin(.02))]^2+\ldots$$
$$+.01[u(.99)-2\cos(.99)$$
$$-.03(u(.01)\sin(.01)+\cdots+u(.98)\sin(.98))]^2. \quad (19)$$

With an initial condition $u(0)$=2, we have solved (19) by *Matlab* software, and obtained data , $u(x_i)$ for $i=1,2,\ldots,100$ are collected in table I. According to these data, the value of (19) is 1.0172e-008, which is very close to zero. According to theorem 3.1, the function $u(x)$, with the values in table I, is a solution for (17). By plotting these data and comparing this graph with the graph of the exact solution, the accuracy of the obtained results are clear (see Fig.1). Obviously, if we choose a smaller h, (say h=.001) better results will be achieved. Now, what has just remained is to find a formula for $u(x)$ which interpolates $u(x_i)$, or find a curve that fits these data in the best possible way. By using Spline interpolation, good results will be achieved.

In the following figures, dots and circles show the approximate values obtained by our approach and the connected line indicates the graph of exact solutions.

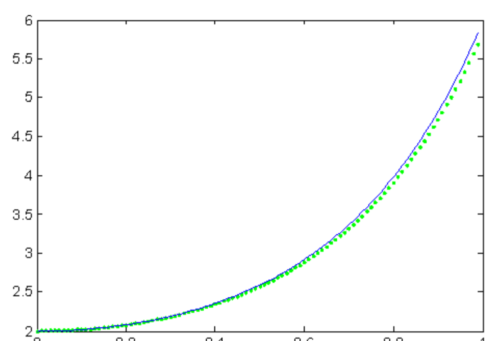

Fig. 1. The values $u(x_i)$ of example 4.1

Table I. Approximate values for $u(x_i)$ of example 4.1

i=1:10	i=10:20	i=20:30	i=30:40	i=40:50	i=50:60	i=60:70	i=70:80	i=80:90	i=90:100
2.0000	2.0171	2.0760	2.1812	2.3402	2.5672	2.8829	3.3160	3.9063	4.7097
2.0012	2.0210	2.0844	2.1945	2.3596	2.5944	2.9204	3.3672	3.9759	4.8045
2.0012	2.0253	2.0932	2.2083	2.3796	2.6225	2.9590	3.4199	4.0477	4.9024
2.0017	2.0301	2.1025	2.2227	2.4004	2.6515	2.9989	3.4744	4.1218	5.0034
2.0026	2.0354	2.1122	2.2377	2.4219	2.6815	3.0401	3.5305	4.1982	5.1075
2.0039	2.0410	2.1225	2.2532	2.4441	2.7124	3.0826	3.5884	4.2770	5.2150
2.0057	2.0471	2.1332	2.2694	2.4671	2.7444	3.1264	3.6481	4.3583	5.3259
2.0079	2.0537	2.1444	2.2861	2.4909	2.7774	3.1716	3.7097	4.4421	5.4403
2.0105	2.0607	2.1562	2.3035	2.5155	2.8114	3.2183	3.7732	4.5285	5.5583
2.0136	2.0681	2.1684	2.3216	2.5409	2.8466	3.2664	3.8387	4.6177	5.6801

Table II. Approximate values for $u(x_i)$ of example 4.2

i=1:10	i=11:20	i=21:30	i=31:40	i=41:50	i=51:60	i=61:70	i=71:80	i=81:90	i=91:100
0	0.0322	0.1232	0.2601	0.4412	0.6814	1.0269	1.6035	2.8189	6.8789
0	0.0389	0.1350	0.2762	0.4621	0.7100	1.0709	1.6845	3.0185	7.8482
0.0008	0.0463	0.1472	0.2926	0.4836	0.7397	1.1172	1.7717	3.2424	9.0909
0.0023	0.0541	0.1598	0.3095	0.5058	0.7706	1.1660	1.8659	3.4952	10.7365
0.0046	0.0625	0.1729	0.3269	0.5286	0.8026	1.2176	1.9680	3.7827	13.0081
0.0076	0.0714	0.1864	0.3447	0.5521	0.8360	1.2723	2.0791	4.1124	16.3179
0.0112	0.0808	0.2003	0.3630	0.5763	0.8709	1.3303	2.2002	4.4938	21.5051
0.0155	0.0907	0.2146	0.3818	0.6013	0.9072	1.3920	2.3329	4.9399	30.5094
0.0205	0.1011	0.2294	0.4011	0.6271	0.9453	1.4579	2.4788	5.4681	48.6854
0.0260	0.1119	0.2445	0.4209	0.6538	0.9851	1.5282	2.6400	6.1028	95.2489

Example 4.2. Solve the following non-linear integral equation

$$u(x)=\int_0^x 2t(4+u^2(t))dt, \quad 0\le x\le 0.88.$$

Clearly $u(0)=0$. The exact solution of this equation is $u(x)=2\tan(2x^2)$, the results obtained by applying our method are gathered in table II, and the objective function value is 0.0015. Fig.2 shows the result.

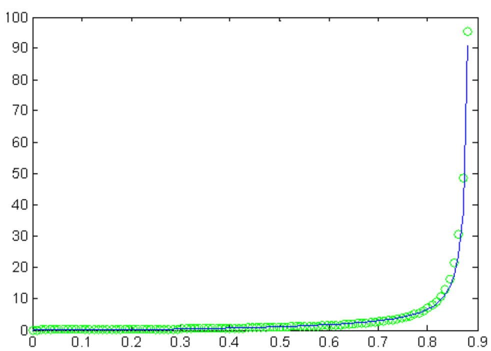

Fig. 2. The values $u(x_i)$ of example 4.2

Example 4.3. Consider the following NVIE:

$$u(x)=2x-(1/12)x^4+0.25\int_0^x (x-t)u^2(t)dt, \; 0\le x\le 1$$

With the exact solution $u(x)=2x$. Clearly u(0)=0. Using the proposed method and considering $h=0.1$, approximate values of $u(x_i)$ $(i=1,2,\ldots,10)$ are collected in table III. In this case, the objective function value is 5.1270e-010. It is clear that these data are very close to the values of the exact solution $u(x)=2x$.

Table III. Approximate values for $u(x_i)$ of example 4.3

i=1:2	i=3:4	i=5:6	i=7:8	i=9:10
0	0.4000	0.7999	1.1997	1.5993
0.2000	0.5999	0.9998	1.3995	1.7991

Example 4.4. Consider the non-linear Volterra integral equation,

$$u(x)=x+(1/5)x^5-\int_0^x tu^3(t)dt, \quad 0\le x\le 1,$$

with exact solution $u(x)=x$ and $u(0)=0$. By choosing $h=0.1$, then the values of $u(x_i)$ $(i=1,2,\ldots,10)$ are given in table IV. The function value is 1.2460e-011.

Table IV. Approximate values for $u(x_i)$ of example 4.4

i=1:2	i=3:4	i=5:6	i=7:8	i=9:10
0	0.2001	0.4011	0.6056	0.8172
0.1000	0.3003	0.5027	0.7104	0.9261

Example 4.5.([9]) Now, we solve the non-linear Volterra integral equation as follows:

$$u(x)=\tan(x)-.25\sin(2x)+\int_0^x \frac{1}{1+u^2(t)}dt, \qquad 0\le x\le 1,$$

with the exact solution $u(x)=\tan(x)$. Using the proposed method and letting $h=0.01$, values for $u(x_i)$ are presented in table V. The function value is 4.4747e-009 . See Fig.3.

Table V. Approximate values for $u(x_i)$ of example 4.5

i=1:10	i=11:20	i=21:30	i=31:40	i=41:50	i=51:60	i=61:70	i=71:80	i=81:90	i=91:100
0	0.1009	0.2037	0.3103	0.4235	0.5468	0.6849	0.8436	1.0314	1.2620
0.0101	0.1111	0.2141	0.3213	0.4353	0.5599	0.6997	0.8609	1.0523	1.2881
0.0201	0.1213	0.2246	0.3323	0.4472	0.5731	0.7147	0.8785	1.0736	1.3150
0.0302	0.1315	0.2351	0.3434	0.4593	0.5865	0.7300	0.8964	1.0953	1.3425
0.0403	0.1417	0.2457	0.3546	0.4714	0.6000	0.7455	0.9146	1.1175	1.3708
0.0503	0.1519	0.2563	0.3659	0.4837	0.6137	0.7612	0.9332	1.1402	1.3999
0.0604	0.1622	0.2670	0.3772	0.4960	0.6276	0.7771	0.9521	1.1634	1.4298
0.0705	0.1725	0.2777	0.3887	0.5085	0.6416	0.7934	0.9714	1.1872	1.4605
0.0806	0.1829	0.2885	0.4002	0.5212	0.6558	0.8098	0.9910	1.2115	1.4922
0.0908	0.1932	0.2994	0.4118	0.5339	0.6703	0.8266	1.0110	1.2364	1.5248

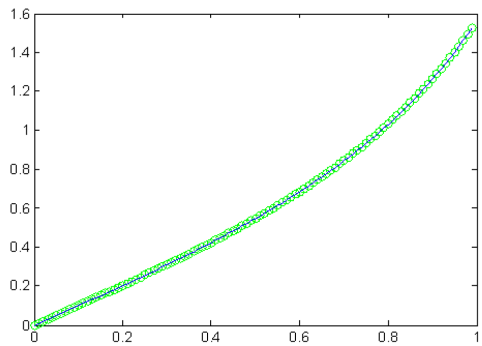

Fig. 3. The values $u(x_i)$ of example 4.5

Example 4.6: Consider the following Volterra integral equation of the first kind

$$\int_0^x \cos(x-t)u(t)dt = x\sin x.$$

This integral equation has been solved by an approximate method in [7] and the exact solution of this integral equation is $u(x) = 2\sin x$. We have compared the results obtained from our method by the exact solution in Fig.4.

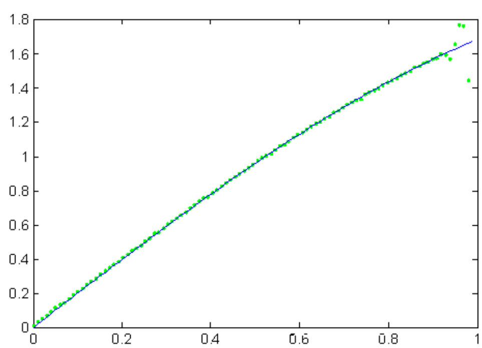

Fig. 4. The values $u(x_i)$ of example 4.6

V. Conclusion

In this paper, we proposed a numerical method for solving LVIE and NVIE of the first and the second kind. We showed the accuracy of this method by solving different examples, and compared the exact solutions with the approximate solutions. We can also suggest that this approach be used for solving Volterra integro-differential equations as well. For this purpose, we just need to use some approximate formulas for the derivatives of unknown function u. For example, we can use $\dfrac{u(x_{i+1})-u(x_i)}{\Delta x_i}$ instead of $u'(x)$, and for higher order derivatives, one can use other formulas from numerical calculus. Afterward, by substituting these in (10) and doing the same procedure as we did before, we reach a discrete problem. Then, we solve this problem instead of solving the main integro-differential equation.

References

[1] S. Abbasbandy, "Application of He's homotopy perturbation method to functional integral equations," Chaos, Solitons and Fractals. Vol. 31.2007. pp. 1243-1247.

[2] K. Atkinson, "The Numerical solution of Integral Equations of second kind, " Cambridge University Press, 1997.

[3] M.S. Bazaraa, C.M. shetty, "Nonlinear programming: Theory and Algorithms," Jon Wiley and Sons, Inc., New York, 1979.

[4] A. Ghorbani, J. Saberi-Nadjafi, "Exact solutions for nonlinear integral equations by a modified homotopy perturbation method," Computers & Mathematics with Applications, Vol. 56, Issue 4, August 2008, pp. 1032-1039.

[5] A. Hashemi Borzabadi, A. Vahidian Kamyad, H. Hashemi Mehne, "A different approach for solving the nonlinear Fredholm Integral Equations of second kind," Appl. Math. Comput. Vol. 173, 2006, pp. 724-735.

[6] A. J. Jerri, "Introduction to Integral Equations with Applications," John Wiley and Sons, Inc. 1999.

[7] K. Maleknejad, H. Derili, "Numerical solution of integral equations by using combination of spline-collocation method and lagrange interpolation," Appl. Math. Comput. Vol. 175, 2006, pp. 1235-1244.

[8] W. Wang, "A mechanical algorithm for solving the Volterra integral equations," Appl. Math. Comput. Vol. 172, 2006 pp. 1323-1341.

[9] A.M. Wazwaz, "A First Course in Integral Equations, " World Scientific, New Jersey,1997.