Palestine Journal of Mathematics

Vol. 12(4)(2023) , 425–439

Numerical Solution of System of Linear Volterra Integro-Differential Equations by Reconstruction of Variational Iteration Method

Ahmad Issa

Communicated by Suheel Khoury

MSC 2010 Classifications: Primary 34K28, 45J05, 65R20.

Keywords and phrases: VIDEs, RVIM, Chebyshev wavelets, Sinc collocation, stability.

Abstract In this paper, the Reconstruction of Variational Iteration Method (RVIM) is implemented to solve a linear system of Volterra Integro-Differential Equations (VIDEs). Three numerical examples with known exact solutions are presented to test the efficiency of this method. Moreover, a comparison of the RVIM with the Sinc collocation method and the Chebyshev wavelets method from our previous work has been carried out. Numerical results clearly show that the Chebyshev wavelets method provides more accurate results when the kernel and exact solution of VIDEs are not polynomials whereas, the RVIM is more efficient than its counterparts when the kernel and exact solution of VIDEs are polynomials.

1 Introduction

Systems of integro-differential equations appear frequently in various fields involving science and engineering, namely, wind ripple in the desert, nano-hydrodynamics, population growth model, glass-forming process, and oceanography [3][11]. Many numerical methods for solving systems of linear integro-differential equations have been developed by many researchers. Rahimi [12] used the Reconstruction of variational iteration method for solving systems of Volterra integro-differential equations. This method was compared with the homotopy perturbation method and proved to be more accurate. Agarwal et al. [16] applied the Optimal Homotopy Asymptotic Method for the solution of system of Volterra integro-differential equations. In [7] Biazar et al. implemented the Homotopy Perturbation Method to approximate the solution of systems of Volterra integro-differential equations. In addition, Issa et al. [5] have employed the Sinc collocation and the Chebyshev wavelets methods for solving systems of linear Volterra integro-differential equations. Other numerical techniques for solving systems of integro-differential equations include the differential transform method [2], Power series method [9], Chebyshev polynomials [3], and Single Term Walsh Series technique [1]. In [10], Holmaker proved the global asymptotic stability for a stationary solution of a system of integro-differential equations describing the formation of liver zones. Bloom [4] has studied the concavity arguments and growth estimates for damped linear integro-differential equations with applications to a class of holohedral isotropic dielectrics.
In this work, we implement the reconstruction of variational iteration method for solving a system of linear Volterra integro-differential equations given by

$$u_i^{(n)}(x) = f_i(x) + \int_0^x \left(\sum_{j=1}^{N} k_{ij}(x,t) u_j(t) \right) dt, a \le x \le b, 1 \le i \le N \tag{1.1}$$

$$u_i^{(s)} = a_{is}, i = 1, 2, 3, \ldots, N, s = 0, 1, 2, \ldots, (n-1), \tag{1.2}$$

where the kernels $k_{ij}(x,t)$ and the functions $f_i(x)$ are given real-valued functions and $u_i(x)$ are the unknown functions of equation (1.1).

In addition, a comparison of the RVIM with the Sinc collocation method and the Chebyshev wavelets method from our previous work has been carried out.

2 Reconstruction of Variational Iteration Method

The reconstruction of variational iteration method based on the Laplace transform has been induced from the variational iteration method that was developed from the Inokutti technique. The main feature of RVIM is that it provides fast convergence to the exact solution and is known for its simplicity in computation without any restrictive assumptions [12][15].
Before introducing the RVIM, it is necessary to present the following definitions and theorems related to the Laplace transform method

Definition 2.1. (see [13][15]) Let $f(x)$ be a real-valued function defined for $x \geq 0$, then the Laplace transform of $f(x)$ denoted by $L[f(x)]$ is given as:

$$L[f(x)] = F(s) = \int_0^\infty e^{-sx} f(x)dx = \lim_{a\to\infty} \int_0^a e^{-sx} f(x)dx \tag{2.1}$$

Definition 2.2. (see [13][15]) If $L[f(x)] = F(s)$, then $f(x) = L^{-1}[F(s)]$ is called the inverse Laplace transform of $F(s)$.

Definition 2.3. (see [12][15]) The convolution of two piecewise continuous functions $f, g : R \to R$ is the function $f * g$ given by

$$(f * g)(x) = \int_0^x f(\tau)g(x-\tau)d\tau$$

Theorem 2.4. *(see [12][15]) (Convolution Theorem)*
Let $f(x)$ and $g(x)$ are piecewise continuous functions on $[0,\infty)$, then

$$L[f * g] = L\left[\int_0^x f(\tau)g(x-\tau)d\tau\right] = L[f(x)].L[g(x)] = F(s).G(s)$$

Also we can conclude that:

$$L^{-1}[F(s).G(s)] = f * g = \int_0^x f(\tau)g(x-\tau)d\tau$$

Definition 2.5. (see [13][15]) The Laplace transform for the derivatives of $f(x)$ is given by:

$$L[f^{(n)}(x)] = s^n F(s) - s^{n-1}f(0) - s^{n-2}f'(0) - \ldots - f^{(n-1)}(0) \tag{2.2}$$

We consider the system of integro-differential equation of the form:

$$u_i^{(n)}(x) = f_i(x) + \int_0^x (k_{i1}(x,t)u_1(t) + k_{i2}(x,t)u_2(t) + \ldots + k_{ii}(x,t)u_i(t))dt \tag{2.3}$$

where $u_k^{(n)}$ is the derivative of u_k of order n, subject to the initial conditions:

$$u_k^{(j)} = c_j^k, 1 \leq k \leq i, 0 \leq j \leq n-1 \tag{2.4}$$

We can write system (2.3) as

$$u_k^{(n)}(x) = M_k(x, u_1(x), u_2(x), \ldots, u_i(x)), k = 1, \ldots, i, \tag{2.5}$$

with the zero artificial initial conditions.
Applying the Laplace transform to (2.5) and using the artificial initial conditions, we obtain:

$$s^n L[u_k(x)] = L[M_k(x, u_1(x), u_2(x), \ldots, u_i(x))]$$

Dividing both sides by s^n we get:

$$L[u_k(x)] = \frac{1}{s^n} L[M_k(x, u_1(x), u_2(x), \ldots, u_i(x))]$$

If we set $\frac{1}{s^n} = G(s)$, then by using the convolution theorem, we get :

$$L[u_k(x)] = G(s)L[M_k(x, u_1(x), u_2(x), \ldots, u_i(x))] = L[(g * M_k)(x)], \tag{2.6}$$

where $k = 1, \ldots, i; L^{-1}[G(s)] = g(x)$.
Taking the inverse Laplace transform to both sides of (2.6), we obtain:

$$u_k(x) = \int_0^x g(x-t)M_k(t, u_1(t), u_2(t), \ldots, u_i(t))dt, k = 1, \ldots, i$$

To impose the particular initial conditions to get the answer of (2.3), we have the following iteration form:

$$u_k^{<m+1>}(x) = u_k^{(0)}(x) + \int_0^x g(x-t)M_k(t, u_1^{<m>}(t), u_2^{<m>}(t), \ldots, u_i^{<m>}(t))dt, k = 1, \ldots, i. \tag{2.7}$$

The values $u_1^{(0)}(x), u_2^{(0)}(x), \ldots, u_i^{(0)}(x)$ are given by:

$$u_k^{(0)}(x) = u_k(0) + xu_k'(0) + \frac{x^2 u_k''(0)}{2!} + \frac{x^r u_k^{(r)}(0)}{r!} \tag{2.8}$$

Therefore, the reconstruction of variation iteration method $u_k(x)$ is obtained as follows:

$$u_k(x) = \lim_{m\to\infty} u_k^{<m>}(x), k = 1, \ldots, i, \tag{2.9}$$

where $u_k^{<m>}(x)$ indicates m^{th} approximation of $u_k(x)$.

The RVIM is illustrated by Algorithm 1

Algorithm 1 **Numerical Realization Using RVIM**

1. Input
   - *a.* N, n
   - *b.* $f_i(x)$ for $i = 1, 2, \ldots, N$
   - *c.* $k_{ij}(x,t)$ for $i = 1, 2, \ldots, N; j = 1, 2, \ldots, N$
   - *d.* Initial condition $U_i(0); U_i^{(d)}(0)$ for $i = 1, 2, \ldots, N; d = 1, 2, \ldots, (n-1)$

2. Define $u_i^{(n)}(x) = f_i(x) + \int_0^x \left(\sum_{j=1}^N k_{ij}(x,t)u_j(t)\right) dt$

3. Applying Laplace transformation for both sides

4. Substitute artificial initial condition $u_i(0) = 0$ for $i = 1, 2, \ldots, N$;
$u_i^{(d)}(0) = 0$ for $i = 1, 2, \ldots, N; d = 1, 2, \ldots, (n-1)$

5. Use convolution theory

6. Set $u_{i0}(t) = U_i(0) + \sum_{j=1}^{n-1} \frac{u_i^{(j)}(0) \times t^{(j)}}{j!}$ for $i = 1, 2, \ldots, N$

7. Applying recurrence iteration to calculate $u_{im}(x)$ for $i = 1, 2, \ldots, N; m = 1, 2, \ldots, M$

8. Input
   - *a.* $u_{approx(i)}(x) = u_{im}(x)$ for $i = 1, 2, \ldots, N$
   - *b.* $u_{exact(i)}(x)$ for $i = 1, 2, \ldots, N$

9. Plot $u_{approx(i)}(x); u_{exact(i)}(x)$

10. Define error $|u_{exact(i)}(x) - u_{approx(i)}(x)|$; Plot error.

3 Stability of Systems of Volterra Integro-Differential Equations (VIDEs)

In this section, we present some important results on the stability of VIDEs (1.1)(for more details see [5][6]).

Definition 3.1. (see [5][6]) If $\mu : [0, x_0] \longrightarrow R$ is a continuous initial function, then $u(x, x_0, \mu)$ will denote the solution of (1.1) on $[x_0, \infty]$. Frequently, it is sufficient to write $u(x)$. If $f_i(0) = u_i(0) = 0$ then $u(x) \equiv 0$ is a solution of (1.1) called the zero solution.
The norm on the initial function $\mu(x) = (\mu_1(x), \mu_2(x), \ldots, \mu_n(x))$ is given by

$$\parallel \mu \parallel = \sup \left[|\mu| = \sum_{i=1}^{n} \mu(i) : 0 \leq x \leq x_0 \right]. \tag{3.1}$$

The definition of stability of the zero solution is given in Burton [14] and is restated below

Definition 3.2. (see [5][6]) The zero solution of (1.1) is stable if for each $\epsilon > 0$ and each $x_0 \geq 0$, there exists δ such that $|\mu| \leq \delta$ on $[0, x_0]$ and $x \geq x_0$ imply $|u(x, x_0, \mu)| \leq \epsilon$.

We next define the statement "a Lyapunov functional for system (1.1)". Let $V(x, \psi(.))$ be defined for $x \geq 0$ and $\psi \in ([0, x]; R^n)$ and let V be locally Lipschitz in ψ. For each $x \geq 0$ and every $\psi \in ([0, x]; R^n)$, we define the derivative V along a solution of (1.1) by

$$V'_{(1)}(x, \psi(.)) = \lim_{\triangle x \to 0^+} \sup \frac{V(x + \triangle x, u(\cdot, x, \psi)) - V(x, \psi(.))}{\triangle x} \tag{3.2}$$

where $u(\xi; x, \psi)$ is the unique solution of (1.1) with initial conditions x and ψ. Then the following result by Driver [5][8] gives a definition of the Lyapunov functional.

Theorem 3.3. *(see [5][8]) If $V(x, \psi(.))$ is defined for $x \geq 0$ and every $\psi \in ([0, x]; R^n)$ with*

(i) $V(x, 0) \equiv 0$.

(ii) *V continuous in x and Lipschitz in ψ.*

(iii) *$V(x, \psi(.)) \geq W(|\psi(.)|)$, where $W : [0, \infty] \longrightarrow [0, \infty]$ is a continuous function with*

$$W(0) = 0, W(r) > 0, r > 0 \tag{3.3}$$

and W strictly increasing (positive definiteness).

(iv) *$V'_{(1)}(x, \psi(.)) \leq 0$, then the zero solution of (1.1) is stable, and*

$$V(x, \psi(.)) = V(x, \psi(t)) : 0 \leq t \leq x \tag{3.4}$$

is called a Lyapunov functional for system (1.1).

Finally, we assumed that the functions in (1.1) are well behaved, that continuous initial functions generate solutions, and that solutions which remain bounded can be continued.

4 Numerical Examples and Results

In this section, some numerical examples are presented to demonstrate the efficiency of the proposed method. Also, the numerical results are compared with the Sinc collocation method and the Chebyshev wavelet method.

Example 4.1. [5] Consider the system of Volterra integro-differential equations

$$u'_1(x) = -2 + x^2 - x^4 + \frac{3}{20}x^5 + 2x^6 + \frac{1}{5}x^7 - \frac{1}{8}x^8 + \int_0^x (t^3 - x^2)u_1 + (12t^2 - x)u_2 dt,$$

$$u'_2(x) = 4 - 8x - \frac{1}{3}x^3 + 2x^4 - \frac{8}{5}x^5 + \frac{1}{30}x^6 - 4e^x + \int_0^x (t - x)u_1 + 8(1 - t)u_2 + 2u_3 dt,$$

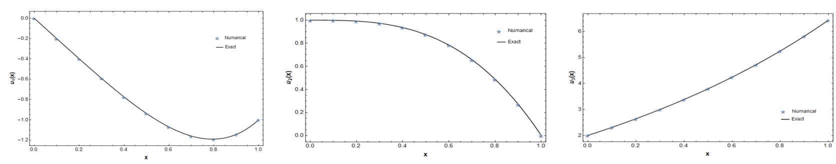
$$u_3'(x) = 3 - \frac{7}{2}x^2 + \frac{4}{3}x^3 + \frac{6}{5}x^5 - \frac{7}{30}x^6 + \int_0^x (2x-t)u_1 + 6tu_2 + u_3 dt, \tag{4.1}$$

subject to the initial conditions

$$u_1(0) = 0, u_2(0) = 1, u_3(0) = 2$$

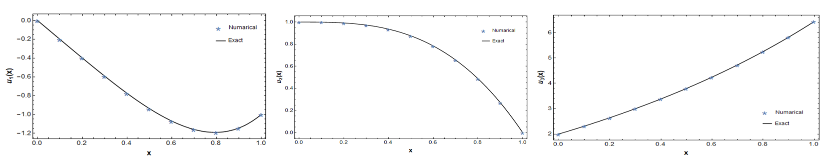
The exact solution of system (4.1) is $u_1(x) = x^4 - 2x, u_2(x) = 1 - x^3$ and $u_3(x) = x + 2e^x$.

We start by applying Algorithm 1 to solve system (4.1). Tables 1 and 2 contain the numerical results for the RVIM and the Sinc collocation method (SCM) [5] in addition to the Chebyshev wavelet method (CWM) [5]. Tables 3 and 4 show the resulting error for RVIM, SCM, and CWM respectively with $M = 3$. Figure 1:(a,b,c) compares both the numerical and exact solutions using RVIM. Figure 2:(a,b,c) compares both the numerical and exact solutions using SCM. Figure 3:(a,b,c) compares both the numerical and exact solutions using CWM. Figure 4: shows the absolute error resulting of applying Algorithm 1. The maximum error corresponding to u_1, u_2 and u_3 respectively by RVIM is $E_1 \approx 9.44e-5$, $E_2 \approx 2.98e-5$ and $E_3 \approx 5.11e-5$.



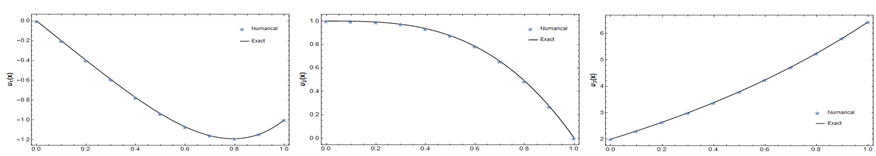
(a) (b) (c)

Figure 1: The exact and numerical solutions using Algorithm 1 for system (4.1).



(a) (b) (c)

Figure 2: The exact and numerical solutions using Algorithm of the Sinc method for system (4.1).



(a) (b) (c)

Figure 3: The exact and numerical solutions using Algorithm of the Chebyshev wavelet method for system (4.1).

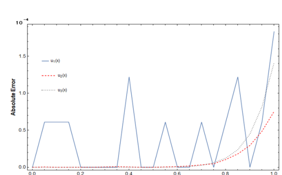


Figure 4: The resulting error after applying Algorithm 1 for system (4.1).

Table 1: The exact and numerical solutions of u_1 and u_2 for system (4.1)

x	u_{1exact}	u_{1app} [RVIM]	u_{1app} [SCM][5]	u_{1app} [CWM][5]	u_{2exact}	u_{2app} [RVIM]	u_{2app} [SCM][5]	u_{2app} [CWM][5]
0	0	0	0	2.690833666996184e-13	1	1	1	0.9999999999543171
0.1	−0.1999	-0.19989013671875	-0.1998999986410037	-0.19990000000022942	0.999	0.9989995956420898	0.9989999996164405	0.9990000000727636
0.2	−0.3984	-0.3984375	-0.39840000469382025	-0.39839999999984066	0.992	0.991999626159668	0.992000001268595	0.991999999990548
0.3	-0.5919	-0.59185791015625	-0.5919000795392859	-0.5919000000004384	0.973	0.9730005264282227	0.9730000216099085	0.9729999999906025
0.4	-0.7744	-0.77435302734375	-0.774400228030929	-0.7744000000009819	0.936	0.9360003471374512	0.9360000615945837	0.9360000000694738
0.5	-0.9375	-0.9375	-0.9375003156846542	-0.9374999999997702	0.875	0.875	0.8750000858865024	0.875000000131186
0.6	-1.0704	-1.07037353515625	-1.0704001918522101	-1.0703999999956941	0.784	0.783998966217041	0.784000056734032	0.7840000001191981
0.7	-1.1599	-1.15985107421875	-1.1598998894031196	-1.1598999999883794	0.657	0.6569967269897461	0.6569999842789827	0.6570000000729469
0.8	-1.1904	-1.19030761171875	-1.190399643757478	-1.190399999977122	0.488	0.4879894256591797	0.4879999282244343	0.4880000000954549
0.9	-1.1439	-1.14398193359375	-1.1438996142131692	-1.1438999999585011	0.271	0.27097034454345703	0.2709999270259329	0.27100000021849296

Table 2: The exact and numerical solutions of u_3 for system (4.1)

x	u_{3exact}	u_{3app}		
		[RVIM]	[SCM][5]	[CWM][5]
0	2	2	2	1.9999999974820484
0.1	2.3103418361512955	2.310341827571392	2.310341836911406	2.3103418351947798
0.2	2.64280551632034	2.6428054869174957	2.642805513660997	2.6428055151638277
0.3	2.999717615152006	2.999717593193054	2.9997175706314225	2.999717617646919
0.4	3.3836493952825406	3.3836494088172913	3.3836492678242998	3.3836493955712172
0.5	3.7974425414002564	3.7974424362182617	3.7974423632300143	3.7974425389296056
0.6	4.244237600781018	4.244237184524536	4.244237484652817	4.244237600772698
0.7	4.727505414940953	4.727502703666687	4.72750545433764	4.727505417496077
0.8	5.251081856984936	5.25106954574585	5.251082019406921	5.2510818561222665
0.9	5.8192062223139	5.819161415100098	5.819206393371196	5.819206221276965

Example 4.2. [5] Consider the system of Volterra integro-differential equations

$$u_1''(x) = -1 - x + coshx - \frac{sin^3 x}{3} - sinhx + e^x + \int_0^x (e^{-t})u_1 + (sin^2 t)u_2 dt,$$

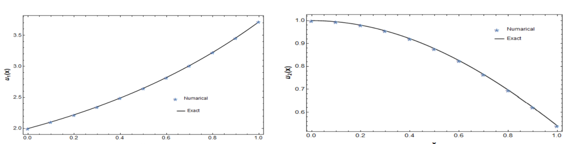
$$u_2''(x) = -3 + x^2 - \frac{2}{3}x^3 - 2e^x(x-1) + \int_0^x (x^2 - t^2)u_1 + (x-t)u_2 dt, \tag{4.2}$$

subject to the initial conditions

$$u_1(0) = 2, u_1'(0) = 1, u_2(0) = 1, u_2'(0) = 0$$

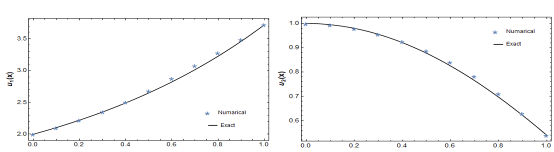
The exact solution of system (4.2) is $u_1(x) = e^x + 1$ and $u_2(x) = cosx$

We start by applying Algorithm 1 to solve system (4.2). Table 5 contains the numerical results for the RVIM and the Sinc collocation method (SCM) [5] in addition to the Chebyshev wavelet method (CWM) [5]. Table 6 shows the resulting error for RVIM, SCM, and CWM respectively with $M = 3$. Figure 5:(a,b) compares both the numerical and exact solutions using RVIM. Figure 6:(a,b) compares both the numerical and exact solutions using SCM. Figure 7:(a,b) compares both the numerical and exact solutions using CWM. Figure 8: shows the absolute error resulting of applying Algorithm 1. The maximum error corresponding to u_1 and u_2 respectively by RVIM is $E_1 \approx 4.68e-6$ and $E_3 \approx 1.01e-6$.



(a) (b)

Figure 5: The exact and numerical solutions using Algorithm 1 for system (4.2).



(a) (b)

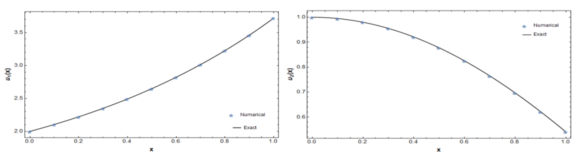
Figure 6: The exact and numerical solutions using Algorithm of the Sinc method for system (4.2).

Table 3: The resulting error for the numerical solution of u_1 and u_2 with $M = 3$.

	Absolute error $\|u_1 - u_{1app}\|$			Absolute error $\|u_2 - u_{2app}\|$		
x	[RVIM]	[SCM][5]	[CWM][5]	[RVIM]	[SCM][5]	[CWM][5]
0	0	0	2.690833666996184e-13	0	0	4.568290190576363e-11
0.1	0.000009863281250022116	1.358996248868038e-9	2.293998324631729e-13	4.043579101553618e-7	3.835595174805917e-10	7.276357294472291e-11
0.2	0.00003749999999968114	4.693820221390865e-9	1.593725151849412e-13	3.738403320241446e-7	1.268595006820305e-9	9.451994742448733e-12
0.3	0.000042089843750092726	7.953928582438152e-8	4.383160501220118e-13	5.264282226802308e-7	2.160990852928535e-8	9.397482791939638e-12
0.4	0.00004697265624997726	2.280309290281224e-7	9.818812429784884e-13	3.471374512287184e-7	6.159458376675531e-8	6.947387110045611e-11
0.5	0	3.156846541951807e-7	2.298161660974074e-13	0	8.588650235452633e-8	1.311859509911528e-10
0.6	0.00002646484375001812	1.91852210118526e-7	4.305888978706207e-12	0.00000103378295890355	5.673403202788307e-8	1.191982068604602e-10
0.7	0.00004892578124993108	1.10596880320557e-7	1.162048235414658e-11	0.000003273010253823649	1.572101726576846e-8	7.294698178839099e-11
0.8	0.00009238281249990266	3.562425212599862e-7	2.287792177924075e-11	0.00001057434082019082	7.177556560211684e-8	9.545503276697787e-11
0.9	0.00008193359375008313	3.857868307033385e-7	4.14988043928588e-11	0.00002965545654287638	7.297406701134435e-8	2.184930569804066e-10

Table 4: The resulting error for the numerical solution of u_3 with $M = 3$.

	Absolute error $\vert u_3 - u_{3app}\vert$		
x	[RVIM]	[SCM][5]	[CWM][5]
0	0	0	2.517951624980696e-9
0.1	8.579903454375426e-9	7.601106410959346e-10	9.565157554902726e-10
0.2	2.940284415942074e-8	2.659342968058808e-9	1.156512219324668e-9
0.3	2.195895199008646e-8	4.452058366410938e-8	2.494912720862885e-9
0.4	1.353475065357656e-8	1.274582408505864e-7	2.886766381493544e-10
0.5	1.051819946695786e-7	1.781702421155273e-7	2.470650795061146e-9
0.6	4.16256481727828e-7	1.161282012773767e-7	8.319567257331073e-12
0.7	0.000002711274266431473	3.939668680175146e-8	2.555123224112776e-9
0.8	0.000012311239085960324	1.624219851947828e-7	8.62669047307918e-10
0.9	0.00004480721380240027	1.710572963276035e-7	1.0369349823236e-9



(a) (b)

Figure 7: The exact and numerical solutions using Algorithm of the Chebyshev wavelet method for system (4.2).

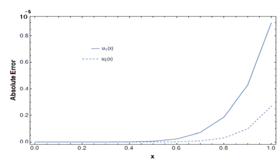


Figure 8: The resulting error after applying Algorithm 1 for system (4.2).

Example 4.3. Consider the system of Volterra integro-differential equations

$$u_1'(x) = 1 - 7x^2 - \frac{x^3}{6} - 4x^4 - \frac{x^5}{20} + \int_0^x (x-t)u_1 + 6xu_2 dt,$$

$$u_2'(x) = 4x + \frac{10}{3}x^2 - \frac{x^3}{3} + x^4 - \frac{x^5}{5} + \int_0^x tu_1 + (2t-3x)u_2 dt, \tag{4.3}$$

subject to the initial conditions

$$u_1(0) = 0, u_2(0) = \frac{5}{3}$$

The exact solution of system (4.3) is $u_1(x) = x^3 + x$ and $u_2(x) = 2x^2 + \frac{5}{3}$

We start by applying Algorithm 1 to solve system (4.3). Table 7 contains the numerical results for the RVIM and the Sinc collocation method (SCM) in addition to the Chebyshev wavelet method (CWM). Table 8 shows the resulting error for RVIM, SCM, and CWM respectively with $M = 10$. Figure 9:(a,b) compares both the numerical and exact solutions using RVIM. Figure 10:(a,b) compares both the numerical and exact solutions using SCM. Figure 11:(a,b) compares both the numerical and exact solutions using CWM. Figure 12: shows the absolute error resulting of applying algorithm of the Sinc method. Figure 13: shows the absolute error resulting of applying algorithm of the Chebyshev wavelet method. We see clearly that the maximum error corresponding to u_1 and u_2 is zero when applying the RVIM, that is the approximate solution converges to the exact one. On the other hand the maximum error corresponding to u_1 and u_2 when applying the SCM is $E_1 \approx 8.5e-8$ and $E_2 \approx 1.42e-8$. Moreover, the maximum error

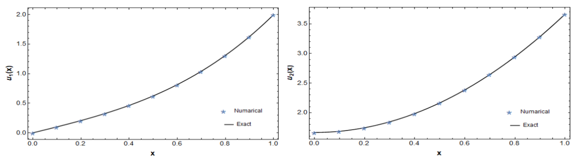
Table 5: The exact and numerical solutions for system (4.2)

		u_{1app}				u_{2app}		
x	u_{1exact}	[RVIM]	[SCM][5]	[CWM][5]	u_{2exact}	[RVIM]	[SCM][5]	[CWM][5]
0	2	2	2	1.999999998740645	1	1	1	0.9999999993413167
0.1	2.1051709180756477	2.105170918075422	2.1051397636262448	2.105170917580036	0.9950041652780258	0.9950041652780328	0.9949953813521188	0.995004164957528l
0.2	2.2214027581601696	2.2214027581072777	2.2212621771873557	2.22140275754089	0.9800665778412416	0.980066577840887	0.9800269511507578	0.9800665775316376
0.3	2.349858807576003	2.349858806343985	2.353150577162296	2.349858808759274	0.955336489125606	0.9553364891056901	0.9562638272781463	0.9553364897343352
0.4	2.4918246976412703	2.491824686493494	2.5101191909529916	2.491824697676863	0.9210609940028851	0.9210609936577114	0.9262143139859589	0.9210609939384683
0.5	2.648721270700128	2.6487212107128038	2.694898115680843	2.6487212692830155	0.8775825618903728	0.8775825587614037	0.890587846895059	0.8775825611033266
0.6	2.822118800390509	2.822118568272824	2.8955876162197285	2.822118800114823	0.8253356149096782	0.8253355960995421	0.8460218395509642	0.8253356147660844
0.7	3.0137527074704766	3.013751992402314	3.094820636647796	3.01375270838393	0.7648421872844884	0.7648421021871812	0.7876576255740133	0.764842187697923
0.8	3.225540928492468	3.225539063307953	3.289093529586907	3.2255409275988933	0.6967067093471654	0.6967063969228278	0.7145821371224699	0.6967067087823426
0.9	3.45960311115695	3.4595988198549152	3.4945194299151083	3.459603110051884	0.6216099682706644	0.6216089910088343	0.631444245918761	0.6216099676731845

Table 6: The resulting error for the numerical solution with $M = 3$.

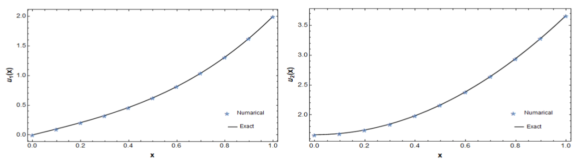
	Absolute error $\|u_1 - u_{1app}\|$			Absolute error $\|u_2 - u_{2app}\|$		
x	[RVIM]	[SCM][5]	[CWM][5]	[RVIM]	[SCM][5]	[CWM][5]
0	0	0	1.25935506467556e-9	0	0	6.586833301014394e-10
0.1	2.255973186038318e-13	0.00003115444940293699	4.956115517984472e-10	6.994405055138486e-15	0.00000878392590697441	3.204977394588582e-10
0.2	5.289191307156216e-11	0.00014058097281388626	6.192797386006532e-10	3.54605234065275e-13	0.00003962669048385159	3.096040090966312e-10
0.3	1.232018043140215e-9	0.0032917695862928475	1.183270814664183e-9	1.991584674954083e-11	0.0009273381525403135	6.087291781753379e-10
0.4	1.114777647970299e-8	0.018294493311721283	3.559286199106282e-11	3.451736674264793e-10	0.005153319983073779	6.441680522328852e-11
0.5	5.99873244411242e-8	0.046176844980714726	1.417112649448881e-9	3.128969083832089e-9	0.013005285004686229	7.870462059855754e-10
0.6	2.321176850728079e-7	0.07346881582921938	2.756861405828203e-10	1.881013611537696e-8	0.020686224641286	1.435938035143635e-10
0.7	7.150681624601418e-7	0.08106792917731953	9.134533129895317e-10	8.509730720085429e-8	0.022815438289524925	4.134346198725325e-10
0.8	0.000001865184514837636	0.06355260109443917	8.935745476890133e-10	3.124243376229074e-7	0.017875427775304487	5.64822744131277e-10
0.9	0.000004291302034609146	0.03491631875815848	1.105065816631167e-9	9.772618301262526e-7	0.009834277648096634	5.974798433783235e-10

corresponding to u_1 and u_2 when applying the CWM is
$E_1 \approx 5.7e - 16$ and
$E_2 \approx 4.4e - 16$.



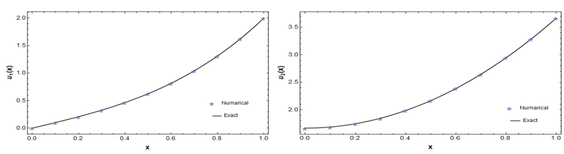
(a) (b)

Figure 9: The exact and numerical solutions using Algorithm 1 for system (4.3).



(a) (b)

Figure 10: The exact and numerical solutions using Algorithm of the Sinc method for system (4.3).



(a) (b)

Figure 11: The exact and numerical solutions using Algorithm of the Chebyshev wavelet method for system (4.3).

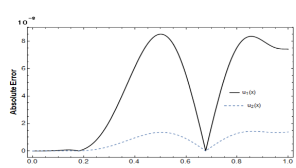


Figure 12: The resulting error after applying Algorithm of the Sinc method for system (4.3).

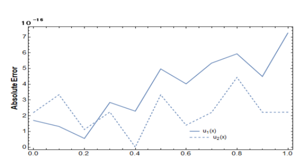


Figure 13: The resulting error after applying Algorithm of the Chebyshev wavelet method for system (4.3).

5 Conclusion

In this paper, we have solved linear systems of Volterra integro-differential equations by the reconstruction of variational iteration method. This method has been implemented in the form of

Table 7: The exact and numerical solutions for system (4.3)

x	u_{1exact}	u_{1app}			u_{2exact}	u_{2app}		
		[RVIM]	[SCM]	[CWM]		[RVIM]	[SCM]	[CWM]
0	0	0	0	1.123904917174857e-16	1.6666666666666667	1.6666666666666667	1.6666666666666667	1.666666666666667
0.1	0.1010	0.1010	0.10100000039140195	0.10100000000000009	1.6866666666666668	1.6866666666666668	1.6866666667257286	1.686666666666667
0.2	0.2080	0.2080	0.2079999987555035	0.20800000000000007	1.7466666666666668	1.7466666666666668	1.7466666664854993	1.7466666666666668
0.3	0.3270	0.3270	0.32699997849740003	0.3270000000000003	1.8466666666666667	1.8466666666666667	1.8466666633866398	1.846666666666667
0.4	0.4640	0.4640	0.46399993875638507	0.4640000000000004	1.9866666666666668	1.9866666666666668	1.9866666571094465	1.9866666666666672
0.5	0.6250	0.6250	0.6249999148772128	0.6250000000000004	2.166666666666667	2.166666666666667	2.166666653108086	2.166666666666667
0.6	0.8160	0.8160	0.8159999445594568	0.8160000000000006	2.3866666666666667	2.3866666666666667	2.3866666576718694	2.386666666666667
0.7	1.0430	1.0430	1.0430000177818546	1.0430000000000008	2.646666666666667	2.646666666666667	2.646666669626049	2.646666666666667
0.8	1.3120	1.3120	1.3120000754536585	1.3120000000000007	2.946666666666667	2.946666666666667	2.946666679401857	2.946666666666667
0.9	1.6290	1.6290	1.6290000803363964	1.6290000000000007	3.286666666666667	3.286666666666667	3.2866666806256957	3.286666666666667

Table 8: The resulting error for the numerical solution with $M = 10$.

	Absolute error $\|u_1 - u_{1app}\|$			Absolute error $\|u_2 - u_{2app}\|$		
x	[RVIM]	[SCM]	[CWM]	[RVIM]	[SCM]	[CWM]
0	0	0	1.123904917174857e-16	0	0	2.220446049250313e-16
0.1	0	3.914019386375145e-10	8.326672684688674e-17	0	5.9061866508614e-11	2.220446049250313e-16
0.2	0	1.244496505847792e-9	5.551115123125783e-17	0	1.811675254259626e-10	0
0.3	0	2.15026000338625e-8	2.220446049250313e-16	0	3.280026916741008e-9	2.220446049250313e-16
0.4	0	6.124361495718489e-8	3.885780586188048e-16	0	9.557220348455076e-9	4.440892098500626e-16
0.5	0	8.512278715233634e-8	4.440892098500626e-16	0	1.355858092466633e-8	0
0.6	0	5.544054337836002e-8	4.440892098500626e-16	0	8.994797351391526e-9	4.440892098500626-16
0.7	0	1.778185443335189e-8	6.661338147750939e-16	0	2.959382072731387e-9	0
0.8	0	7.545365821037819e-8	4.440892098500626e-16	0	1.273518979161281e-8	0
0.9	0	8.033639642412993e-8	6.661338147750939e-16	0	1.3959028599686e-8	0

an algorithm to solve some numerical examples. Numerical results clearly show that:
Example 1 and Example 2 show that the Chebyshev wavelets method is more effective than its counterparts in particular when the kernel and the exact solution are not polynomials. On the other hand, the results for Example 3 indicate that the reconstruction of variational iteration method is more efficient in comparison with the Sinc collocation method and Chebyshev wavelets method when the kernel and exact solution are polynomials.

References

[1] R. C. Guru Sekar and K. Murugesan, System of Linear Second Order Volterra Integro-Differential Equations Using Single Term Walsh Series technique, *Applied Mathematics and Computation* **273**, 484–492 (2016).

[2] A. Arikoglu and I. Ozkol, Solutions of Integral and Integro-Differential Equation systems by Using Differential Transform Method, *Comput. Math. Appl* **56**, 2411–2417 (2008).

[3] H. Aminikhah and S. Hosseini, Numerical Solution of Linear System of Integro-Differential Equations by Using Wavelet Method, *Math. Sci. Letter* **4:1**, 45–50 (2015).

[4] F. Bloom, Concavity Arguments and Growth Estimates for Damped Linear Integro-Differential Equations with Applications to a class of Holohedral Isotropic Dielectrics, *Zeitschrift für angewandte Mathematik und Physik ZAMP* **29:4**, 644–663 (1978).

[5] A. Issa, N. Qatanani and A. Daraghmeh, Approximation Techniques for Solving Linear Systems of Volterra Integro-Differential Equations, *Journal of Applied Mathematics* (2020).

[6] J. Vanualailai and S. Nakagiri, Stability of a system of Volterra Integro-Differential Equations, *Journal of Mathematical Analysis and Applications* **281:2**, 602–619 (2003).

[7] J. Biazar, H. Ghazvini and M. Eslami, He's Homotopy Perturbation Method for Systems of Integro-Differential Equations, *Chaos, Solitons and Fractals* **39:3**, 1253–1258 (2009).

[8] R. D. Driver, Existence and Stability of Solutions of a delay-Differential System, *Archive for Rational Mechanics and Analysis* **10:11**, 401–426 (1962).

[9] M. Gachpazan, Numerical Scheme to Solve Integro-Difierential Equations System, *Journal of Advanced Research in Scientific Computing* **1:1**, 11–21 (2009).

[10] K. Holmaker, Global Asymptotic Stability for A stationary Solution of A system of Integro-Differential Equations Describing the Formation of Liver Zones, *Siam j. Math. Anal* **24:1**, 116–128 (1993).

[11] T. Bo, L. Xie and X. Zheng, Numerical Approach to Wind Ripple in Desert, *International Journal of Nonlinear Sciences and Simulation* **8:2**, 223–228 (2007).

[12] A. Rahimi and E. Hesameddini, A new Numerical Scheme for Solving Systems of Integro-Differential Equations, *Computational Methods for Differential Equations* **1:2**, 108–119 (2013).

[13] J. Schiff, *The Laplace Transform: Theory and Applications* 1999.

[14] T. A. Burton, *Volterra Integral and Differential Equations*, Academic Press, New York (1983).

[15] A. Issa, *Approximate Solutions of Systems of Integro-Differential Equations*, An-Najah National University (2019).

[16] P. Agarwal, M. Akbar, R. Nawaz, M. Jleli, Solutions of System of Volterra Integro-Differential Equations Using Optimal Homotopy Asymptotic Method, *Mathematical Methods in the Applied Sciences* **44:3**, 2671–2681 (2021).

Author information

Ahmad Issa, Department of Mathematics, Karabük University, Karabük, Türkiye.
E-mail: ahmad93.issa18@gmail.com

Received: August 24, 2022.
Accepted: December 21, 2022.