Kirkuk University Journal /Scientific Studies (KUJSS)
Volume 11, Issue 3, September 2016 , p.p(15-25)
ISSN 1992 – 0849

Non-Polynomial Spline Method for the Solution of the System of two Nonlinear Volterra Integral Equations

Borhan F. Jumaa[1] , Abbas H. Taqi[2]

[1,2]Department of Mathematics / College of Science / University of Kirkuk

[1]borhan_nissan@yahoo.com

[2]dr.altaqi@yahoo.com

Received date : 18 / 11 / 2015 **Accepted date : 22 / 5 / 2016**

ABSTRACT

In this paper, using cubic non-polynomial spline method to solve the system of two nonlinear volterra integral equations of the second kind, we have used a matlab14 program to solve the system. Finally, several illustrative examples to show the effectiveness and accuracy of this method.

Keywords: *nonlinear integral equation, Volterra second kind, Non-polynomial spline.*

Kirkuk University Journal /Scientific Studies (KUJSS)
Volume 11, Issue 3, September 2016 , p.p(15-25)
ISSN 1992 – 0849

طريقة الثلمة غير متعددة الحدود لحل منظومة من معادلتين فولتيرا تكامليتين غير خطيتين

برهان فخرالدين جمعة[١] ، عباس حسن تقي[٢]

جامعة كركوك[١،٢] / كلية العلوم / قسم الرياضيات

[١]borhan_nissan@yahoo.com

[2]dr.altaqi@yahoo.com

تاريخ استلام البحث: ١٨ / ١١ / ٢٠١٥ **تاريخ قبول البحث: ٢٢ / ٥ / ٢٠١٦**

الملخص

في هذا البحث، استخدمنا طريقة الثلمة التكعيبية غير متعددة الحدود لحل منظومة مكونة من معادلتين فولتيرا التكاملية غير الخطية من النوع الثاني وقد استخدمنا برنامج matlab14 لحل النظام. وأخيرا، قدمنا العديد من الأمثلة التوضيحية لإظهار فعالية ودقة هذه الطريقة.

الكلمات الدالة: المعادلات التكاملية غير خطية، فولتيرا النوع الثاني، الثلمة غير متعددة الحدود.

1. INTRODUCTION

Many problems in mathematical physics, chemistry, biology, electrostatics, theory of elasticity, populations growth models and mixed problems of mechanics of continuous media reduce to a system of nonlinear integral equations of the second kind.[1,2]

The books edited by Wazwaz[3] and Linz[4] contain some different methods for solveley the system of nonlinear volterra integral equations analytically. Numerical methods also take an important place in solving the system of nonlinear integral equations. [5,6,7]

The cubic spline polynomial has the most application, because of the smoothness of the curve pieces and continuity of C^2. [7]

Kirkuk University Journal /Scientific Studies (KUJSS)
Volume 11, Issue 3, September 2016 , p.p(15-25)
ISSN 1992 – 0849

In this paper we solveley a system of two nonlinear Volterra integral equations of the second kind, the unknown functions appear inside and outside the integral sing of the form[8]

$$u(x) = f_1(x) + \int_0^x k_{11}(x,t,u(t))dt + \int_0^x k_{12}(x,t,v(t))dt \ldots\ldots(1)$$

$$v(x) = f_2(x) + \int_0^x k_{21}(x,t,u(t))dt + \int_0^x k_{22}(x,t,v(t))dt \ldots\ldots(2)$$

where $u(x), v(x)$ are unknown functions, the kernels $k_{11}(.,.,.), k_{12}(.,.,.), k_{21}(.,.,.), k_{22}(.,.,.)$ and the functions $f_1(x), f_2(x)$ are given real-valued functions on subsets of R^3 *and* R^1 respectively.

2. The Numerical Methods

For simplicity, we take an interval [a,b], in order to develop the numerical method for approximation solution of a system of the type (1),(2). For this purpose we define a grid of N+1 equally spaced points $x_i = a + ih, i = 0,1,\ldots,N$ where $h = \frac{b-a}{N+1}$. For each ith segment, the cubic non- polynomial spline $S_i(x)$ has the form [9]

$$S_i(x) = a_i sink(x - x_i) + b_i cosk(x - x_i) + c_i(x - x_i) + d_i \ldots\ldots(3)$$

where a_i, b_i, c_i, d_i are real finite constants and k is the frequency of the trigonometric functions which will be used to raise the accuracy of the method. To solve the equation (1) and (2) we rewrite the cubic non- polynomial spline $S_i(x)$ as the form

$$S_{1i}(x) = a_{1i} sink(x - x_i) + b_{1i} cosk(x - x_i) + c_{1i}(x - x_i) + d_{1i} \ldots\ldots(4)$$

$$S_{2i}(x) = a_{2i} sink(x - x_i) + b_{2i} cosk(x - x_i) + c_{2i}(x - x_i) + d_{2i} \ldots\ldots(5)$$

Let us consider the following relations:

$$S'_{1i}(x_i) = kb_{1i} + c_{1i} \approx u'(x_i)$$

$$S''_{1i}(x_i) = -k^2 a_{1i} \approx u''(x_i)$$

$$S'''_{1i}(x_i) = -k^3 b_{1i} \approx u'''(x_i)$$

$$S'_{2i}(x_i) = kb_{2i} + c_{2i} \approx v'(x_i)$$

$$S''_{2i}(x_i) = -k^2 a_{2i} \approx v''(x_i)$$

Kirkuk University Journal /Scientific Studies (KUJSS)
Volume 11, Issue 3, September 2016 , p.p(15-25)
ISSN 1992 – 0849

$S_{2i}'''(x_i) = -k^3 b_{2i} \approx v'''(x_i)$

We can obtain the values of a_{1i}, b_{1i}, c_{1i}, and d_{1i} as follows:

$$a_{1i} = -\frac{1}{k^2}u''(x_i) \qquad \ldots\ldots(6)$$

$$b_{1i} = -\frac{1}{k^3}u'''(x_i) \qquad \ldots\ldots(7)$$

$$c_{1i} = u'(x_i) + \frac{1}{k^2}u'''(x_i) \qquad \ldots\ldots(8)$$

$$d_{1i} = u(x_i) + \frac{1}{k}u''(x_i) \qquad \ldots\ldots(9)$$

And we can obtain the values of a_{2i}, b_{2i}, c_{2i}, and d_{2i} as follows:

$$a_{2i} = -\frac{1}{k^2}v''(x_i) \qquad \ldots\ldots(10)$$

$$b_{2i} = -\frac{1}{k^3}v'''(x_i) \qquad \ldots\ldots(11)$$

$$c_{2i} = v'(x_i) + \frac{1}{k^2}v'''(x_i) \qquad \ldots\ldots(12)$$

$$d_{2i} = v(x_i) + \frac{1}{k}v''(x_i) \qquad \ldots\ldots(13)$$

We differentiate equation (1) two times with respect to x and then put x=a, to get:

$$u_0 = f_1(a) \qquad \ldots\ldots(14)$$

$$u_0' = f_{10}' + k_{11}\big(0,0,u(0)\big) + k_{12}\big(0,0,v(0)\big) \qquad \ldots\ldots(15)$$

$$u_0'' = f_{10}'' + \left\{ \begin{array}{l} \left.\dfrac{\partial k_{11}\big(x,t,u(t)\big)}{\partial x}\right|_{t=x=0} + \left[\dfrac{\partial k_{11}\big(x,t,u(t)\big)}{\partial x}\right|_{x=0} \\ +k_{11}\big(0,0,u(0)\big)u'(0) \end{array} \right\}$$

$$+ \left\{ \begin{array}{l} \left.\dfrac{\partial k_{12}\big(x,t,v(t)\big)}{\partial x}\right|_{t=x=0} + \left[\dfrac{\partial k_{12}\big(x,t,v(t)\big)}{\partial x}\right|_{x=0} \\ +k_{12}\big(0,0,v(0)\big)v'(0) \end{array} \right\} \qquad \ldots\ldots(16)$$

Kirkuk University Journal /Scientific Studies (KUJSS)
Volume 11, Issue 3, September 2016 , p.p(15-25)
ISSN 1992 – 0849

$$u_0''' = f_{10}''' + \left\{ \begin{array}{l} \left.\frac{\partial k_{11}(x,t,u(t))}{\partial x}\right|_{t=x=0} + \frac{\partial}{\partial x}\left(\left[\left.\frac{\partial k_{11}(x,t,u(t))}{\partial x}\right|_{t=x}\right)\right|_{x=0} \\ + \left(\frac{\partial^2}{\partial x^2} k_{11}(x,t,u(t)) \Big|_{t=x}\right)\Big|_{x=0} + \frac{\partial}{\partial x^2}\, k_{11}(x,x,u(x)) \end{array} \right\}$$

$$+ \left\{ \begin{array}{l} \left.\frac{\partial k_{12}(x,t,v(t))}{\partial x}\right|_{t=x=0} + \frac{\partial}{\partial x}\left(\left[\left.\frac{\partial k_{12}(x,t,v(t))}{\partial x}\right|_{t=x}\right)\right|_{x=0} \\ + \left(\frac{\partial^2}{\partial x^2} k_{12}(x,t,v(t)) \Big|_{t=x}\right)\Big|_{x=0} + \frac{\partial}{\partial x^2}\, k_{12}(x,x,v(x)) \end{array} \right\} \quad \text{……..(17)}$$

We differentiate equation (2) two times with respect to x and then put x=a, to get:

$$v_0 = f_2(a) \quad \text{……………………………………………..(18)}$$

$$v_0' = f_{20}' + k_{21}(0,0,u(0)) + k_{22}(0,0,v(0)) \quad \text{………………………….(19)}$$

$$v_0'' = f_{20}'' + \left\{ \begin{array}{l} \left.\frac{\partial k_{21}(x,t,u(t))}{\partial x}\right|_{t=x=0} + \left[\left.\frac{\partial k_{21}(x,t,u(t))}{\partial x}\right|_{x=0} \\ +k_{21}(0,0,u(0))u'(0) \end{array} \right\}$$

$$+ \left\{ \begin{array}{l} \left.\frac{\partial k_{22}(x,t,v(t))}{\partial x}\right|_{t=x=0} + \left[\left.\frac{\partial k_{22}(x,t,v(t))}{\partial x}\right|_{x=0} \\ +k_{22}(0,0,v(0))v'(0) \end{array} \right\} \quad \text{………………….(20)}$$

$$v_0''' = f_{20}''' + \left\{ \begin{array}{l} \left.\frac{\partial k_{21}(x,t,u(t))}{\partial x}\right|_{t=x=0} + \frac{\partial}{\partial x}\left(\left[\left.\frac{\partial k_{21}(x,t,u(t))}{\partial x}\right|_{t=x}\right)\right|_{x=0} \\ + \left(\frac{\partial^2}{\partial x^2} k_{21}(x,t,u(t)) \Big|_{t=x}\right)\Big|_{x=0} + \frac{\partial}{\partial x^2}\, k_{21}(x,x,u(x)) \end{array} \right\}$$

Kirkuk University Journal /Scientific Studies (KUJSS)
Volume 11, Issue 3, September 2016 , p.p(15-25)
ISSN 1992 – 0849

$$+\left\{\begin{array}{c} \left.\frac{\partial k_{22}(x,t,v(t))}{\partial x}\right|_{t=x=0}+\frac{\partial}{\partial x}\left(\left[\left.\frac{\partial k_{22}(x,t,v(t))}{\partial x}\right|_{t=x}\right)\right|_{x=0} \\ +\left.\left(\frac{\partial^2}{\partial x^2}k_{22}(x,t,v(t))\,\Big|_{t=x}\right)\right|_{x=0}+\frac{\partial}{\partial x^2}\,k_{22}(x,x,v(x)) \end{array}\right\} \qquad \ldots\ldots\ldots(21)$$

Therefore, we approximate the solution of the system of two nonlinear Volterra integral equations of the second kind (1) using equations (4) and (2) using equation (5) in the following algorithm (NLVIENPS):

Algorithm (NLVIENPS)

Step 1: Set $h=\frac{b-a}{N}$; $x_i=x_0+ih$, $i=0,1,2,\ldots N$, $x_i=a$, $x_N=b$.

Step 2: 1- Evaluate a_{10}, b_{10}, c_{10}, and d_{10} by substituting 14-17 in equation 6-9.

2- Evaluate a_{20}, b_{20}, c_{20}, and d_{20} by substituting 18-21 in equation 10-13.

Step 3: 1-Calculate $S_{10}(x)$ using step2(1) and equation (4) for i=0.

2-Calculate $S_{20}(x)$ using step2(2) and equation (5) for i=0.

Step 4: Approximate $u_1 \approx S_{10}(x_1)$ and $v_1 \approx S_{20}(x_1)$.

Step 5: For i=1 to N-1 do the following Steps:

Step 6: 1- Evaluate a_{1i}, b_{1i}, c_{1i}, and d_{1i} using equation 6-9 and replacing

$u'(x_i), u''(x_i)$, and $u'''(x_i)$ in $S'_{1i}(x_i), S''_{1i}(x_i)$, and $S'''_{1i}(x_i)$.

2- - Evaluate a_{2i}, b_{2i}, c_{2i}, and d_{2i} using equation 10-13 and replacing
$v'(x_i), v''(x_i), and\ v'''(x_i)\ in\ S'_{2i}(x_i), S''_{2i}(x_i)$, and $S'''_{2i}(x_i)$.

Step 7: Approximate $u_{i+1}=S_{1i}(x_{i+1})$ and $v_{i+1}=S_{2i}(x_{i+1})$.

3. Numerical Example

The developed method in section (3) are illustrated in the following example:-

Example(1):

Consider the following system of two nonlinear Volterra integral equations of the second kind:

Kirkuk University Journal /Scientific Studies (KUJSS)
Volume 11, Issue 3, September 2016 , p.p(15-25)
ISSN 1992 – 0849

$$u(x) = x - \frac{2}{3}x^3 + \int_0^x \left(u^2(t) + v(t)\right)dt$$

$$v(x) = x^2 - \frac{1}{4}x^4 + \int_0^x u(t)v(t)dt$$

which has the exact solution: $\left(u(x), v(x)\right) = (x, x^2)$

Table (1): shows the comparison between the exact and the numerical solution via applying (**NLVIENPS**) algorithm respectively depends on the least square error (L.S.E.) which is defined as (L.S.E.=$\sum_{i=1}^{N}(Exact\ solution - Numerical\ solution)^2$), whene h=0.1, $x = x_i + ih, N = 10\ and\ i = 0,1, \dots ,10$.

x_i	$u_i(x)$	$v_i(x)$	$S_{u(x)}$	$S_{v(x)}$
0.0	0	0	0	0
0.1	0.100000000000000	0.010000000000000	0.100000000000000	0.009991669443949
0.2	0.200000000000000	0.040000000000000	0.200000000000000	0.039866844317517
0.3	0.300000000000000	0.090000000000000	0.300000000000000	0.089327021748788
0.4	0.400000000000000	0.160000000000000	0.400000000000000	0.157878011994230
0.5	0.500000000000000	0.250000000000000	0.500000000000000	0.244834876219255
0.6	0.600000000000000	0.360000000000000	0.600000000000000	0.349328770180644
0.7	0.700000000000000	0.490000000000000	0.700000000000000	0.470315625431024
0.8	0.800000000000000	0.640000000000000	0.800000000000000	0.606586581305670
0.9	0.900000000000000	0.810000000000000	0.900000000000000	0.756780063458672
1.0	1.000000000000000	1.00000000000000	1.000000000000000	0.919395388263721
L.S.E.			6.162975822039155e-032	0.010978923411819

Kirkuk University Journal /Scientific Studies (KUJSS)
Volume 11, Issue 3, September 2016 , p.p(15-25)
ISSN 1992 – 0849

Table (2): Presents the error for the numerical solution when h is changing

h	u(x)	v(x)
0.05	1.540743955509789e-032	4.483104770775418e-005
0.01	2.407412430484045e-034	1.164110663271072e-010

Example(2):

Consider the following system of two nonlinear Volterra integral equations of the second kind:

$$u(x) = \cos(x) - \frac{1}{2}x^2 + \int_0^x (x-t)\big(u^2(t) + v^2(t)\big)dt$$

$$v(x) = \sin(x) - \frac{1}{2}sin^2(x) + \int_0^x (x-t)(u^2(t) - v^2(t))dt$$

which has the exact solution: $\big(u(x), v(x)\big) = (\cos(x), \sin(x))$

Table (3): shows the comparison between the exact and the numerical solution via applying (**NLVIENPS**) algorithm respectively depends on the least square error (L.S.E.), whene h=0.1, $x = x_i + ih, N = 10\ and\ i = 0,1, \dots, 10.$

x_i	$u_i(x)$	$v_i(x)$	$S_{u(x)}$	$S_{v(x)}$
0.0	1.0000000000000	0	1.000000000000000	0
0.1	0.995004165278026	0.099833416646828	0.995004165278026	0.099833416646828
0.2	0.980066577841242	0.198669330795061	.980066577841242	0.198669330795061
0.3	0.955336489125606	0.295520206661340	0.955336489125606	0.295520206661340
0.4	0.921060994002885	0.389418342308651	0.921060994002885	0.389418342308650
0.5	0.877582561890373	0.479425538604203	0.877582561890373	0.479425538604203
0.6	0.825335614909678	0.564642473395035	0.825335614909678	0.564642473395035
0.7	0.764842187284488	0.644217687237691	0.764842187284488	0.644217687237691
0.8	0.696706709347165	0.71735609899523	0.696706709347165	0.717356090899523
0.9	0.621609968270664	0.783326909627483	0.621609968270664	0.783326909627483
1.0	0.540302305868140	0.84147098480789	0.540302305868139	0.84147098480789
L.S.E.			6.03971630559e-031	4.83793602030e-031

Kirkuk University Journal /Scientific Studies (KUJSS)
Volume 11, Issue 3, September 2016 , p.p(15-25)
ISSN 1992 – 0849

Table (4): Presents the error for the numerical solution when h is changing

h	u(x)	v(x)
0.05	4.560602108308975e-031	4.468157470978387e-032
0.01	6.162975822039155e-032	2.407412430484045e-034

Example(3):

Consider the following the system of two nonlinear Volterra integral equations of the second kind:

$$u(x) = e^x + x - \frac{1}{2}\sinh(2x) + \int_0^x (x-t)\big(u^2(t) - v^2(t)\big)dt$$

$$v(x) = e^{-x} + x - xe^x + \int_0^x (xu^2(t)v(t))dt$$

which has the exact solution: $\big(u(x), v(x)\big) = (e^x, e^{-x})$

Table (5): shows the comparison between the exact and the numerical solution via applying (**NLVIENPS**) algorithm respectively depends on the least square error (L.S.E.), whene h=0.1, $x = x_i + ih, N = 10\ and\ i = 0,1, \dots, 10.$

X_i	$u_i(x)$	$v_i(x)$	$S_{u(x)}$	$S_{v(x)}$
0.0	1.000000000000000	1.000000000000000	1.000000000000000	1.000000000000000
0.1	1.105170918075648	0.904837418035960	1.105162418075146	0.904829251368803
0.2	1.221402758160170	0.818730753077982	1.221264091363697	0.818602752953820
0.3	1.349858807576003	0.740818220681718	1.349143304213055	0.740183717535734
0.4	1.491824697641270	0.670320046035639	1.489520663688465	0.668357348305765
0.5	1.648721270700128	0.606530659712633	1.642991899505425	0.601842976713830
0.6	1.822118800390509	0.548811636094026	1.810021911695287	0.539306858485357
0.7	2.013752707470477	0.496585303791409	1.990940125477822	0.479375499953203
0.8	2.225540928492468	0.449328964117222	2.185937199753314	0.420649381552357
0.9	2.459603111156950	0.406569659740599	2.395063122101854	0.361716941356819
1.0	2.718281828459046	0.367879441171442	2.618226709323966	0.301168678939757
L.S.E.			0.016450306508223	0.007697374344827

Kirkuk University Journal /Scientific Studies (KUJSS)
Volume 11, Issue 3, September 2016 , p.p(15-25)
ISSN 1992 – 0849

Table (6): Presents the error for the numerical solution when h is changing

h	u(x)	v(x)
0.05	5.444895358291389e-005	3.736582216275992e-005
0.01	1.208946021180994e-010	1.121484626531558e-010

4. Conclusions

Non-polynomial cubic spline method has been presented for solving the system of two nonlinear Volterra integral equations of the second kind numerically. The results show a marked improvement in the least square errors (L.S.E.) and the following points are included:

1. Non-polynomial cubic spline method give better accuracy to solution of the system of two nonlinear Volterra integral equations of the second kind.

2. The good approximation results depend on the size of h, if h is decreased then the number of points increases and the L.S.E. approaches Zero.

References

[1] J.R. Rise, "***Numerical Method Software and Analysis***", macgra Hill, 1985.

[2] G.Ch. Class of Bezier, "***Aided Geom, Design***", p.p.20,29,39, 2003.

[3] A. M. Wazwaz , "***Linear and Nonlinear Integral Equation***", Methods and Applications, Springer, 2011.

[4] P. Linz, "***Analytical and numerical methods for Volterra Equations***", studies in Applied Mathematics, SIAM, Philadelphia, 1985.

[5] N. M.Saba, "***Solution of Second Kind Volterra Integro Equations using Linear Non-polynomial Spline Function***", Mathematical Theory and Modeling, vol.4, No.9, 2014.

[6] p. A. Hossin, "***The Solve of Integral Differential Equation by Non-Polynomial Spline Function and Quadrature Formula***", International Conference on Applied mathematics and Pharmaceutical Science, Jan.7-8, P.595-597, 2012.

[7] S. Al-Khalidi and M. M. Mustafa, "***Algorithms for Solving Volterra Integral Equations using Non-Polynomial Spline Functions***", MSC. Thises, College of Science Women, University of Baghdad, 2013.

Volume 11, Issue 3, September 2016 , p.p(15-25)
ISSN 1992 – 0849

[8] A. M. Dalal , "***Numerical Solution of System of Two Nonlinear Volterra Integral Equations***", International Journal of Computers and Technology, Vol.12, No.10, P.3967-3975, 2014.

[9] H. Brunner, "***Non-Polynomial Spline Collection for Volterra Equation with weakly singular kernels***", SIAM Journal on Numerical Analysis, 20(6), 1106-1119, 1983.

AUTHOR

Borhan F. Jumaa: BSc. Salahaddin University MSc. University Technology Prof. University Technology , Appled Mathematics General Specialty , Mathematics and Computer Applications Accurate Specialty , Scientific Grade Prof Assistant.