



# Comparison Between Adomain Decomposition Method and Numerical Solutions of Linear Volterra Integral Equations of the Second Kind by Using the Fifth Order of Non-Polynomial Spline Functions

Elgaili Abdalla Elhassan Ibrahim<sup>1</sup>, Abdel Radi Abdel Rahman Abdel Gadir Abdel Rahman<sup>2,\*</sup>,  
Neama Yahia Mohammed<sup>3</sup>, Nageeb Abdallah Hamed Haroun<sup>2</sup>

<sup>1</sup>Department of Mathematics, Faculty of Science, Sudan University for Science and Technology, Khartoum, Sudan

<sup>2</sup>Department of Mathematics, Faculty of Education, Omdurman Islamic University, Omdurman, Sudan

<sup>3</sup>Department of Mathematics, College of Science, Tabuk University, Tabuk, Saudi Arabia

## Email address:

dibdelradi78@gmail.com (A. R. A. R. A. G. A. Rahman)

\*Corresponding author

## To cite this article:

Elgaili Abdalla Elhassan Ibrahim, Abdel Radi Abdel Rahman Abdel Gadir Abdel Rahman, Neama Yahia Mohammed, Nageeb Abdallah Hamed Haroun. Comparison Between Adomain Decomposition Method and Numerical Solutions of Linear Volterra Integral Equations of the Second Kind by Using the Fifth Order of Non-Polynomial Spline Functions. *International Journal of Applied Mathematics and Theoretical Physics*. Vol. 7, No. 3, 2021, pp. 68-79. doi: 10.11648/j.ijamtp.20210703.12

Received: August 11, 2021; Accepted: August 23, 2021; Published: August 31, 2021

---

**Abstract:** Volterra integral equations are a special type of integrative equations; they are divided into two categories referred to as the first and second type. This paper will deal with the second type which has wide range of the applications in physics and engineering problems. Spline functions are piece-wise polynomials of degree  $n$  joined together at the break points with  $n-1$  continuous derivatives. The break points of splines are called Knot, spline function can be integrated and differentiated due to being piece wise polynomials and can easily store and implemented on digital computer, non-polynomial spline function apiece wise is a blend of trigonometric, as well as, polynomial basis function, which form a complete extended Chebyshev space. Matlab is a powerful computing system for handling the calculations involved scientific and engineering problems. The aim of this paper is to compare between Adomain decomposition method and numerical solution to solve Volterra Integral Equations of second kind using the fifth order non-polynomial Spline functions by Matlab. We followed the applied mathematical method numerically by Matlab. Numerical examples are presented to illustrate the applications of this methods and to compare the computed results with analytical solutions. Finally by comparison of numerical results, Simplicity and efficiency of this method be shown.

**Keywords:** Linear Volterra Integral Equations, Second Kind, Non-Polynomial Spline Functions, Fifth Order, Adomain Decomposition Method

---

## 1. Introduction

Recently there is increase in concern by integrated equations, particularly Volterra integral equations has wide range of the applications in physics and engineering such as potential theory, Dirichlet problems, electrostatics, the particle transport problems of astrophysics, reactor problems, contact problems, diffusion problems and heat transfer problems(for more details see [4]).

In the mid of 1960s the numerical solution of integral

equations was begun when the Kernel is unwell

In the beginning of 1980s the numerical methods began to take more importance when there is no analytical solution.

In 1997 Diogo, T and Lima, B presented deductive method [6], in order to find a numerical solution for Volterra integral equations with weakly singular kernel.

In 2007, Danaf & Ramadan [11] presented applications of non-polynomial spline functions to solve Volterra equations.

In 2009 Haq [2] studied the numerical solution to the boundary value problem and the initial value problem using

the Spline function.

In the year 2011, Zarebnia - Hoshyar - Sedahti [17] presented the numerical solution based on the Cube Spline function to find a solution to the boundary value problem.

In 2011 Majeed, S [8] presented a numerical solution to the Volterra differential integrative equation of the second kind using non-polynomial Spline functions.

In 2012 Hossinpour presented [5] a solution for differential integral equations by using non-polynomial spline functions.

In 2015, Harbis, S, Murad, M and Majed, S [3] presented a numerical solution for linear Volterra integral equation from second kind by using non-polynomial spline function from the third degree.

In the year 2016; Taqi. A, Jumma. B [15] Method of Non-Boundary Spline Functions to Solve the System of Two Nonlinear Volterra Integral Equations.

In 2017, Najwa, S and Mohammed, S [7] presented a numerical solution for linear Volterra integral equations with weakly singular kernel by using non-polynomial spline function from the fifth degree.

In this paper we introduced a numerical solution for linear Volterra integral equation from second kind with by using non-polynomial spline function from the fifth degree by Matlab.

## 2. Analytical Solution of Linear Volterra Integral Equations of the Second Kind: [1, 9, 16]

First we define Volterra integral equations of the second kind are given by:

$$u(x) = f(x) + \int_a^x k(x, t)u(t)dt \quad 0 \leq x \leq b \quad (1)$$

The unknown function  $u(x)$  that determined, occurs inside and outside the integral sign, The Kernel  $k(x, t)$  and the function  $f(x)$  are given continues functions. [1, 9, 16]

There are many analytical methods available for solving Linear Volterra Integral Equations of second kind. We focus on the follow method: the Adomain decomposition method:

## 3. Adomain Decomposition Method: [1, 9, 16]

The Adomain decomposition method (ADM) was introduced and developed by George Adomain. The Adomain decomposition method consists of decomposition the unknown function  $u(x)$  of any equation into a sum of infinite number of components defined by the decompositions series

$$u(x) = \sum_{n=0}^{\infty} u_n(x) \quad (2)$$

Or equivalently

$$u(x) = u_0(x) + u_1(x) + u_2(x) + \dots \quad (3)$$

Where the components  $(u_n(x), n \geq 0)$  are to be determined in a recursive manner. The decomposition method

concerns itself with finding the components individually; we substitute (2) into the Volterra integral equation (1) to obtain

$$\sum_{n=0}^{\infty} u_n(x) = f(x) + \int_a^x k(x, t)(\sum_{n=0}^{\infty} u_n(t))dt \quad (4)$$

The zeroth components  $u_0(x)$  is identified by all items that are not included under the integral sign. Consequently, the components  $u_j(x), j \geq 1$  of the unknown function  $u(x)$  is completely determined by setting the recurrence relation:

$$u_0(x) = f(x_0) \quad (5)$$

$$u_{n+1}(x) = \int_a^x k(x, t)u_n(t)dt, n \geq 0 \quad (6)$$

*Example (1): [1, 9, 16]*

To solve the following Volterra integral equation:

$$u(x) = 1 - \int_0^x u(t)dt \quad (7)$$

Where  $f(x) = 1$  and  $k(x, t) = -1$ ,

Substituting decomposition series (2) into the both two sides of VIE's (12) gives

$$\sum_{n=0}^{\infty} u_n(x) = 1 - \int_0^x \sum_{n=0}^{\infty} u_n(t)dt \quad (8)$$

We identify the zeroth components by all terms that are not included under the integral sign. Therefore, we obtain the following recurrence relation:

$$u_0(x) = 1,$$

$$u_{n+1}(x) = - \int_0^x u_n(t)dt, k \geq 0 \quad (9)$$

So that  $u_0(x) = 1$ ,

$$u_1(x) = - \int_0^x u_0(t)dt = - \int_0^x 1dt, = -x,$$

$$u_2(x) = - \int_0^x u_1(t)dt = - \int_0^x -tdt, = \frac{x^2}{2!},$$

$$u_3(x) = - \int_0^x u_2(t)dt = - \int_0^x \frac{t^2}{2}dt, = -\frac{x^3}{3!},$$

$$u_4(x) = - \int_0^x u_3(t)dt = - \int_0^x -\frac{t^3}{6}dt, = \frac{x^4}{4!}, \forall \text{ equations } (10)$$

And so on, gives the series solution

$$u(x) = 1 - x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots = e^{-x} \quad (11)$$

which is equation (11) the exact solution for equation (7)

## 4. A Numerical Solution of Linear VIE's of 2nd Kind Using the 5th Order of Non-Polynomial Spline Functions

The form of the five non-polynomial spline function is:

$$P_i(t) = a_i \cos K(t - t_i) + b_i \sin K(t - t_i) + c_i(t - t_i) + d_i(t - t_i)^2 + e_i(t - t_i)^3 + r_i(t - t_i)^4 + z_i(t - t_i)^5 + g_i \quad (12)$$

where  $a_i, b_i, c_i, d_i, e_i, r_i, z_i$ , and  $g_i$  are constants, to be determined. In order to obtain the values of  $a_i, b_i, c_i, d_i, e_i, r_i, z_i$ , and  $g_i$ , we differentiate equation (12) seven times with respect to  $t$ , and we get the following equations

$$\left. \begin{aligned} P_i^{(1)}(t) &= -Ka_i \sin K(t - t_i) + Kb_i \cos K(t - t_i) + c_i + 2d_i(t - t_i) + 3e_i(t - t_i)^2 \\ &\quad + 4r_i(t - t_i)^3 + 5z_i(t - t_i)^4 \\ P_i^{(2)}(t) &= -K^2 a_i \cos K(t - t_i) - K^2 b_i \sin K(t - t_i) + 2d_i + 6e_i(t - t_i) \\ &\quad + 12r_i(t - t_i)^2 + 20z_i(t - t_i)^3 \\ P_i^{(3)}(t) &= K^3 a_i \sin K(t - t_i) - K^3 b_i \cos K(t - t_i) + 6e_i + 24r_i(t - t_i) + 60z_i(t - t_i)^2 \\ P_i^{(4)}(t) &= K^4 a_i \cos K(t - t_i) + K^4 b_i \sin K(t - t_i) + 24r_i + 120z_i(t - t_i) \\ P_i^{(5)}(t) &= -K^5 a_i \sin K(t - t_i) + K^5 b_i \cos K(t - t_i) + 120z_i \\ P_i^{(6)}(t) &= -K^6 a_i \cos K(t - t_i) - K^6 b_i \sin K(t - t_i) \\ P_i^{(7)}(t) &= K^7 a_i \sin K(t - t_i) - K^7 b_i \cos K(t - t_i) \end{aligned} \right\} \quad (13)$$

Hence replace  $t$  by  $t_i$  in the relation (12) and (13) yields:

$$d_i = \frac{1}{2} [P_i^{(2)}(t) + K^2 a_i] \quad (17)$$

$$P(t) = a_i + g_i$$

$$e_i = \frac{1}{6} [P_i^{(3)}(t) + K^3 b_i] \quad (18)$$

$$P_i^{(1)}(t) = b_i K + c_i$$

$$r_i = \frac{1}{24} [P_i^{(4)}(t) - K^4 a_i] \quad (19)$$

$$P_i^{(2)}(t) = -K^2 a_i + 2d_i$$

$$z_i = \frac{1}{120} [P_i^{(5)}(t) - K^5 b_i] \quad (20)$$

$$P_i^{(3)}(t) = -K^3 b_i + 6e_i$$

$$g_i = P_i(t) - a_i \quad (21)$$

$$P_i^{(4)}(t) = K^4 a_i + 24r_i$$

for  $i = 0, 1, \dots, n$

$$P_i^{(5)}(t) = K^5 a_i + 120z_i$$

$$P_i^{(6)}(t) = -K^6 a_i$$

$$P_i^{(7)}(t) = -K^7 b_i$$

We obtained the values of  $a_i, b_i, c_i, d_i, e_i, r_i, z_i$ , and  $g_i$  from the above relations as follows:

$$a_i = -\frac{1}{K^6} P_i^{(6)}(t) \quad (14)$$

$$b_i = \frac{1}{K^7} P_i^{(7)}(t) \quad (15)$$

$$c_i = P_i^{(1)}(t) - Kb_i \quad (16)$$

## 5. Linear Volterra Integral Equations of the Second Kind [1, 9, 16]

First we define Volterra integral equations of the second kind are given by:

$$u(x) = f(x) + \int_a^x k(x, t) u(t) dt \quad 0 \leq x \leq b \quad (22)$$

To solve the equation (22). We need to differentiate equation (22) seven times with respect to  $x$ , by using Libenze formula we realize:

$$u^{(1)}(x) = f^{(1)}(x) + \int_a^x \frac{\partial k(x, t)}{\partial x} u(t) dt + k(x, x) u(x) \quad (23)$$

$$u^{(2)}(x) = f^{(2)}(x) + \int_a^x \frac{\partial^2 k(x, t)}{\partial x^2} u(t) dt + \left( \frac{\partial k(x, t)}{\partial x} \right)_{t=x} u(x) + \frac{dk(x, x)}{dx} u(x) + k(x, x) u^{(1)}(x) \quad (24)$$

$$u^{(3)}(x) = f^{(3)}(x) + \int_a^x \frac{\partial^3 k(x, t)}{\partial x^3} u(t) dt + \left( \frac{\partial^2 k(x, t)}{\partial x^2} \right)_{t=x} u(x) + \frac{d}{dx} \left( \frac{\partial k(x, t)}{\partial x} \right)_{t=x} u(x) + \left( \frac{\partial k(x, t)}{\partial x} \right)_{t=x} u^{(1)}(x) + \frac{d^{(2)} k(x, x)}{dx^2} u(x) + 2 \frac{dk(x, x)}{dx} u^{(1)}(x) + k(x, x) u^{(2)}(x) \quad (25)$$

$$u^{(4)}(x) = f^{(4)}(x) + \int_a^x \frac{\partial^4 k(x, t)}{\partial x^4} u(t) dt + \left( \frac{\partial^3 k(x, t)}{\partial x^3} \right)_{t=x} u(x) + \frac{d}{dx} \left( \frac{\partial^2 k(x, t)}{\partial x^2} \right)_{t=x} u(x) + \left( \frac{\partial^2 k(x, t)}{\partial x^2} \right)_{t=x} u^{(1)}(x) + \frac{d^{(2)} k(x, x)}{dx^2} \left( \frac{\partial k(x, t)}{\partial x} \right)_{t=x} u(x) + 2 \frac{d}{dx} \left( \frac{\partial k(x, t)}{\partial x} \right)_{t=x} u^{(1)}(x) + \left( \frac{\partial k(x, t)}{\partial x} \right)_{t=x} u^{(2)}(x) + \frac{d^{(3)} k(x, x)}{dx^3} u(x) + 3 \frac{dk(x, x)}{dx} u^{(1)}(x) + 3 \frac{dk(x, x)}{dx} u^{(2)}(x) + k(x, x) u^{(3)}(x) \quad (26)$$

$$u^{(5)}(x) = f^{(5)}(x) + \int_a^x \frac{\partial^{(5)}k(x,t)}{\partial x^{(5)}} u(t) dt + \left( \frac{\partial^{(4)}k(x,t)}{\partial x^{(4)}} \right)_{t=x} u(x) + \frac{d}{dx} \left( \frac{\partial^{(3)}k(x,t)}{\partial x^{(3)}} \right)_{t=x} u(x) + \left( \frac{\partial^{(3)}k(x,t)}{\partial x^{(3)}} \right)_{t=x} u^{(1)}(x) + \frac{d^{(2)}}{dx^{(2)}} \left( \frac{\partial^{(2)}k(x,t)}{\partial x^{(2)}} \right)_{t=x} u(x) + 2 \frac{d}{dx} \left( \frac{\partial^{(2)}k(x,t)}{\partial x^{(2)}} \right)_{t=x} u^{(1)}(x) + \left( \frac{\partial^{(2)}k(x,t)}{\partial x^{(2)}} \right)_{t=x} u^{(2)}(x) + \frac{d^{(3)}}{dx^{(3)}} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} u(x) + 3 \frac{d^{(2)}}{dx^{(2)}} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} u^{(1)}(x) + 3 \frac{d}{dx} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} u^{(2)}(x) + \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} u^{(3)}(x) + \frac{d^{(4)}k(x,x)}{dx^{(4)}} u(x) + 4 \frac{d^{(3)}k(x,x)}{dx^{(3)}} u^{(1)}(x) + 6 \frac{d^{(2)}k(x,x)}{dx^{(2)}} u^{(2)}(x) + 4 \frac{dk(x,x)}{dx} u^{(3)}(x) + k(x,x) u^{(4)}(x) \quad (27)$$

$$u^{(6)}(x) = f^{(6)}(x) + \int_a^x \frac{\partial^{(6)}k(x,t)}{\partial x^{(6)}} u(t) dt + \left( \frac{\partial^{(5)}k(x,t)}{\partial x^{(5)}} \right)_{t=x} u(x) + \frac{d}{dx} \left( \frac{\partial^{(4)}k(x,t)}{\partial x^{(4)}} \right)_{t=x} u(x) + \left( \frac{\partial^{(4)}k(x,t)}{\partial x^{(4)}} \right)_{t=x} u^{(1)}(x) + \frac{d}{dx} \left( \frac{\partial^{(3)}k(x,t)}{\partial x^{(3)}} \right)_{t=x} u^{(1)}(x) + 2 \frac{d}{dx} \left( \frac{\partial^{(3)}k(x,t)}{\partial x^{(3)}} \right)_{t=x} u^{(1)}(x) + \left( \frac{\partial^{(3)}k(x,t)}{\partial x^{(3)}} \right)_{t=x} u^{(2)}(x) + \frac{d^{(3)}}{dx^{(3)}} \left( \frac{\partial^{(2)}k(x,t)}{\partial x^{(2)}} \right)_{t=x} u(x) + 3 \frac{d^{(2)}}{dx^{(2)}} \left( \frac{\partial^{(2)}k(x,t)}{\partial x^{(2)}} \right)_{t=x} u^{(1)}(x) + 3 \frac{d}{dx} \left( \frac{\partial^{(2)}k(x,t)}{\partial x^{(2)}} \right)_{t=x} u^{(2)}(x) + \left( \frac{\partial^{(2)}k(x,t)}{\partial x^{(2)}} \right)_{t=x} u^{(3)}(x) + \frac{d^{(4)}}{dx^{(4)}} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} u(x) + 4 \frac{d^{(3)}}{dx^{(3)}} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} u^{(1)}(x) + 6 \frac{d^{(2)}}{dx^{(2)}} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} u^{(2)}(x) + 4 \frac{d}{dx} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} u^{(3)}(x) + \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} u^{(4)}(x) + \frac{d^{(5)}k(x,x)}{dx^{(5)}} u(x) + 5 \frac{d^{(4)}k(x,x)}{dx^{(4)}} u^{(1)}(x) + 10 \frac{d^{(3)}k(x,x)}{dx^{(3)}} u^{(2)}(x) + 10 \frac{d^{(2)}k(x,x)}{dx^{(2)}} u^{(3)}(x) + 5 \frac{dk(x,x)}{dx} u^{(4)}(x) + k(x,x) u^{(5)}(x) \quad (28)$$

$$u^{(7)}(x) = f^{(7)}(x) + \int_a^x \frac{\partial^{(7)}k(x,t)}{\partial x^{(7)}} u(t) dt + \left( \frac{\partial^{(6)}k(x,t)}{\partial x^{(6)}} \right)_{t=x} u(x) + \frac{d}{dx} \left( \frac{\partial^{(5)}k(x,t)}{\partial x^{(5)}} \right)_{t=x} u(x) + \left( \frac{\partial^{(5)}k(x,t)}{\partial x^{(5)}} \right)_{t=x} u^{(1)}(x) + \frac{d^2}{dx^2} \left( \frac{\partial^{(4)}k(x,t)}{\partial x^{(4)}} \right)_{t=x} u(x) + 2 \frac{d}{dx} \left( \frac{\partial^{(4)}k(x,t)}{\partial x^{(4)}} \right)_{t=x} u^{(1)}(x) + \left( \frac{\partial^{(4)}k(x,t)}{\partial x^{(4)}} \right)_{t=x} u^{(2)}(x) + \frac{d^{(3)}}{dx^{(3)}} \left( \frac{\partial^{(3)}k(x,t)}{\partial x^{(3)}} \right)_{t=x} u(x) + 3 \frac{d^{(2)}}{dx^{(2)}} \left( \frac{\partial^{(3)}k(x,t)}{\partial x^{(3)}} \right)_{t=x} u^{(1)}(x) + 3 \frac{d}{dx} \left( \frac{\partial^{(3)}k(x,t)}{\partial x^{(3)}} \right)_{t=x} u^{(2)}(x) + \frac{d^{(4)}}{dx^{(4)}} \left( \frac{\partial^{(2)}k(x,t)}{\partial x^{(2)}} \right)_{t=x} u(x) + 4 \frac{d^{(3)}}{dx^{(3)}} \left( \frac{\partial^{(2)}k(x,t)}{\partial x^{(2)}} \right)_{t=x} u^{(1)}(x) + 6 \frac{d^{(2)}}{dx^{(2)}} \left( \frac{\partial^{(2)}k(x,t)}{\partial x^{(2)}} \right)_{t=x} u^{(2)}(x) + 4 \frac{d}{dx} \left( \frac{\partial^{(2)}k(x,t)}{\partial x^{(2)}} \right)_{t=x} u^{(3)}(x) + \left( \frac{\partial^{(3)}k(x,t)}{\partial x^{(3)}} \right)_{t=x} u^{(3)}(x) + \left( \frac{\partial^{(2)}k(x,t)}{\partial x^{(2)}} \right)_{t=x} u^{(4)}(x) + \left( \frac{d^{(5)}}{dx^{(5)}} \left( \frac{\partial k(x,t)}{\partial x} \right) \right)_{t=x} u(x) + 5 \frac{d^{(4)}}{dx^{(4)}} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} u^{(1)}(x) + 10 \frac{d^{(3)}}{dx^{(3)}} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} u^{(2)}(x) + 10 \frac{d^{(2)}}{dx^{(2)}} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} u^{(3)}(x) + 5 \frac{d}{dx} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} u^{(4)}(x) + \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} u^{(5)}(x) + \frac{d^{(6)}k(x,x)}{dx^{(6)}} u(x) + 6 \frac{d^{(5)}k(x,x)}{dx^{(5)}} u^{(1)}(x) + 15 \frac{d^{(4)}k(x,x)}{dx^{(4)}} u^{(2)}(x) + 20 \frac{d^{(3)}k(x,x)}{dx^{(3)}} u^{(3)}(x) + 15 \frac{d^{(2)}k(x,x)}{dx^{(2)}} u^{(4)}(x) + 6 \frac{dk(x,x)}{dx} u^{(5)}(x) + k(x,x) u^{(6)}(x) \quad (29)$$

To complete our procedure for solving VIE's we substitute  $x = a$  in equations (22)-(29) then we get:

$$u_0 = u(a) = f(a) \quad (30)$$

$$u_0^{(1)} = u^{(1)}(a) = f^{(1)}(a) + k(a,a).u(a) \quad (31)$$

$$u_0^{(2)} = u^{(2)}(a) = f^{(2)}(a) + \left( \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u(a) + \left( \frac{dk(x,x)}{dx} \right)_{x=a} u(a) + k(a,a)u^{(1)}(a) \quad (32)$$

$$u_0^{(3)} = u^{(3)}(a) = f^{(3)}(a) + \left( \left( \frac{\partial^{(2)}k(x,t)}{\partial x^{(2)}} \right)_{t=x} \right)_{x=a} u(a) + \left( \frac{d}{dx} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u(a) + \left( \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u^{(1)}(a) + \left( \frac{d^{(2)}k(x,x)}{dx^{(2)}} \right)_{x=a} u(a) + 2 \left( \frac{dk(x,x)}{dx} \right)_{x=a} u^{(1)}(a) + k(x,x)u^{(2)}(a) \quad (33)$$

$$u_0^{(4)} = u^{(4)}(a) = f^{(4)}(a) + \left( \left( \frac{\partial^{(3)}k(x,t)}{\partial x^{(3)}} \right)_{t=x} \right)_{x=a} u(a) + \left( \frac{d}{dx} \left( \frac{\partial^{(2)}k(x,t)}{\partial x^{(2)}} \right)_{t=x} \right)_{x=a} u(a) + \left( \left( \frac{\partial^{(2)}k(x,t)}{\partial x^{(2)}} \right)_{t=x} \right)_{x=a} u^{(1)}(a) + \left( \frac{d^{(2)}}{dx^{(2)}} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u(a) + 2 \left( \frac{d}{dx} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u^{(1)}(a) + \left( \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u^{(2)}(a) + \left( \frac{d^{(3)}k(x,x)}{dx^{(3)}} \right)_{x=a} u(a) + 3 \left( \frac{d^{(2)}k(x,x)}{dx^{(2)}} \right)_{x=a} u^{(1)}(a) + 3 \left( \frac{dk(x,x)}{dx} \right)_{x=a} u^{(2)}(a) + k(x,x)u^{(3)}(a) \quad (34)$$

$$u_0^{(5)} = u^{(5)}(a) = f^{(5)}(a) + \left( \left( \frac{\partial^{(4)}k(x,t)}{\partial x^{(4)}} \right)_{t=x} \right)_{x=a} u(a) + \left( \frac{d}{dx} \left( \frac{\partial^{(3)}k(x,t)}{\partial x^{(3)}} \right)_{t=x} \right)_{x=a} u(a) + \left( \left( \frac{\partial^{(3)}k(x,t)}{\partial x^{(3)}} \right)_{t=x} \right)_{x=a} u^{(1)}(a) + \left( \frac{d^{(2)}}{dx^{(2)}} \left( \frac{\partial^{(2)}k(x,t)}{\partial x^{(2)}} \right)_{t=x} \right)_{x=a} u(a) + 2 \left( \frac{d}{dx} \left( \frac{\partial^{(2)}k(x,t)}{\partial x^{(2)}} \right)_{t=x} \right)_{x=a} u^{(1)}(a) + \left( \left( \frac{\partial^{(2)}k(x,t)}{\partial x^{(2)}} \right)_{t=x} \right)_{x=a} u^{(2)}(a) + \left( \frac{d^{(3)}}{dx^{(3)}} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u(a) + 3 \left( \frac{d^{(2)}}{dx^{(2)}} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u^{(1)}(a) + 3 \left( \frac{d}{dx} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u^{(2)}(a) + \left( \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u^{(3)}(a) + \left( \frac{d^{(4)}k(x,x)}{dx^{(4)}} \right)_{x=a} u(a) + 4 \left( \frac{d^{(3)}k(x,x)}{dx^{(3)}} \right)_{x=a} u^{(1)}(a) + 6 \left( \frac{d^{(2)}k(x,x)}{dx^{(2)}} \right)_{x=a} u^{(2)}(a) +$$

$$4 \left( \frac{dk(x,x)}{dx} \right)_{x=a} u^{(3)}(a) + k(x,x) u^{(4)}(a) \quad (35)$$

$$\begin{aligned} u_0^{(6)} = u^{(6)}(a) = f^{(6)}(a) &+ \left( \left( \frac{\partial^{(5)} k(x,t)}{\partial x^{(5)}} \right)_{t=x} \right)_{x=a} u(a) + \left( \frac{d}{dx} \left( \frac{\partial^{(4)} k(x,t)}{\partial x^{(4)}} \right)_{t=x} \right)_{x=a} u(a) + \left( \left( \frac{\partial^{(4)} k(x,t)}{\partial x^{(4)}} \right)_{t=x} \right)_{x=a} u^{(1)}(a) + \\ &\left( \frac{d^{(2)}}{dx^{(2)}} \left( \frac{\partial^{(3)} k(x,t)}{\partial x^{(3)}} \right)_{t=x} \right)_{x=a} u(a) + 2 \left( \frac{d}{dx} \left( \frac{\partial^{(3)} k(x,t)}{\partial x^{(3)}} \right)_{t=x} \right)_{x=a} u^{(1)}(a) + \left( \left( \frac{\partial^{(3)} k(x,t)}{\partial x^{(3)}} \right)_{t=x} \right)_{x=a} u^{(2)}(a) + \\ &\left( \frac{d^{(3)}}{dx^{(3)}} \left( \frac{\partial^{(2)} k(x,t)}{\partial x^{(2)}} \right)_{t=x} \right)_{x=a} u(a) + 3 \left( \frac{d^{(2)}}{dx^{(2)}} \left( \frac{\partial^{(2)} k(x,t)}{\partial x^{(2)}} \right)_{t=x} \right)_{x=a} u^{(1)}(a) + 3 \left( \frac{d}{dx} \left( \frac{\partial^{(2)} k(x,t)}{\partial x^{(2)}} \right)_{t=x} \right)_{x=a} u^{(2)}(a) + \\ &\left( \left( \frac{\partial^{(2)} k(x,t)}{\partial x^{(2)}} \right)_{t=x} \right)_{x=a} u^{(3)}(a) + \left( \frac{d^{(4)}}{dx^{(4)}} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u(a) + 4 \left( \frac{d^{(3)}}{dx^{(3)}} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u^{(1)}(a) + \\ &6 \left( \frac{d^{(2)}}{dx^{(2)}} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u^{(2)}(a) + 4 \left( \frac{d}{dx} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u^{(3)}(a) + \left( \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u^{(4)}(a) + \left( \frac{d^{(5)} k(x,x)}{dx^{(5)}} \right)_{x=a} u(a) + \\ &5 \left( \frac{d^{(4)} k(x,x)}{dx^{(4)}} \right)_{x=a} u^{(1)}(a) + 10 \left( \frac{d^{(3)} k(x,x)}{dx^{(3)}} \right)_{x=a} u^{(2)}(a) + 10 \left( \frac{d^{(2)} k(x,x)}{dx^{(2)}} \right)_{x=a} u^{(3)}(a) + 5 \left( \frac{dk(x,x)}{dx} \right)_{x=a} u^{(4)}(a) + k(x,x) u^{(5)}(a) \end{aligned} \quad (36)$$

$$\begin{aligned} u_0^{(7)} = u^{(7)}(a) = f^{(7)}(a) &= \left( \left( \frac{\partial^{(6)} k(x,t)}{\partial x^{(6)}} \right)_{t=x} \right)_{x=a} u(a) + \left( \frac{d}{dx} \left( \frac{\partial^{(5)} k(x,t)}{\partial x^{(5)}} \right)_{t=x} \right)_{x=a} u(a) + \left( \left( \frac{\partial^{(5)} k(x,t)}{\partial x^{(5)}} \right)_{t=x} \right)_{x=a} u^{(1)}(a) + \\ &\left( \frac{d^{(2)}}{dx^{(2)}} \left( \frac{\partial^{(4)} k(x,t)}{\partial x^{(4)}} \right)_{t=x} \right)_{x=a} u(a) + 2 \left( \frac{d}{dx} \left( \frac{\partial^{(4)} k(x,t)}{\partial x^{(4)}} \right)_{t=x} \right)_{x=a} u^{(1)}(a) + \left( \left( \frac{\partial^{(4)} k(x,t)}{\partial x^{(4)}} \right)_{t=x} \right)_{x=a} u^{(2)}(a) + \\ &\left( \frac{d^{(3)}}{dx^{(3)}} \left( \frac{\partial^{(3)} k(x,t)}{\partial x^{(3)}} \right)_{t=x} \right)_{x=a} u(a) + 3 \left( \frac{d^{(2)}}{dx^{(2)}} \left( \frac{\partial^{(3)} k(x,t)}{\partial x^{(3)}} \right)_{t=x} \right)_{x=a} u^{(1)}(a) + 3 \left( \frac{d}{dx} \left( \frac{\partial^{(3)} k(x,t)}{\partial x^{(3)}} \right)_{t=x} \right)_{x=a} u^{(2)}(a) + \\ &\left( \frac{d^{(4)}}{dx^{(4)}} \left( \frac{\partial^{(2)} k(x,t)}{\partial x^{(2)}} \right)_{t=x} \right)_{x=a} u(a) + 4 \left( \frac{d^{(3)}}{dx^{(3)}} \left( \frac{\partial^{(2)} k(x,t)}{\partial x^{(2)}} \right)_{t=x} \right)_{x=a} u^{(1)}(a) + 6 \left( \frac{d^{(2)}}{dx^{(2)}} \left( \frac{\partial^{(2)} k(x,t)}{\partial x^{(2)}} \right)_{t=x} \right)_{x=a} u^{(2)}(a) + \\ &4 \left( \frac{d}{dx} \left( \frac{\partial^{(2)} k(x,t)}{\partial x^{(2)}} \right)_{t=x} \right)_{x=a} u^{(3)}(a) + \left( \left( \frac{\partial^{(3)} k(x,t)}{\partial x^{(3)}} \right)_{t=x} \right)_{x=a} u^{(3)}(a) + \left( \left( \frac{\partial^{(2)} k(x,t)}{\partial x^{(2)}} \right)_{t=x} \right)_{x=a} u^{(4)}(a) + \\ &\left( \frac{d^{(5)}}{dx^{(5)}} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u(a) + 5 \left( \frac{d^{(4)}}{dx^{(4)}} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u^{(1)}(a) + 10 \left( \frac{d^{(3)}}{dx^{(3)}} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u^{(2)}(a) + \\ &10 \left( \frac{d^{(2)}}{dx^{(2)}} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u^{(3)}(a) + 5 \left( \frac{d}{dx} \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u^{(4)}(a) + \left( \left( \frac{\partial k(x,t)}{\partial x} \right)_{t=x} \right)_{x=a} u^{(5)}(a) + \left( \frac{d^{(6)} k(x,x)}{dx^{(6)}} \right)_{x=a} u(a) + \\ &6 \left( \frac{d^{(5)} k(x,x)}{dx^{(5)}} \right)_{x=a} u^{(1)}(a) + 15 \left( \frac{d^{(4)} k(x,x)}{dx^{(4)}} \right)_{x=a} u^{(2)}(a) + 20 \left( \frac{d^{(3)} k(x,x)}{dx^{(3)}} \right)_{x=a} u^{(3)}(a) + 15 \left( \frac{d^{(2)} k(x,x)}{dx^{(2)}} \right)_{x=a} u^{(4)}(a) + \\ &6 \left( \frac{dk(x,x)}{dx} \right)_{x=a} u^{(5)}(a) + k(x,x) u^{(6)}(a) \end{aligned} \quad (37)$$

Now, we try to solve equation (22) using fifth order non-polynomial Spline functions.

Step 8: Approximate  $u_{i+1} = p_i(t_{i+1})$ .

## 6. Using Fifth Order Non-Polynomial Spline Function

In order to approximate the solution of linear VIE's of second kind (22) by using fifth order non-polynomial spline function (1), we present a method of solution in algorithm (VIE2NPS5).

The algorithm (VIE2NPS5):

Step 1: set  $h = \frac{b-a}{n}$ ,  $t_i = t_0 + ih$ ,  $i = 0, 1, \dots, n$

where  $t_0 = a$ ,  $t_n = b$  and  $u_0 = f(a)$ .

Step 2: Evaluate  $a_0, b_0, c_0, d_0, e_0, r_0, z_0$  and  $g_0$  by substituting (30)-(37) In equations (13) - (21).

Step 3: Calculate  $p_0(t)$  using step 2 and equation (12).

Step 4: Approximate  $u_1 = p_0(t_i)$ .

Step 5: For  $i = 1$  to  $n - 1$  do following steps:

Step 6: Evaluate  $a_i, b_i, c_i, d_i, e_i, r_i, z_i$  and  $g_i$  by using equations (13)-(21) and replacing  $u(t_i), u^1(t_i), u^2(t_i), u^3(t_i), u^4(t_i), u^5(t_i), u^6(t_i)$  and  $u^7(t_i)$

by  $p_i(t_i), p_i^1(t_i), p_i^2(t_i), p_i^3(t_i), p_i^4(t_i), p_i^5(t_i), p_i^6(t_i)$  and  $p_i^7(t_i)$

Step 7: Calculate  $p_i(t)$  using step 6 and equation (12).

## 7. Programming

Program 1: linear non-polynomial spline function for solving VIE's of the second kind.

function [x, y, err, u] = volnonpolyspline1st(ker, f, ex, a, b, n)  
syms X T s

h = (b-a)/(n-1);

u = subs(f, X, a)

if isempty(diff(f,1)) == 1

z1=0;

else

z1=diff(f,1)

end

du = subs(z1,a) + subs(ker, {X,T}, {a,a})\*u(1);

if isempty(diff(f,2)) == 1

z2=0;

else

z2=diff(f,2);

end

if isempty(diff(subs(ker, {T}, {X}, 'X')) == 1

z3=0;

else

```

z3=diff(subs(ker,{T},{X}),'X');
end
if isempty(diff(ker,'X'))==1
    zz3=0;
else
    zz3=diff(ker,'X');
end
d2u(1)=subs(z2,a)+subs(zz3,{X,T},{a,a})*u(1)+subs(z3,a)
*u(1)+subs(ker,{X,T},{a,a})*du(1);
if isempty(subs(diff(subs(diff(ker,X),T,X),X,X,a)))==1;
    s=0;
else
    s=subs(diff(subs(diff(ker,X),T,X),X,X,a);
end
if isempty(diff(f,3))==1
    z4=0;
else
    z4=diff(f,3);
end
if isempty(diff(ker,2,'X'))==1
    z5=0;
else
    z5=diff(ker,2,'X');
end
if isempty(subs(diff(subs(ker,T,X),X,2),X,a))==1
    z6=0;
else
    z6=subs(diff(subs(ker,T,X),X,2),X,a);
end
if isempty(subs(diff(subs(ker,T,X),X),X,a))==1
    z7=0;
else
    z7=subs(diff(subs(ker,T,X),X),X,a);
end
d3u(1)=subs(z4,a)+subs(subs(z5,T,X),X,a)*u(1)+...
s*u(1)+subs(subs(diff(ker,X),T,X),X,a)*du(1)+...
z6*u(1)+2*z7*du(1)+subs(ker,{X,T},{a,a})*d2u(1);
a(1)=-d2u(1);
b(1)=-d3u(1);
c(1)=du(1)+d3u(1);
d(1)=u(1)+d2u(1);
for i=1:n-1
    u(i+1)=a(i)*cos(h)+b(i)*sin(h)+h*c(i)+d(i);
    du(i+1)=-a(i)*sin(h)+b(i)*cos(h)+c(i);
    d2u(i+1)=-a(i)*cos(h)-b(i)*sin(h);
    d3u(i+1)=a(i)*sin(h)-b(i)*cos(h);
    d4u(i+1)=a(i)*cos(h)+b(i)*sin(h);
    a(i+1)=-d2u(i+1);
    b(i+1)=-d3u(i+1);
    c(i+1)=du(i+1)+d3u(i+1);
    d(i+1)=u(i+1)+d2u(i+1);
end
x = a:h:b;
y = subs(ex, x);
for i = 1:n
    err(i)=abs(u(i)-subs(ex,x(i)));
end

```

```

format long
disp(' i      X      Y      u      err')
[1:n; x; y; u; err]
figure(1)
plot(x, y, 'b --', 'linewidth', 3)
hold on
plot(x, u, 'r.-', 'linewidth', 1)
grid on
xlabel('X')
ylabel('Y')
title('exact vs. approx.')
legend('exact', 'approx.')
Program 2: sixth no –polynomial spline function for
solving VIE's of the second kind with weakly singular kernel.
function [X, Y, err, u] = volnonpolyspline5st(ker,f, ex, a, b, n)
syms X T
h = (b-a)/(n-1);
u = subs(f, X, a)
x = a:h:b;
y = subs(ex, x);
if isempty(diff(f,1))==1
    z1=0;
else
    z1=diff(f,1)
end
du = subs(z1,a) + subs(ker,{X,T},{a,a})*u(1);
if isempty(diff(f,2))==1
    z2=0;
else
    z2=diff(f,2);
end
if isempty(diff(subs(ker,{T},{X}),'X'))==1
    z3=0;
else
    z3=diff(subs(ker,{T},{X}),'X');
end
if isempty(diff(ker,'X'))==1
    zz3=0;
else
    zz3=diff(ker,'X');
end
d2u(1)=subs(z2,a)+subs(zz3,{X,T},{a,a})*u(1)+subs(z3,a)
*u(1)+subs(ker,{X,T},{a,a})*du(1);
if isempty(subs(diff(subs(diff(ker,X),T,X),X,X,a)))==1;
    s=0;
else
    s=subs(diff(subs(diff(ker,X),T,X),X,X,a);
end
if isempty(diff(f,3))==1
    z4=0;
else
    z4=diff(f,3);
end
if isempty(diff(ker,2,'X'))==1
    z5=0;
else
    z5=diff(ker,2,'X');
end

```

```

end
if isempty(subs(diff(subs(ker, T, X), 2, X), X, a))==1
    z6=0;
else
    z6=subs(diff(subs(ker, T, X), 2, X), X, a);
end
if isempty(subs(diff(subs(ker, T, X), X), X, a))==1
    z7=0;
else
    z7=subs(diff(subs(ker, T, X), X), X, a);
end
d3u(1)=subs(z4, a)+subs(subs(z5, T, X), X, a)*u(1)+...
    s*u(1)+subs(diff(subs(diff(ker, X), T, X), X, a))*du(1)+...
    z6*u(1)+2*z7*du(1)+subs(ker, {X, T}, {a, a})*d2u(1);
if isempty(diff(f,4))==1
    z8=0;
else
    z8=diff(f,4);
end
if isempty(diff(ker,3,'X'))==1
    z9=0;
else
    z9=diff(ker,3,'X');
end
if isempty(subs(diff(ker,X,2),T,X))==1
    z10=0;
else
    z10=subs(diff(ker, X, 2),T,X);
end
if isempty(subs(diff(z10,X),X,a))==1
    z1010=0;
else
    z1010=subs(diff(z10,X),X, a);
end
if isempty(diff(ker,2,'X'))==1
    z11=0;
else
    z11=diff(ker,2,'X');
end
if isempty(subs(diff(ker,'X'),T,X))==1
    z12=0;
else
    z12=subs(diff(ker,'X'),T,X);
end
if isempty(subs(diff(z12,X,2),X,a))==1
    z1212=0;
else
    z1212= subs(diff(z12,X,2),X,a);
end
if isempty(subs(diff(subs(ker, T, X), X, 3), X, a))==1
    z14=0;
else
    z14=subs(diff(subs(ker, T, X), X, 3), X, a);
end
if isempty(subs(diff(subs(ker, T, X), X, 2), X, a))==1
    z15=0;
else

```

```

    z15=subs(diff(subs(ker, T, X), X, 2), X, a);
end
if isempty(subs(diff(subs(ker, T, X), X), X, a))==1
    z16=0;
else
    z16=subs(diff(subs(ker, T, X), X), X, a);
end
d4u(1)=subs(z8, a)+subs(subs(z9, T, X), X, a)*u(1)+...
    z1010*u(1)+subs(subs(z11, T, X), X, a)*du(1)+...
    z1212*u(1)+2*s*du(1)+subs(subs(diff(ker,X),T,X),
    X,a)*d2u(1)+...
z14*u(1)+3*z15*u(1)+3*z16*d2u(1)+subs(ker,{X,T},{a,a})
*d3u(1);
if isempty(diff(f,5))==1
    z17=0;
else
    z17=diff(f,5);
end
if isempty(diff(ker,4,'X'))==1
    z18=0;
else
    z18=diff(ker,4,'X');
end
if isempty(subs(diff(ker, X,3),T,X))==1
    z19=0;
else
    z19=subs(diff(ker,X,3),T, X);
end
if isempty(subs(diff(z19,X), X,a))==1
    z1919=0;
else
    z1919=subs(diff(z19,X), X, a);
end
if isempty(diff(ker,3,'X'))==1
    z20=0;
else
    z20=diff(ker,3,'X');
end
if isempty(subs(diff(ker,2,'X'),T,X))==1
    z21=0;
else
    z21=subs(diff(ker,2,'X'),T,X );
end
if isempty(subs(diff(z21,X,2),X,a))==1
    z2121=0;
else
    z2121=subs(diff(z21,X,2),X,a);
end
if isempty(subs(diff(ker,X, 2),T,X))==1
    z22=0;
else
    z22=subs(diff(ker, X,2),T, X);
end
if isempty(subs(diff(z22,X), X, a))==1
    z2222=0;
else

```

```

z2222=subs(diff(z22,X ), X, a);
end
if isempty(subs(diff(ker,'X'),T,X))==1
    z24=0;
else
    z24=subs(diff(ker,'X'),T,X);
end
if isempty(subs(diff(z24,X,3),X,a))==1
    z2424 =0;
else
    z2424=subs(diff(z24,X,3),X,a);
end
if isempty(subs(diff(subs(ker, T, X), X, 4), X, a))==1
    z28=0;
else
    z28=subs(diff(subs(ker, T, X), X, 4), X, a);
end
d5u(1)=subs(z17,a)+subs(subs(z18, T, X), X, a)*u(1)+...
    z1919*u(1)+subs(subs(z20, T, X), X, a)*du(1)+z2121*u(1)+...
z2121*d2u(1)+2*z2222*du(1)+subs(subs(z11,T,X),X,a)*d2u
(1)+...
    z2424*u(1)+3*z1212*du(1)+3*s*d2u(1)+...
    subs(subs(diff(ker, X),T,X), X, a)*d3u(1)+...
    z28*u(1)+4*z14*du(1)+6*z15*d2u(1)+...
    4*z16*d3u(1)+ subs(ker,{X,T},{a,a})*d4u(1);
if isempty(diff(f,6))==1
    z31=0;
else
    z31=diff(f,6);
end
if isempty(diff(ker,5,'X'))==1
    z32=0;
else
    z32=diff(ker,5,'X');
end
if isempty(subs(diff(ker, X,4),T,X))==1
    z33=0;
else
    z33=subs(diff(ker,X,4),T, X);
end
if isempty(subs(diff(z33,X), X,a))==1
    z3333=0;
else
    z3333=subs(diff(z33,X), X, a);
end
if isempty(diff(ker,4,'X'))==1
    z34=0;
else
    z34=diff(ker,4,'X');
end
if isempty(subs(diff(ker,3,'X'),T,X))==1
    z35=0;
else
    z35=subs(diff(ker,3,'X'),T,X );
end

```

```

if isempty(subs(diff(z35,X,2),X,a))==1
    z3535 =0;
else
    z3535=subs(diff(z35,X,2),X,a);
end
if isempty(subs(diff(ker,X, 2),T,X))==1
    z36=0;
else
    z36=subs(diff(ker, X,2),T, X);
end
if isempty(subs(diff(z36,X,3), X, a))==1
    z3636=0;
else
    z3636=subs(diff(z36,X,3), X, a);
end
if isempty(subs(diff(ker,'X'),T,X))==1
    z37=0;
else
    z37=subs(diff(ker,'X'),T,X);
end
if isempty(subs(diff(z37,X,3),X,a))==1
    z3737 =0;
else
    z3737=subs(diff(z37,X,3),X,a);
end
if isempty(subs(diff(subs(ker, T, X), X, 5), X, a))==1
    z38=0;
else
    z38=subs(diff(subs(ker, T, X), X, 5), X, a);
end
d6u(1)=subs(z31,a)+subs(subs(z32, T, X), X, a)*u(1)+...
    z3333*u(1)+subs(subs(z34, T, X), X, a)*du(1)+z3535*u(1)+...
2*z1919*du(1)+z3636*u(1)+3*z2121*du(1)+3*z2222*d2u(
1)+...
    subs(subs(z11,T,X),X,a)*d3u(1)+z3737*u(1)+...
    4*z2424*du(1)+6*z1212*d2u(1)+4*s*d3u(1)+...
    subs(subs(diff(ker, X),T,X), X, a)*d4u(1)+...
z38*u(1)+5*z28*du(1)+10*z14*d2u(1)+10*z15*d3u(1)+...
    5*z16*d4u(1)+ subs(ker,{X,T},{a,a})*d5u(1);
if isempty(diff(f,7))==1
    z39=0;
else
    z39=diff(f,7);
end
if isempty(diff(ker,6,'X'))==1
    z40=0;
else
    z40=diff(ker,6,'X');
end
if isempty(subs(diff(ker, X,5),T,X))==1
    z41=0;
else
    z41=subs(diff(ker,X,5),T, X);
end

```



```

if isempty(subs(diff(z41,X), X,a))==1
    z4141=0;
else
    z4141=subs(diff(z41,X), X, a);
end
if isempty(diff(ker,5,'X'))==1
    z42=0;
else
    z42=diff(ker,5,'X');
end
if isempty(subs(diff(ker,4,'X'),T,X))==1
    z43=0;
else
    z43=subs(diff(ker,4,'X'),T,X);
end
if isempty(subs(diff(z43,X,2),X,a))==1
    z4343=0;
else
    z4343=subs(diff(z43,X,2),X,a);
end
if isempty(subs(diff(ker,X,3),T,X))==1
    z44=0;
else
    z44=subs(diff(ker,X,3),T,X);
end
if isempty(subs(diff(z44,X,3),X,a))==1
    z4444=0;
else
    z4444=subs(diff(z44,X,3),X,a);
end
if isempty(subs(diff(ker,X,2),T,X))==1
    z45=0;
else
    z45=subs(diff(ker,X,2),T,X);
end
if isempty(subs(diff(z45,X,3),X,a))==1
    z4545=0;
else
    z4545=subs(diff(z45,X,3),X,a);
end
if isempty(subs(diff(ker,'X'),T,X))==1
    z46=0;
else
    z46=subs(diff(ker,'X'),T,X);
end
if isempty(subs(diff(z46,X,5),X,a))==1
    z4646=0;
else
    z4646=subs(diff(z46,X,5),X,a);
end
if isempty(subs(diff(subs(ker,T,X),X,6),X,a))==1
    z47=0;
else
    z47=subs(diff(subs(ker,T,X),X,6),X,a);
end
d7u(1)=subs(z39,a)+subs(subs(z40,T,X),X,a)*u(1)+...
    z4141*u(1)+subs(subs(z42,T,X),X,

```

```

a)*du(1)+z4343*u(1)+...
    2*z3333*du(1)+subs(subs(z34,T,X),X,a)*d2u(1)+...
    z4444*u(1)+3*z3535*du(1)+3*z1919*d2u(1)+...
    z4545*u(1)+4*z3636*du(1)+6*z2121*d2u(1)+...
    4*z2222*d3u(1)+subs(subs(z20,T,X),X,a)*d3u(1)+...

subs(subs(z11,T,X),X,a)*d4u(1)+z4646*u(1)+5*z3737*du(1)
)+...
    10*z2424*d2u(1)+10*z1212*d3u(1)+5*s*d4u(1)+...
    subs(subs(diff(ker,X),T,X),X,a)*d5u(1)+...

z47*u(1)+6*z38*du(1)+15*z28*d2u(1)+20*z14*d3u(1)+...

15*z15*d4u(1)+6*z16*d5u(1)+subs(ker,{X,T},{a,a})*d6u(1);
a(1)=-d6u(1);
b(1)=d7u(1);
c(1)=du(1)-d7u(1);
d(1)=(1/2)*(d2u(1)-d6u(1));
e(1)=(1/6)*(d3u(1)+d7u(1));
r(1)=(1/24)*(d4u(1)+d6u(1));
z(1)=(1/120)*(d5u(1)-d7u(1));
g(1)=u(1)+d6u(1);
for i=1:n-1
    u(i+1)=a(i)*cos(h)+b(i)*sin(h)+h*c(i)+d(i)*h^2+
e(i)*h^3+r(i)*h^4+z(i)*h^5+g(i);
    du(i+1)=-a(i)*sin(h)+b(i)*cos(h)+c(i)+2*d(i)*h+
3*e(i)*h^2+4*r(i)*h^3+5*z(i)*h^4;
    d2u(i+1)=-a(i)*cos(h)-b(i)*sin(h)+2*d(i)+6*e(i)*h
+12*r(i)*h^2+20*z(i)*h^3;
    d3u(i+1)=a(i)*sin(h)-b(i)*cos(h)+6*e(i)+24*r(i)*h
+60*z(i)*h^2;
    d4u(i+1)=a(i)*cos(h)+b(i)*sin(h)+24*r(i)+
120*z(i)*h;
    d5u(i+1)=-a(i)*sin(h)+b(i)*cos(h)+120*z(i);
    d6u(i+1)=-a(i)*cos(h)-b(i)*sin(h);
    d7u(i+1)=a(i)*sin(h)-b(i)*cos(h);
    a(i+1)=-d6u(i+1);
    b(i+1)=d7u(i+1);
    c(i+1)=du(i+1)-d7u(i+1);
    d(i+1)=(1/2)*(d2u(i+1)-d6u(i+1));
    e(i+1)=(1/6)*(d3u(i+1)+d7u(i+1));
    r(i+1)=(1/24)*(d4u(i+1)+d6u(i+1));
    z(i+1)=(1/120)*(d5u(i+1)-d7u(i+1));
    g(i+1)=u(i+1)+d6u(i+1);
end
err=zeros(1,n);
disp(h)
disp('length of x')
disp(length(x))
disp('length of y')
disp(length(y))
for k=1:n
    err(k)=abs(u(k)-y(k));
end
format long
disp(' i      X      Y      u      err')
[1:n;x;y;u;err]

```

```
figure(1)
plot(x, y, 'b --', 'linewidth', 3)
hold on
plot(x, u, 'r-', 'linewidth', 1)
grid on
xlabel('X')
ylabel('Y')
title('exact vs. approx.')
legend('exact', 'approx.')
```

## 8. Numerical Examples and Figures

Example (2): Consider the VIE of second kind [10]:

$$u(x) = x + \int_0^x (t-x)u(t)dt \quad 0 \leq x \leq 1$$

Where  $f(x) = x$  and kernel  $= t - x$ , with  $y = \sin x$

Table 1 present a comparison between the exact and numerical solution using linear [13] and fifth order non-polynomial spline functions, where  $u_i(x)$  denotes the approximate solution using non- polynomial spline functions, with  $h = 0.1$ .

**Table 1.** Analytical and numerical solution of test example(2).

x	Exact solution	$u_i(x)$	
		linear	Five order
0	0.000000000000000	0.000000000000000	0.000000000000000
0.1	0.09983341664683	0.09983341664683	0.09983341668651
0.2	0.19866933079506	0.19916808204436	0.19866933583213
0.3	0.29552020666134	0.29800648787140	0.29552028832715
0.4	0.38941834230865	0.39635111335875	0.38941890981745
0.5	0.47942553860420	0.49420442535136	0.47942806425312
0.6	0.56464247339504	0.59156887837027	0.56465102008993
0.7	0.64421768723769	0.68844691467410	0.64424167375464
0.8	0.71735609089952	0.78484096432036	0.71741482576318
0.9	0.78332690962748	0.88075344522641	0.78345650410261
1	0.84147098480790	0.97618676323008	0.84173434026604

Table 2: present a comparison between the error in our methods and other method] where error=|exact value - numerical

**Table 2.** Comparison between the error with.

x	Error in linear	Error in five order
0	0.000000000000000	0.000000000000000
0.1	0.000000000000000	0.00000000003968
0.2	0.00049875124930	0.00000000503707
0.3	0.00248628121006	0.00000008166582
0.4	0.00693277105010	0.00000056750880
0.5	0.01477888674716	0.00000252564892
0.6	0.02692640497523	0.00000854669489
0.7	0.04422922743640	0.00002398651695
0.8	0.06748487342084	0.00005873486365
0.9	0.09742653559892	0.00012959447513
1	0.13471577842218	0.00026335545814

Example (3): Consider the VIE of second kind [14]:

$$u(x) = 1 + \int_0^x (t-x)u(t)dt \quad 0 \leq x \leq 1$$

Where  $f(x) = 1$  and kernel  $= t - x$ , with  $y = \cos x$

Table 3 present a comparison between the exact and numerical solution using linear [13] and five order non-polynomial spline functions, where  $u_i(x)$  denotes the approximate solution using non- polynomial spline functions, with  $h = 0.1$ .

**Table 3.** Exact and numerical solution of test example(3).

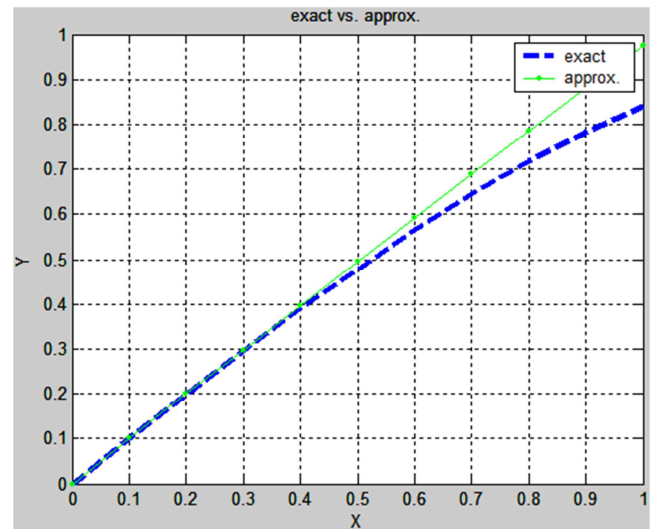
x	Exact solution	$u_i(x)$	
		linear	Five order
0	1.000000000000000	1.000000000000000	1.000000000000000
0.1	0.99500416527803	0.99500416527803	0.99500416527803
0.2	0.98006657784124	0.99003328892062	0.98006657783728
0.3	0.95533648912561	0.98508724623992	0.95533648861880
0.4	0.92106099400289	0.98016591317098	0.92106098534563
0.5	0.87758256189037	0.97526916626867	0.87758249662003
0.6	0.82533561490968	0.97039688270455	0.82533529782125
0.7	0.76484218728449	0.96554894026384	0.76484101853443
0.8	0.69670670934717	0.96072521734230	0.69670315178005
0.9	0.62160996827066	0.95592559294323	0.62160056477445
1	0.54030230586814	0.95114994667438	0.54028001149101

Table 4: present a comparison between the error in our methods and other method in [10] where error=|exact value - numerical

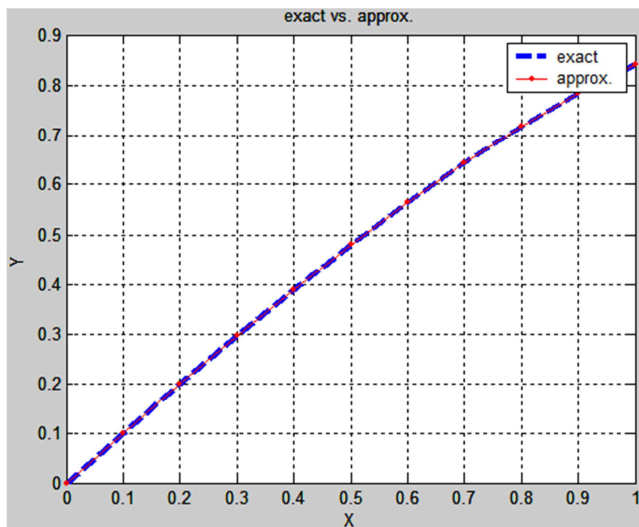
**Table 4.** Comparison between the error with.

x	Error in linear	Error in five order
0	0.000000000000000	0.000000000000000
0.1	0.000000000000000	0.000000000000000
0.2	0.00996671107938	0.000000000000396
0.3	0.02975075711431	0.00000000050681
0.4	0.05910491916810	0.00000000865725
0.5	0.09768660437829	0.00000006527035
0.6	0.14506126779487	0.00000031708843
0.7	0.20070675297935	0.00000116875006
0.8	0.26401850799514	0.00000355756712
0.9	0.33431562467256	0.00000940349622
1	0.41084764080624	0.00002229437713

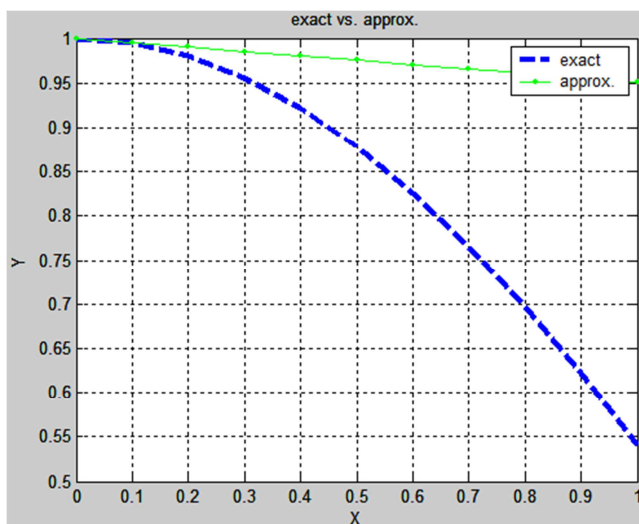
The Figures show the comparison between the analytical and the approximate solution using linear and fifth order non-polynomial spline functions for the examples.



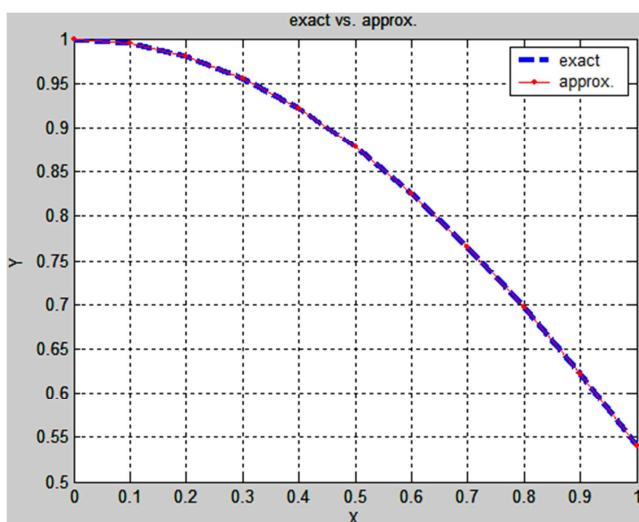
**Figure 1.** Comparison between the analytical and the approximate solution using linear non-polynomial spline functions for example (2).



**Figure 2.** Comparison between the analytical and the approximate solution using fifth order non-polynomial spline functions for example (2).



**Figure 3.** Comparison between the analytical and the approximate solution using linear non-polynomial spline functions for example (3).



**Figure 4.** Comparison between the analytical and the approximate solution using fifth order non-polynomial spline functions for example (3).

## 9. Results

In general the methods which are used in this paper proved their effectiveness in solve linear Volterra Integral Equations of second kind numerically and finding an accurate results. The Figures show a comparison between the Adomian Decomposition and numerical solution which was presented in example (2) and example (3), the results show that the fifth order non-polynomial Spline functions gives the best approximation than the linear non-polynomial Spline functions.

## 10. Conclusion

Finally we found that the fifth order non- polynomial Spline functions characterized with an easy, fast and accuracy is high comparing with the less fifth order non-polynomial Spline functions and the results that we obtained show that the fifth order non-polynomial Spline functions gives the best approximation to solve the linear non-polynomial Spline functions. The proposed scheme is simple and computationally attractive and its accuracy is high and we can simply execute this method by Matlab.

## References

- [1] Abdel Radi. A. A and Samia. A. Y, Comparison Between Analysis Solutions of Volterra and Fredholm Integral Equations of Second Kind, EPH - International Journal of Mathematics and Statistics vol. 6, issue 9, p.p 3-5, 2020.
- [2] Haq, F. I. (2009). Numerical Solution of Bounded Value Problem and Initial Value Problems Using Spline Function. ph.d, thesis, GhulamIshaq Institute of Engineering Science and Technology, Pakistan.
- [3] HARBI, S, MURAD, M and MAJEED, M 2015-A solution of second integral equation using third order Non-polynomial Spline function. College of Science for Women Baghdad University.
- [4] H. Brunner, theory and numerical solution of Volterra functional integral equations, Hong Kong Baptist University, (2010).
- [5] HOSSINPOUR, A 2012- The Solve of Integral Differential Equation by Non-polynomial Spline Function and Quadrature Formula. International Conference on Applied Mathematics and Pharmaceutical science Jan, pp 7-8, pp. 595-597.
- [6] LIMA, P and DIAGO, T 1997- An extrapolation method for a Volterra integral equation with weakly singular kernel Appl. Numer. Math, vol. 24, pp. 131-148.
- [7] Najwa, S and Mohammed, S presented a numerical solution for linear Volterra integral equations with weakly singular kernel by using non-polynomial spline function from the fifth degree, pp. 109-110.
- [8] Majeed. S. N (2014) Solution Of Second Kind Volterra Integro Equation Using Linear Non Polynomial Spline Equation. Mathematical Theory and Modeling.

- [9] M. S. Islam, M. Z. I. Bangalee, A. K.. Khan, and, A, Halder," approximate solution of systems of VIE's of second kind by Adomain decomposition method," Dhaka university journal of science, vol. 63, no. 1, pp. 15-18, 2007.
- [10] Rahman, M. M, Hakim M. A., Hassan M. K., Alam M. K. and Nowsher, L., 2012, Numerical Solution of Volterra Integral Equations of Second kind with the Help of Chebyshev Polynomials, Annals of Pure and Applied Mathematics, 1 (2): 158-167.
- [11] Ramadan, M. A.; EL-Danaf, T.; and E. I. Abdaal, F (2007). Application of the Non- Polynomial Spline Approach to the Solution of the Burgers Equation. The Open Applied Mathematics Journal (1): 15-20.
- [12] Rice, J. R. (1985). Numerical Method software and Analysis. Software and analysis, Mcgra Hill.
- [13] Sara. H. H, (2013), Algorithms for Solving Volterra Integral Equations Using Non-Polynomial Notch Functions, College of Science for Women Baghdad University.
- [14] Tahmasbi, A. (2008). New Approach numerical solution of Linear Volterra Integral Equations of Second kind. 3 (32), 1607-1610.
- [15] Taqi, A; Jumaa, B (2016). Non- polynomial spline functions to solve a system of two non- linear volterra integral equations. Kirkuk University, Scientific Studies.
- [16] WAZWAZ, A 2011-Linear and Non Linear Integral Equation Method and Application. Higher Education Press, Beijing, p. 66-69.
- [17] Zarebnia, M.; Hoshyar, M.; and Sedahti, M. (2011). Non-Polynomial Spline Method for the Solution of Problem in Calculus of Variations. word Academy Engendering Technology (51): 986-991.