

Modeling of The Quadratic Non-Polynomial Spline Function for Solving Volterra Integral Equations Using MATLAB.

Asst.L.Douaa Kh. Abid¹

Asst.L.Zahraa A. Ali²

¹Directorate of Education of First Karkh, 1The Secondary School for Distinguished Boys in Harithiyah, Baghdad, Iraq.

²MDepartment of mathematics, Faculty of science for women, University of Baghdad, Baghdad, Iraq.

*corresponding author: Douaa.abid.da@gmail.com

E-mail address: zahra.am2009@yahoo.com

Abstract:

Integral equations play an important role in many theoretical and applied types of research, due to the possibility of expressing the Integral equation as a continuous or discontinuous Integral operator and thus modeling some problems in research that accept the Integral operator as a model for the mathematical description of the applied issue being treated, and from here we see that the Integral equations play a role It is essential for mathematical modeling with integrative influences, such as elasticity problems in continuous mechanics, where the behavior of viscoelastic-elastic long-memory media can be expressed by Volterra's integral equations, in addition to many other applications in architectural structures in addition to calculating transformations in the field of electronics, analytical mechanics and other fields of physics. In this research, we will review the integrative equations, and we will focus on Volterra's integral equations and ways to solve them. The solution will be detailed using the Spline quadratic function of VIE from the second type and programmed using MATLAB, in addition to practical applications to reach a deep and easy understanding of it.

Key Words: Volterra Integral Equation, Fredholm integral equations, Integro-differential equations, Singular integral equations, MATLAB Code, Quadratic Spline Non-Polynomial.

نمذجة دالة الشريحة التربيعية غير متعددة الحدود في حل معادلة فولتيرا التكاملية باستخدام

MATLAB

م.م. دعاء خضير عبد¹

م.م. زهراء اموري علي²

¹ ثانوية المتميزين للبنين الحارثية/ تربية الكرخ الاولى ، بغداد ، العراق.

² قسم تقنيات هندسة التبريد والتكييف ، كاية دجلة الجامعة الاهلية، بغداد، العراق.

خلاصة:

تلعب المعادلات التكاملية دوراً مهماً في العديد من أنواع البحث النظرية والتطبيقية ، نظراً لإمكانية التعبير عن المعادلة التكاملية كعامل تكامل مستمر أو متقطع وبالتالي نمذجة بعض المشكلات في البحث التي تقبل العامل التكامل كمنموذج للوصف الرياضي من المسألة التطبيقية التي يتم معالجتها ، ومن هنا نرى أن المعادلات التكاملية تلعب دوراً.إنها ضرورية للنمذجة الرياضية ذات التأثيرات التكاملية ، مثل

مشاكل المرونة في الميكانيكا المستمرة ، حيث يمكن أن يكون سلوك وسائط الذاكرة الطويلة اللزجة المرنة معبراً عنها بمعادلات فولتيرا المتكاملة ، بالإضافة إلى العديد من التطبيقات الأخرى في الهياكل المعمارية بالإضافة إلى حساب التحولات في مجال الإلكترونيات والميكانيكا التحليلية ومجالات الفيزياء الأخرى. في هذا البحث سنراجع المعادلات التكاملية وسنركز على معادلات فولتيرا التكاملية وطرق حلها. سيتم تفصيل الحل باستخدام وظيفة الشريحة التربيعية لـ VIE من النوع الثاني وبرمجتها باستخدام MATLAB ، بالإضافة إلى التطبيقات العملية للوصول إلى فهم عميق وسهل لها. الكلمات الافتتاحية : معادلة فولتيرا التكاملية ، معادلة فريدهولم التكاملية، المعادلات التكاملية التفاضلية، المعادلات التكاملية الشاذة.

Introduction:

Equations with the unknown function $u(x)$ to be determined appearing under the integral sign are referred to as integral equations. One of the most helpful mathematical tools in both pure and applied mathematics is the study of integral equations. It is extremely useful in solving numerous physical issues. Solving a few approximatively integral equations can help with a number of initial and boundary value problems related to ordinary differential equations (ODE) and partial differential equations (PDE)¹⁻⁴. A typical form of an integral equation in $u(x)$ is of the form:

$$“u(x) = f(x) + \lambda \int_{\alpha(x)}^{\beta(x)} K(x,t)u(t)dt” \dots\dots\dots 1$$

where $K(x,t)$ is the integral kernel and $\alpha(x)$ and $\beta(x)$ are the limits of integration. The unknown function $u(x)$ appears under the integral sign, which can be easily observed. The kernel $K(x,t)$ and the function $f(x)$ in Eq.1 are both given functions, while λ is a constant parameter. The primary purpose of this text is to identify the unknown function $u(x)$ that satisfies Eq.1 using a variety of techniques. We will expend considerable effort exploring these techniques in an attempt to find solutions for the unknown function.

Integral equation classification:

As with ordinary and partial differential equations, an integral equation can be classified as either a linear or nonlinear integral equation. We observed in the previous section that the differential equation can be represented equivalently by the integral equation. Therefore, these two equations have a strong relationship^{1,2}.

Volterra integral equations and Fredholm integral equations are the two most common types of integral equations. Naturally, we must classify them as homogeneous or heterogeneous, as well as linear or nonlinear. Also encountered in some practical problems are singular equations.

The two main classes and the two related types of integral equations will be distinguished as the four main types of integral equations in this text. The four categories are specifically listed below⁵:

- “Volterra integral equations”
- “Fredholm integral equations”

- “Integro-differential equations”
- “Singular integral equations”

Volterra integral equations”:

Volterra linear integral equations typically take the following shape⁴⁻⁷

$$“\varphi(x).u(x) = f(x) + \lambda \int_{\alpha}^x K(x,t).u(t).dt” \dots\dots 2$$

When the limits of integration are functions of x and the unknown function $u(x)$ appears under the integral sign in a linear fashion. If the function $\varphi(x)=1$, then the second equation becomes:

$$“u(x) = f(x) + \lambda \int_{\alpha}^x K(x,t).u(t).dt” \dots\dots\dots 3$$

This equation is known as the second-kind Volterra integral equation; however, if $\varphi(x)=0$, then Eq.2 becomes:

$$“f(x) + \lambda \int_{\alpha}^x K(x,t).u(t).dt = 0 “ \dots\dots\dots 4$$

which is known as the Volterra equation of the first kind.

Quadratic classic spline non-Polynomial^{8,9}:

Professor Ryabenkiy V.S. used quadratic spline approximations for the very first time. Note that polynomial Courant functions have been utilized for a very long time and are well-known. A function Q is said to as a “Quadratic spline” if it meets the conditions:

Q, Q' are continuous on $[a, b]$, and Q is a maximum degree 2 polynomial on each subinterval. $[x_i, x_{i+1}]$ where $a = x_0 < x_1 < \dots < x_n = b$ Quadratic spline has a form:

$$q_i = a_i + b_i(x - x_i) + c_i(x - x_i)^2$$

Each non- polynomial spline of n order $Q_i(t)$ is known as:

$$Q_i(t) = a_i \cos k(t - t_i) + b_i \sin k(t - t_i) + c_i(t - t_0) + d_i(t - t_i)^2 + e_i \dots\dots\dots 5$$

a_i, b_i, c_i, d_i, e_i are constant.

K is the frequency of the trigonometric functions which will be used to raise the accuracy of the method.

Let's derivate the equation three times to t:

$$Q'_i(t) = -k^2 a_i \sin k(t - t_i) + k b_i \cos k(t - t_i) + c_i + d_i(t - t_i)$$

$$Q''_i(t) = -k^2 a_i \cos k(t - t_i) - k^2 b_i \sin k(t - t_i) + c_i + 2d_i$$

$$Q'''_i(t) = k^3 a_i \sin k(t - t_i) - k^3 b_i \cos k(t - t_i)$$

(6)



$$Q^{(4)}_i(t) = k^4 a_i \cos k(t - t_i) - k^4 b_i \sin k(t - t_i)$$

Now let's replace each t with t_i then we get:

$$Q_i(t_i) = a_i + e_i = u(t_i)$$

$$Q'_i(t_i) = kb_i + c_i = u'(t_i)$$

$$Q''_i(t_i) = -k^2 a_i + 2d_i = u''(t_i)$$

$$Q'''_i(t_i) = -k^3 b_i = u'''(t_i)$$

$$Q^{(4)}_i(t_i) = k^4 a_i = u^{(4)}(t_i) \quad (\text{by solving these equations, we can get } a_i, b_i, c_i, d_i, e_i)$$

$$a_i = -\frac{1}{k^4} Q^{(4)}_i(t_i) \dots\dots\dots 7$$

$$b_i = -\frac{1}{k^3} Q'''_i(t_i) \dots\dots\dots 8$$

$$c_i = Q'_i(t_i) - kb_i \dots\dots\dots 9$$

$$d_i = \frac{1}{2} Q''_i(t_i) - a_i \dots\dots\dots 10$$

$$e_i = Q_i(t_i) - a_i \dots\dots\dots 11$$

For $i=1,2,3,\dots\dots\dots$

Based on the previous equations we can solve Volterra integral equations by transferring it to Non polynomial Spline Quadratic by following the below points:

- $h = \frac{b-a}{n}$
- $t_i = t_0 + ih \quad i=0,1,2,3,\dots\dots\dots n$, where n is positive integer
- $a = t_0, \quad t_n = b$
- $u_0 = u(a) = f(a) \dots\dots\dots 12$
- $u'_0 = u'(a) = f'(a) + k(a,a)u(a) \dots\dots\dots 13$
- $u''_0 = u''(a) = f''(a) + \left(\frac{\partial k(x,t)}{\partial x}\right)_{t=x} u(a) + \left(\frac{\partial k(x,x)}{\partial x}\right)_{x=a} u(a) + k(a,a)u'(a) \dots\dots\dots 14$
- $u'''_0 = u'''(a) = f'''(a) + \left(\frac{\partial^2 k(x,t)}{\partial x^2}\right)_{t=x} u(a) + \left(\frac{\partial}{\partial x}\left(\frac{\partial k(x,t)}{\partial x}\right)_{t=x}\right)_{x=a} u(a) + \left(\frac{\partial k(x,t)}{\partial x}\right)_{t=x} u'(a) + \left(\frac{\partial^2 k(x,x)}{\partial x^2}\right)_{x=a} u(a) + 2\left(\frac{\partial k(x,x)}{\partial x}\right)_{x=a} u'(a) + k(a,a)u''(a) \dots\dots\dots 15$
- $u_0^4 = u^4(a) = f^4(a) + \left(\left(\frac{\partial^3 k(x,t)}{\partial x^3}\right)_{t=x}\right)_{x=a} u(a) + \left(\frac{\partial}{\partial x}\left(\frac{\partial^2 k(x,t)}{\partial x^2}\right)_{t=x}\right)_{x=a} u(a) + \left(\frac{\partial^2 k(x,t)}{\partial x^2}\right)_{t=x} u'(a) + \left(\frac{\partial^3 k(x,x)}{\partial x^3}\right)_{x=a} u'(a) + \left(\frac{\partial^2}{\partial x^2}\left(\frac{\partial k(x,x)}{\partial x}\right)_{x=a}\right) u'(a) + \left(\frac{\partial^2 k(x,x)}{\partial x^2}\right)_{x=a} u''(a) + 3\left(\frac{\partial k(x,x)}{\partial x}\right)_{x=a} u''(a) + k(a,a)u'''(a) \dots\dots\dots 16$

- Let's calculate a_0, b_0, c_0, d_0, e_0 by replacing Equations from (7) to (11) in (12) to (15).
- Let's calculate $u_1 \approx Q_i(t_1)$ (the approximate solution).
- For $i=1:n-1$, lets calculate a_i, b_i, c_i, d_i, e_i replacing $u(t_i), u'(t_i), u''(t_i)$ and $u'''(t_i)$ by $Q_i(t_i), Q_i'(t_i), Q_i''(t_i)$ and $Q_i'''(t_i)$.
- Calculate $Q_i(t)$.
- Let's approximate $u_{i+1} \approx Q_i(t_{i+1})$.

MATLAB Code to solve VIE2 Using Quadratic Non-Polynomial Function (QNPF):

Code Name: VIE2QNPF(K,f,ex,a,b,n)

Parameters:

“**K**: The kernel function that takes two arguments.

f function: The left-hand side (free) function with $f(a) = 0$.

a (float): Lower bound of the integral, defaults to 0.

b float: Upper bound of the estimate, defaults to 1.

n: Number of estimation points between zero and b.”

Output: u: The approximate Solution by using Quadratic Spline non-polynomial

err: is the difference between the exact solution and the approximate.

“**function [u,err]=VIE2QNPF(K,f,ex,a,b,n)**

syms x t s

h=(b-a)/n;

u(1)=subs(f,a);

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

z1=0;

else

z1=diff(f,1);

end

du(1)=subs(z1,a)+subs(K,{x,t},{a,a})*u(1);

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



```
z2=0;
else
z2=diff(f,2);
end
if isempty(diff(subs(K,{t},{x}),'x'))==1
z3=0;
else
z3=diff(subs(K,{t},{x}),'x');
end
if isempty(diff(K,'x'))==1
zz3=0;
else
zz3=diff(K,'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(K,'x'),{t},{x})),{x},{a}))==1
z=0 ;
else
z= subs(diff(subs(diff(K,'x'),{t},{x})),{x},{a});
end
if isempty(diff(f,3))==1
z4=0;
else
z4=diff(f,3);
end
if isempty(diff(K,2,'x'))==1
z5=0;
else
```



```

z5=diff(K,2,'x');
end
if isempty(subs(diff(subs(K,{x,t},{x,x}),2),{x},{a}))=1
z6=0;
else
z6=subs(diff(subs(K,{x,t},{x,x}),2),{x},{a});
end
if isempty(subs(diff(subs(K,{x,t},{x,x})),{x},{a}))=1
z7=0;
else
z7=subs(diff(subs(K,{x,t},{x,x})),{x},{a});
end
d3u(1)=subs(z4,a)+subs(subs(z5,{x,t},{x,x}),{x},{a})*u(1)+z*u(1)+subs(subs(diff(K,'x'),{x,t},{x,x}),{x},{a})*du(1)+z6*u(1)+2*z7*du(1)+subs(K,{x,t},{a,a})
*d2u(1);
if isempty(diff(f,4))==1
z8 =0;
else
z8=diff(f,4);
end
if isempty(diff(K,3,'x'))=1
z9=0;
else
z9= diff(K,3,'x');
end
if isempty(subs(diff(K,2,'x'),{t},{x}))=1
z10=0;
else
z10= subs(diff(K,2,'x'),{t},{x});

```



```
end
if isempty (subs(diff(z10),{x},{a}))==1
z1010=0;
else
z1010=subs(diff(z10),{x},{a});
end
if isempty(diff(K,2,'x'))==1
z11=0;
else
z11=diff(K,2,'x');
end
if isempty (diff(K,'x'))==1
z12=0;
else
z12=diff(K,'x');
end
if isempty(subs(z12,{t},{x}))==1
c=0;
elsehgjf
c=subs(z12,{t},{x});
end
if isempty (sym(c))==1
z1212=0;
else
z1212= sym(c);
end
if isempty(subs(diff(K,'x'),{t},{x}))==1
z13=0;
else
```



```

z13= subs(diff(K,'x'),{t},{x});
end
if isempty (subs(diff(z13),{x},{a}))=1
z1313=0;
else
z1313=subs(diff(z13),{x},{a});
end
if isempty(subs(diff(subs(K,{x,t},{x,x}),3),{x},{a}))=1
z14=0;
else
z14=subs(diff(subs(K,{x,t},{x,x}),3),{x},{a});
end
if isempty(subs(diff(subs(K,{x,t},{x,x}),2),{x},{a}))=1
z15=0;
else
z15=subs(diff(subs(K,{x,t},{x,x}),2),{x},{a});
end
if isempty(subs(diff(subs(K,{x,t},{x,x})),{x},{a}))=1
z16=0;
else
z16=subs(diff(subs(K,{x,t},{x,x})),{x},{a});
end”
“d4u(1)=subs(z8,a)+subs(subs(z9,{x,t},{x,x}),{x},{a})*u(1)+z1010*u(1)+subs(
subs(z11,{x,t},{x,x}),{x},{a})*du(1)+subs(diff(z1212,'x',2),{x},{a})*u(1)+2*z
1313*du(1)+subs(subs(diff(K,'x'),{x,t},{x,x}),{x},{a})*d2u(1)+z14*u(1)+3*z1
5*u(1)+2*z16*d2u(1)+subs(K,{x,t},{a,a})*d3u(1); %%Q4 %%
a(1)=d4u(1); b(1)=-d3u(1);
c(1)= du(1)+d3u(1); d(1)=(1/2)*(d2u(1)+d4u(1));
e(1)=u(1)-d4u(1);
for i=1:n

```



$$u(i)=a(i)*\cos(h)+b(i)*\sin(h)+h*c(i)+d(i)*h^2+e(i);$$

$$du(i)=-a(i)*\sin(h)+b(i)*\cos(h)+c(i)+2*d(i)*h;$$

$$d2u(i)=-a(i)*\cos(h)-b(i)*\sin(h)+2*d(i);$$

$$d3u(i)=a(i)*\sin(h)-b(i)*\cos(h);$$

$$d4u(i)=a(i)*\cos(h)+b(i)*\sin(h);$$

$$a(i+1)=d4u(i);b(i+1)=-d3u(i);$$

$$c(i+1)=du(i)+d3u(i); d(i+1)=(1/2)*(d2u(i)+d4u(i));$$

$$e(i+1)=u(i)-d4u(i);$$

end

$$err=abs(u-\text{subs}(ex,h:h:1));''$$

Notes about some functions used in the MATLAB code:

- 1) **diff Function:** $Y = \text{diff}(X,n)$ calculates the nth difference by applying the $\text{diff}(X)$ operator recursively n times.
- 2) **Isempty Function:** $\text{isempty}(A)$ Determine whether A is empty returns logical 1 (true) if A is empty, and logical 0 (false) if A is not empty.
- 3) **subs Function:** Symbolic substitution $s_{\text{new}} = \text{subs}(s,\text{old},\text{new})$ returns a copy of s in which all instances of old have been replaced by new, and then evaluates s. s is an expression of symbolic scalar variables or a symbolic function, whereas old specifies the symbolic scalar variables or symbolic function to be substituted. If both old and new are identically sized vectors or cell arrays, the subs Function replaces each element of old with its corresponding element in new. If old is a scalar and new is a vector or matrix, then $\text{subs}(s,\text{old},\text{new})$ replaces all occurrences of old in s with new, element-wise. All constant terms in s are replaced by the constant multiplied by an all-ones vector or matrix.

VIE's Solution Examples:

Example 1: prove that the function Fig.1:

$$\varphi(x) = \frac{1}{(1+x^2)^{3/2}}$$

Is solution for the VIE:

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

$$f(x) = \frac{1}{1+x^2}, \quad K(x, t) = \frac{1}{1+x^2} \quad \lambda = 1$$

Solution:- Let's start by substituting $\varphi(x)$ in the RHS (right hand side) of the VIE:

$$\begin{aligned} \varphi(t) = \frac{1}{(1+t^2)^{3/2}} \Rightarrow \varphi(x) &= \frac{1}{1+x^2} - \int_0^x \left(\frac{1}{1+t^2} \right) \left(\frac{1}{(1+t^2)^{3/2}} \right) dt = \frac{1}{1+x^2} - \\ &\frac{1}{1+x^2} \left[-\frac{1}{(1+t^2)^{1/2}} \right]_{t=0}^{t=x} \\ &= \frac{1}{1+x^2} - \frac{1}{1+x^2} \left[-\frac{1}{(1+x^2)^{1/2}} + 1 \right] = \frac{1}{1+x^2} + \frac{1}{(1+x^2)^{1+1/2}} - \frac{1}{1+x^2} \\ &= \frac{1}{(1+x^2)^{3/2}} = \varphi(x) \end{aligned}$$

Which is mean that $\varphi(x) = \frac{1}{(1+x^2)^{3/2}}$ is a solution for the mentioned VIE.

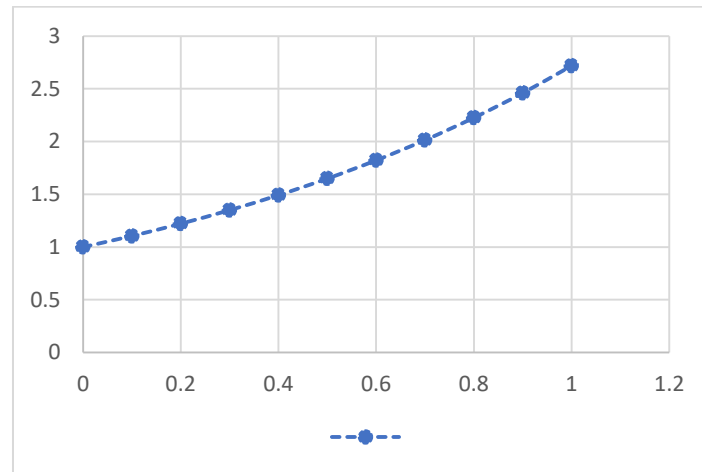


Figure 1: the function $\varphi(x) = \frac{1}{(1+x^2)^{3/2}}$

Example 2: show that $\varphi(x) = e^x$, Fig.2 is a solution for the VIE:

$$\varphi(x) = 1 + \int_0^x e^t dt$$

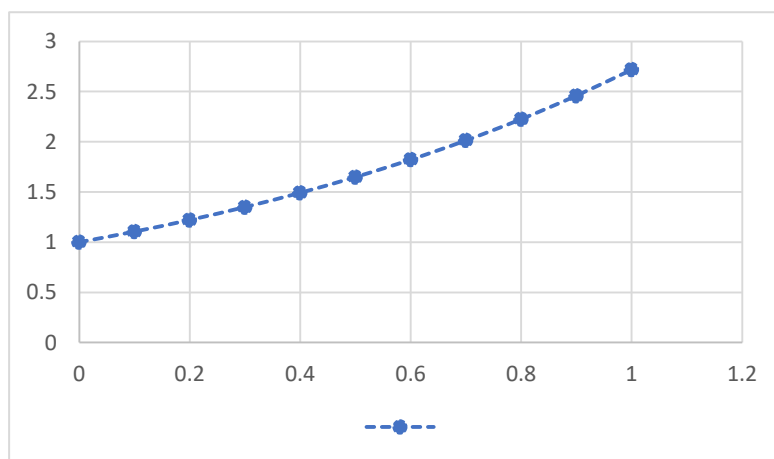
$$f(x) = 1, \quad K(x, t) = 1 \quad \lambda = 1$$

By substituting $\varphi(x) = e^x$ in the VIE type two:

$$\varphi(x) = 1 + \int_0^x e^t dt = 1 + [e^t]_0^x = 1 + (e^x - e^0) = 1 + e^x - 1 = e^x$$

Figure 2: the function $\varphi(x) = 1 + \int_0^x e^t dt$

Example 3: Let's have the following VIE's of the second type :



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

The exact solution is: $\varphi(x) = \sin x$

$$a = 0, b = 1, 0 \leq x \leq 1 \quad \text{for: } n = 10 \Rightarrow h = \frac{b-a}{n} = \frac{1-0}{10} = 0.1$$

$$x_i = t_i = t_0 + i * h \quad i=0,1,2,\dots,n=10$$

$$x_n = t_n = (t_0 + n * h) = b = 1$$

$$f(x) = x, \quad K(x, t) = (t - x), \quad \lambda = 1$$

$$Q'(x) = \varphi'(x) = (x)' + \frac{\partial}{\partial x} \int_0^x (t - x)\varphi(t)dt$$

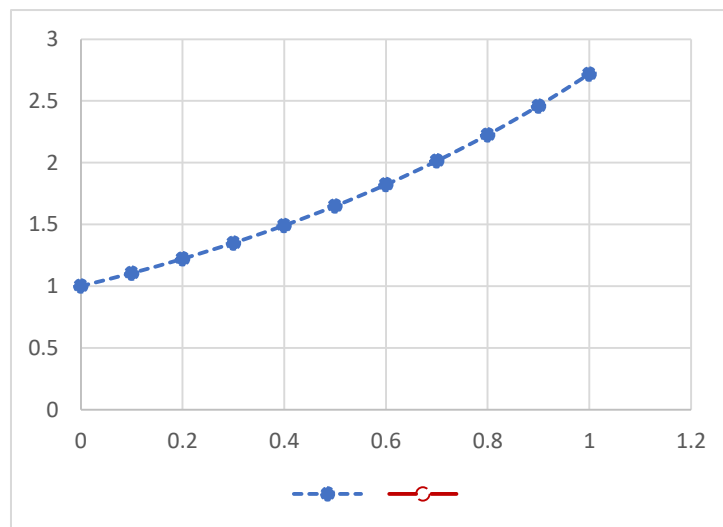
The next Table1 is showing the comparison between the exact solution and the solution using the quadratic spline:

Table1.

x	exact solution = sin(x)	Quadratic spline	error
0	0	0	0.00E+00
0.1	0.99833416647	0.998334166	0.00E+00
0.2	0.19866933080	0.198669331	0.00E+00
0.3	0.29552020666	0.295520207	0.00E+00

0.4	0.38941834231	0.389418342	9.99E-16
0.5	0.47942553860	0.479425539	0.00E+00
0.6	0.56464247340	0.564642473	0.00E+00
0.7	0.64421768724	0.644217687	0.00E+00
0.8	0.71735609090	0.717356091	0.00E+00
0.9	0.78332690963	0.78332691	0.00E+00
1	0.84147098481	0.841470985	9.99E-16

The below figure show both the exact solution and the Quadratic spline solution (Fi.(3)):



(Figure 3)

Example 4: Let's have the following VIE's of the second type:

$$\varphi(x) = x + e^x + x^2 - \frac{1}{2}x^4 - x^2 e^x + \int_0^x x^2 \varphi(t) dt \quad ; 0 \leq x \leq 1$$

The exact solution is $\varphi(x) = x + e^x$

$$a = 0, b = 1, 0 \leq x \leq 1 \quad \text{for: } n = 10 \Rightarrow h = \frac{b-a}{n} = \frac{1-0}{10} = 0.1$$

$$x_i = t_i = t_0 + i * h \quad i=0,1,2,\dots,n=10$$

$$x_n = t_n = (t_0 + n * h) = b = 1$$

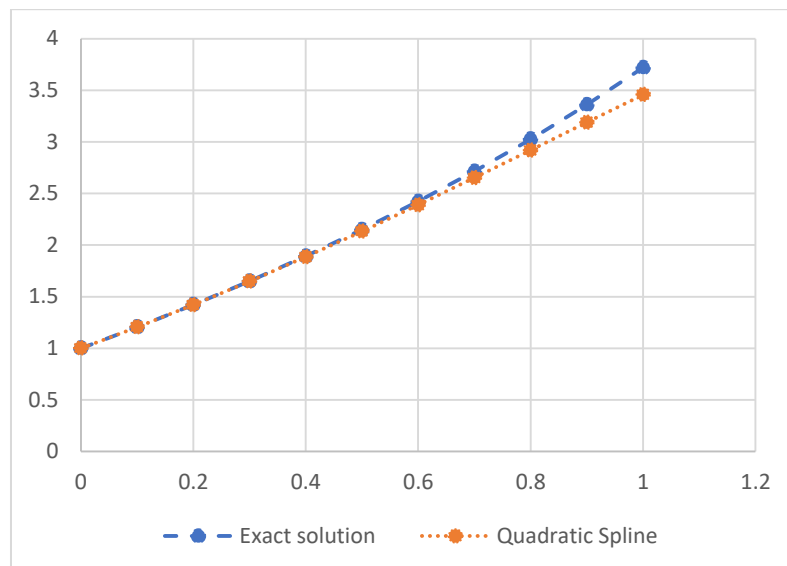
$$f(x) = x + e^x + x^2 - \frac{1}{2}x^4 - x^2 e^x, \quad K(x, t) = x^2, \quad \lambda = 1$$

$$Q'(x) = \varphi'(x) = f'(x) + \frac{\partial}{\partial x} \int_0^x x^2 \cdot \varphi(t) dt$$

x	Exact solution	Quadratic Spline	error
0	1	1	0.0000000

0.1	1.205170918	1.205145757	0.0000252
0.2	1.421402758	1.42099778	0.0004050
0.3	1.649858808	1.647797348	0.0020615
0.4	1.891824698	1.885276688	0.0065480
0.5	2.148721271	2.132661652	0.0160596
0.6	2.4221188	2.388679452	0.0334393
0.7	2.713752707	2.651571376	0.0621813
0.8	3.025540928	2.919110362	0.1064306
0.9	3.359603111	3.18862349	0.1709796
1	3.718281828	3.457017486	0.2612643

The next figure show the difference between the Exact solution and the Quadratic Spline (Fig.(4)):



(Figure 4)

Example 5: Let's have the following VIE's of the second type:

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

The exact solution is $\varphi(x) = \cos(x)$

$$a = 0, b = 1, 0 \leq x \leq 1 \quad \text{for: } n = 10 \Rightarrow h = \frac{b-a}{n} = \frac{1-0}{10} = 0.1$$

$$x_i = t_i = t_0 + i * h \quad i=0,1,2,\dots,n=10$$

$$x_n = t_n = (t_0 + n * h) = b = 1$$

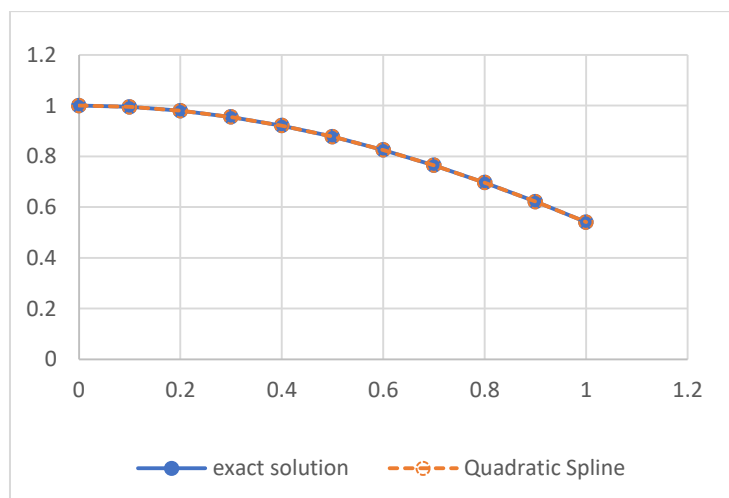
$$f(x) = 1, \quad K(x, t) = (t-x), \quad \lambda = 1$$

$$Q'(x) = \varphi'(x) = f'(x) + \frac{\partial}{\partial x} \int_0^x (t-x) \cdot \varphi(t)dt$$

After following the steps, we can get the below table that show both Exact solution and the Quadratic Non-polynomial Solution:

x	exact solution	Quadratic Spline	error
0	1	1	0.00000
0.1	0.995004165	0.995004165	0.00000
0.2	0.980066578	0.980066578	0.00000
0.3	0.955336489	0.955336489	0.00000
0.4	0.921060994	0.921060994	0.00000
0.5	0.877582562	0.877582562	0.00000
0.6	0.825335615	0.825335615	0.00000
0.7	0.764842187	0.764842187	0.00000
0.8	0.696706709	0.696706709	0.00000
0.9	0.621609968	0.621609968	0.00000
1	0.540302306	0.540302306	0.00000

The Below Graphic show the also both the exact solution and the quadratic solution (Fig.(5)):



(Figure 5)

Conclusion:

In this research, we have studied Volterra's integrative equations of the second using the infinite Quadratic Spline Non-polynomial function, and the solution has been programmed by MATLAB which allow us to insert the Integral Equation and the Code can give us both the Approximate solution and the Error which is the difference between the Exact Solution and the Approximate Solution.



The importance of this research come from the importance of The VIE of second type and its different applications such as the transmission of radiant energy and the ripple of the rope, membranes and axle.

Authors' declaration:

- Conflicts of Interest: None.
- We hereby confirm that all the Figures and Tables in the manuscript are mine ours.

The author has signed an animal welfare statement.

Authors sign on ethical consideration's approval

Authors' contributions statement:

This work was carried out in collaboration between all authors. Douaa Kh. Abid diagnosis the cases then collected the samples and doing the equations. Zahraa A. Ali wrote the introduction and the definition of the research problem. wrote and edited the manuscript with revisions idea. Douaa Kh. Abid analysis the data with revisions idea. All authors read and approved the final manuscript.

References:

1. RB Guenther, JW Lee .Partial differential equations of mathematical physics and integral equations– 1996
2. I.G.Burova, “On left integro-differential splines and Cauchy problem,” International Journal of Mathematical Models and Methods in Applied Sciences.vol.9, pp. 683-690, 2015.
3. R.P Ranwal., Linear Integral Equations, Dover, Second Edition Springer science LLc , 2013
4. A.M.Wazwaz,, A First Course in Integral Equations, World Scientific: Singapore, 1997.
5. H.T.H Piaggio, An Elementary Treatise on Differential Equations and their Applications, G. Bell and Sons, Ltd.: London, 1920.
6. F.G.Tricomi,, Integral Equations, Dover publications ,Inc.: NewYork,2012 .
7. D.G.Zill., Advanced Engineering Mathematics, sixth edition Loyola University, 2016.
8. Rahman, M., Mathematical Methods with Applications, WIT Press: Southampton, UK 2010.
9. Computer Physics Communications, Non-polynomial spline method for solving Bratu's problem, Volume 181, Issue 11, November 2010, Pages 1868-1872.