Splines of the Fourth Order Approximation and the Volterra Integral Equations

I.G.BUROVA, A.G.DORONINA, D.E.ZHILIN
Department of Computational Mathematics
St. Petersburg State University
7/9 Universitetskaya nab., St.Petersburg, 199034
RUSSIA

Abstract: - This paper is a continuation of a series of papers devoted to the numerical solution of integral equations using local interpolation splines. The main focus is given to the use of splines of the fourth order of approximation. The features of the application of the polynomial and non-polynomial splines of the fourth order of approximation to the solution of Volterra integral equation of the second kind are discussed. In addition to local splines of the Lagrangian type, integro-differential splines are also used to construct computational schemes. The comparison of the solutions obtained by different methods is carried out. The results of the numerical experiments are presented.

1 Introduction

A new approach of the development of numerical solutions of the integral equations are often connected with the application of interpolation. The well-known methods of solving Volterra integral equations of the second kind include, first of all, the trapezium method and the Simpson method. The trapezium method is quite simple to use. This should be attributed to the advantages of this method. As it is known, when verifying the result of a solution, several different methods are often used. Therefore, it is advisable to apply several different methods to solve the same equation. It should be noted that different types of splines are quite often used when solving interpolation problems. Let us briefly recall one of the main reasons for their widespread use.

Let P_n be the interpolation polynomial that solves the Lagrange interpolation problem when we use the values of the Runge function $f = \frac{1}{1+25\,x^2}$ in the equidistant nodes in the interval $[-1,1]$ including the ends. As it is known (the fact was established by Runge in 1901), the following relation is true:

$$\| f - P_n \| \to \infty \quad \text{when} \quad n \to +\infty.$$

Thus, the sequence of interpolation polynomials P_n does not tend to the Runge function when n tends to infinity. Thus, when solving various problems of mathematical physics, splines are widely used. It

should be mentioned that papers [1]-[7] are among a variety of papers about numerical methods in solving the Volterra integral equations. In [1], the authors discuss the super convergence of the "interpolated" collocation solutions for weak singular Volterra integral equations of the second kind. In paper [2], the 6th order Runge-Kutta with a seven stage method for finding the numerical solution of the Volterra integro-differential equation is considered. In paper [2], the integral term in the Volterra integro-differential equation was approximated using the Lagrange interpolation numerical method. In paper [3], a numerical solution of the important weak singular type of the Volterra-Fredholm integral equations is provided. In [3], a new computational method based on the special B-spline tight framelets is presented and used to the numerical scheme. In paper [4], the numerical solution to a class of weak singular Volterra integral equations is discussed. In paper [4], the fractional Lagrange interpolation is applied to deal with the singularity of the solution, and efficient fractional collocation boundary value methods are developed. In paper [5], the method of radial basis functions is used for solving the Volterra integral equation. A new collocation technique for numerical solution of Fredholm, Volterra and mixed Volterra-Fredholm integral equations of the second kind was introduced in paper [6]. In paper [7], the quadratic rule for the

numerical solution of linear and nonlinear two-dimensional Fredholm integral equations based on spline quasi-interpolant was studied.

The use of local polynomial and non-polynomial splines allows us to construct new methods for solving the Volterra integral equation of the second kind. Paper [9] discusses the use of the polynomial and non-polynomial splines of the third order of approximation. These splines have shown good numerical stability and are suitable for constructing solutions on both a uniform and non-uniform mesh of nodes.

In this paper, the local spline approximations [8] are used to construct calculation formulas for solving the Volterra integral equation. Here we use both the polynomial and non-polynomial splines of the fourth order of approximation. Section 2 discusses the properties of the local polynomial, polynomial-trigonometrical splines and the integro-differential splines of the fourth order of approximation. Section 3 considers the use of the construction of the solution of the integral equation, of not only the polynomial and non-polynomial local splines of the Lagrangian type, but also the integro-differential splines of the fourth order of approximation. As a result, the integro-differential splines give a smaller error, but in this case it is assumed that the values of the integrals over the grid intervals are known. Here a modification of the Simpson method is also proposed, where the spline approximations are applied.

2 Problem Formulation

Let $\{x_j\}$ be the set of nodes on the interval $[a, b]$. Suppose that the system of functions $\varphi_i(x)$ forms a Chebyshev system. We define the basic splines on the interval $[x_j, x_{j+1}] \subset [a, b]$, solving the system of equations (following Professor S.G. Mikhlin, we refer to this system as the fundamental relations)

$$\sum_{k=k_{j0}}^{k_{jm}} \varphi_i(x_k) w_k(x) = \varphi_i(x), \qquad (1)$$

$$x \in [x_j, x_{j+1}], \qquad i = 0, \dots, m.$$

Here $k_{jm} - k_{j0} = m + 1$.

We consider that the determinant of this system is different from zero. With a different choice of functions $\varphi_i(x)$ and integer k_{j0}, k_{jm}, we obtain basic splines suitable for approximation at the beginning of the interval $[a, b]$, in the middle of the

interval or at the end of the interval $[a, b]$. In the case of a polynomial system x^i, fundamental system (1) has the form

$$\sum_{k=k_{j0}}^{k_{jm}} x_k^i w_k(x) = x^i, i = 0, \dots, m,$$

$$x \in [x_j, x_{j+1}].$$

In this paper we focus on the splines of the fourth order of approximation. Therefore, in our case, we consider $m = 3$. Thus, there are four equations and four unknown basis functions $w_k(x)$.

Let us highlight the important special cases. Suppose that the support of the basis spline occupies the interval $[x_{j-3}, x_{j+1}]$. The splines w_j suitable for approximation near the left end of the interval $[a, b]$ can be obtained when $k_{j0} = j$, $k_{jm} = j + 3$. In this case, the condition $j = 0, \dots, n - 3$, must be satisfied. Suppose that the support of the spline occupies the interval $[x_{j-2}, x_{j+2}]$. Splines w_j suitable for approximation in the middle of the interval $[a, b]$ can be obtained when $k_{j0} = j - 1$, $k_{jm} = j + 2$. In this case, the condition $j = 1, \dots, n - 2$, must be satisfied. Suppose that the support of the spline occupies the interval $[x_{j-1}, x_{j+3}]$. Splines w_j suitable for approximation near the right end of the interval $[a, b]$ can be obtained when $k_{j0} = j - 2$, $k_{jm} = j + 1$. In this case, the condition $j = 2, \dots, n - 1$, must be satisfied.

In the next sub-section, we will take a closer look at cubic polynomial splines.

2.1 Cubic Polynomial Splines

At first, we recall the features of the approximation of the functions with the splines near the right end of the interval $[a, b]$, near the left end of the interval $[a, b]$, and at the middle of the interval. As an example, we write out the formula for the basis spline, which is used to approximate in the middle of the interval $[a, b]$. The formula of the basis spline $\omega_j^M(x)$ can be written as follows:

$$\omega_j^M(x) = \frac{(x - x_{j+1})(x - x_{j+2})(x - x_{j+3})}{(x_j - x_{j+1})(x_j - x_{j+2})(x_j - x_{j+3})},$$
$$x \in [x_{j+1}, x_{j+2}],$$

$$\omega_j^M(x) = \frac{(x - x_{j+1})(x - x_{j-1})(x - x_{j+2})}{(x_j - x_{j+1})(x_j - x_{j-1})(x_j - x_{j+2})},$$
$$x \in [x_j, x_{j+1}],$$

$$\omega_j^M(x) = \frac{(x - x_{j-1})(x - x_{j-2})(x - x_{j+1})}{(x_j - x_{j-1})(x_j - x_{j-2})(x_j - x_{j+1})},$$

$$\omega_j^M(x) = \frac{(x-x_{j-1})(x-x_{j-2})(x-x_{j-3})}{(x_j-x_{j-1})(x_j-x_{j-2})(x_j-x_{j-3})},$$
$$x \in [x_{j-2}, x_{j-1}],$$

$$\omega_j^M(x) = 0, \ x \notin [x_{j-2}, x_{j+2}].$$

Now we write out the formulas of the basis splines that form the continuous polynomial approximation in the interval $x \in [x_j, x_{j+1}] \subset [a, b]$. The continuous polynomial approximation $U_j^R(x)$ near the left end of the interval $[a, b]$ uses the right basis spline $\omega_j^R(x)$ of the form:

$$\omega_j^R(x) = \frac{(x-x_{j+1})(x-x_{j+2})(x-x_{j+3})}{(x_j-x_{j+1})(x_j-x_{j+2})(x_j-x_{j+3})},$$
$$x \in [x_j, x_{j+1}],$$
$$\omega_{j+1}^R(x) = \frac{(x-x_j)(x-x_{j+2})(x-x_{j+3})}{(x_{j+1}-x_j)(x_{j+1}-x_{j+2})(x_{j+1}-x_{j+3})},$$
$$x \in [x_j, x_{j+1}],$$
$$\omega_{j+2}^R(x) = \frac{(x-x_j)(x-x_{j+1})(x-x_{j+3})}{(x_{j+2}-x_j)(x_{j+2}-x_{j+1})(x_{j+2}-x_{j+3})},$$
$$x \in [x_j, x_{j+1}],$$
$$\omega_{j+3}^R(x) = \frac{(x-x_j)(x-x_{j+1})(x-x_{j+2})}{(x_{j+3}-x_j)(x_{j+3}-x_{j+1})(x_{j+3}-x_{j+2})},$$
$$x \in [x_j, x_{j+1}].$$

The approximation with these basis splines can be written in the form:

$$U_j^R(x) = u(x_j)\omega_j^R(x) + u(x_{j+1})\omega_{j+1}^R(x) + u(x_{j+2})\omega_{j+2}^R(x) + u(x_{j+3})\omega_{j+3}^R(x).$$

The continuous polynomial approximation $U_j^L(x)$ near the right end of the interval $[a, b]$ uses the left basis spline $\omega_j^L(x)$ of the form:

$$\omega_{j-2}^L(x) = \frac{(x-x_{j-1})(x-x_j)(x-x_{j+1})}{(x_{j-2}-x_{j-1})(x_{j-2}-x_j)(x_{j-2}-x_{j+1})},$$
$$x \in [x_j, x_{j+1}],$$
$$\omega_{j-1}^L(x) = \frac{(x-x_{j-2})(x-x_j)(x-x_{j+1})}{(x_{j-1}-x_{j-2})(x_{j-1}-x_j)(x_{j-1}-x_{j+1})},$$
$$x \in [x_j, x_{j+1}],$$
$$\omega_j^L(x) = \frac{(x-x_{j-2})(x-x_{j-1})(x-x_{j+1})}{(x_j-x_{j-2})(x_j-x_{j-1})(x_j-x_{j+1})},$$
$$x \in [x_j, x_{j+1}],$$

$$\omega_{j+1}^L(x) = \frac{(x-x_{j-2})(x-x_{j-1})(x-x_j)}{(x_{j+1}-x_{j-2})(x_{j+1}-x_{j-1})(x_{j+1}-x_j)},$$
$$x \in [x_j, x_{j+1}].$$

The approximation on the $[x_j, x_{j+1}]$ with these basis splines can be written in the form:

$$U_j^L(x) = u(x_{j-2})\omega_{j-2}^L(x) + u(x_{j-1})\omega_{j-1}^L(x) + u(x_j)\omega_j^L(x) + u(x_{j+1})\omega_{j+1}^L(x).$$

The continuous polynomial approximation $U_j^M(x)$ in the middle of the interval $[a, b]$ uses the middle basis spline $\omega_j^M(x)$ of the form:

$$\omega_{j-1}^M(x) = \frac{(x-x_j)(x-x_{j+1})(x-x_{j+2})}{(x_{j-1}-x_j)(x_{j-1}-x_{j+1})(x_{j-1}-x_{j+2})},$$
$$x \in [x_j, x_{j+1}],$$
$$\omega_j^M(x) = \frac{(x-x_{j-1})(x-x_{j+1})(x-x_{j+2})}{(x_j-x_{j-1})(x_j-x_{j+1})(x_j-x_{j+2})},$$
$$x \in [x_j, x_{j+1}],$$
$$\omega_{j+1}^M(x) = \frac{(x-x_{j-1})(x-x_j)(x-x_{j+2})}{(x_{j+1}-x_{j-1})(x_{j+1}-x_j)(x_{j+1}-x_{j+2})},$$
$$x \in [x_j, x_{j+1}],$$
$$\omega_{j+2}^M(x) = \frac{(x-x_{j-1})(x-x_j)(x-x_{j+1})}{(x_{j+2}-x_{j-1})(x_{j+2}-x_j)(x_{j+2}-x_{j+1})},$$
$$x \in [x_j, x_{j+1}].$$

The approximation on the $[x_j, x_{j+1}]$ with these basis splines can be written in the form:

$$U_j^M(x) = u(x_{j-1})\omega_{j-1}^M(x) + u(x_j)\omega_j^M(x) + u(x_{j+1})\omega_{j+1}^M(x) + u(x_{j+2})\omega_{j+2}^M(x).$$

The approximation properties of these basis splines are well known. Let us denote $\| u^{(\alpha)} \|_{[c,d]} = \max_{[c,d]} |u^{(\alpha)}(x)|$. The following theorem can be easily proved (see also [8]).

Theorem 1. Let $u \in C^4[a, b]$. $x_j = a + jh, j = 0, 1, \dots, n$, $h = \frac{b-a}{n}, n \geq 3$. To approximate the function $u(x)$, $x \in [x_j, x_{j+1}]$, with the left and right splines, the following inequalities are valid:

$$\left| u(x) - U_j^L(x) \right| \leq Kh^4 \| u^{(4)} \|_{[x_{j-2}, x_{j+1}]}, K = 1.$$

$$\left| u(x) - U_j^R(x) \right| \leq Kh^4 \| u^{(4)} \|_{[x_j, x_{j+3}]}, \ K = 1.$$

To approximate the function $u(x)$, $x \in [x_j, x_{j+1}]$, with the middle splines, the following inequality is valid:

$$\left| u(x) - U_j^M(x) \right| \leq Kh^4 \parallel u^{(4)} \parallel_{[x_{j-1}, x_{j+2}]},$$

$$K = 0.5625.$$

Proof. It is easy to notice that U_j^R is an interpolation polynomial, and $x_j, x_{j+1}, x_{j+2}, x_{j+3}$ are the interpolation nodes, and

$$U_j^R(x_j) = u(x_j), \; U_j^R(x_{j+1}) = u(x_{j+1}),$$

$$U_j^R(x_{j+2}) = u(x_{j+2}), \; U_j^R(x_{j+3}) = u(x_{j+3}).$$

Using the remainder term we get

$$u(x) - U_j^R(x) = \frac{u^{(4)}(\tau)}{4!}(x - x_j)(x - x_{j+1})(x - x_{j+2})(x - x_{j+3}), \; \tau \in [x_j, x_{j+3}] \; .$$

We can use $x = x_j + t\,h, t \in [0,1]$. It can be easily calculated that

$$\max_{t \in [0,1]} |t(t-1)(t-2)(t-3)| = 1.$$

It follows that on the uniform grid with step h

$$\max_{x \in [x_j, x_{j+3}]} |u(x) - U_j^R(x)| \leq h^4 \max_{[x_j, x_{j+3}]} |u^{(4)}|.$$

Thus, we obtain that $K = 1$.
Earlier in paper [8] quadratic splines were considered.

2.2 Cubic and Quadratic Polynomial Splines
The function $u(x)$, $x \in [x_j, x_{j+1}]$, can be approximated with the left or the right polynomial splines as follows. The left quadratic polynomial spline (see [8]) can be written in the form:

$$U_j^L(x) = u(x_{j-1})\omega_{j-1}^L(x) + u(x_j)\omega_j^L(x)$$
$$+ u(x_{j+1})\omega_{j+1}^L(x), x \in [x_j, x_{j+1}], \quad (2)$$
where

$$\omega_{j-1}^L(x) = \frac{(x - x_j)(x - x_{j+1})}{(x_{j-1} - x_j)(x_{j-1} - x_{j+1})},$$

$$\omega_j^L(x) = \frac{(x - x_{j+1})(x - x_{j-1})}{(x_j - x_{j+1})(x_j - x_{j-1})},$$

$$\omega_{j+1}^L(x) = \frac{(x - x_j)(x - x_{j-1})}{(x_{j+1} - x_j)(x_{j+1} - x_{j-1})}.$$

The function $u(x)$, $x \in [x_j, x_{j+1}]$, can be approximated by the right polynomial spline (see [8, 9]) using the form:

$$U_j^R(x) = u(x_j)\omega_j^R(x) + u(x_{j+1})\omega_{j+1}^R(x) + u(x_{j+2})\omega_{j+2}^R(x), \quad x \in [x_j, x_{j+1}], \quad (3)$$

where

$$\omega_j^R(x) = \frac{(x - x_{j+1})(x - x_{j+2})}{(x_j - x_{j+1})(x_j - x_{j+2})},$$

$$\omega_{j+1}^R(x) = \frac{(x - x_{j+2})(x - x_j)}{(x_{j+1} - x_{j+2})(x_{j+1} - x_j)},$$

$$\omega_{j+2}^R(x) = \frac{(x - x_{j+1})(x - x_j)}{(x_{j+2} - x_{j+1})(x_{j+2} - x_j)}.$$

The approximation properties of these basic splines are well studied. The following theorem was proved in [8].

Theorem 2. Let $u \in C^3[a, b]$. To approximate the function $u(x)$, $x \in [x_j, x_{j+1}]$, by spline (2), the following inequality is valid:

$$\left| u(x) - U_j^L(x) \right| \leq Kh^3 \parallel u''' \parallel_{[x_{j-1}, x_{j+1}]}.$$

To approximate the function $u(x)$, $x \in [x_j, x_{j+1}]$, by spline (3), the following inequality is valid:

$$\left| u(x) - U_j^R(x) \right| \leq Kh^3 \parallel u''' \parallel_{[x_j, x_{j+2}]}.$$

Proof. It is easy to notice that U_j^R is an interpolation polynomial of the third degree, and x_j, x_{j+1} are the interpolation nodes, $U_j^R(x_j) = u(x_j), U_j^R(x_{j+1}) = u(x_{j+1}), U_j^R(x_{j+2}) = u(x_{j+2})$. Using the remainder term we get

$$u(x) - U_j^R(x)$$

$$= \frac{u'''(\tau)}{3!}(x - x_j)(x - x_{j+1})(x - x_{j+2}).$$

It follows that

$$\left| u(x) - U_j^R(x) \right| \leq 0.0625h^3 \max_{[x_j, x_{j+2}]} |u'''|.$$

Thus, $K = 0.0625$.

The statements for $U_j^L(x)$ can be proved similarly. The Theorem is proved.

Note. Theorem 1 and Theorem 2 give estimates of the approximation errors when quadratic and cubic splines are used. These estimates are asymptotic ones. The following question may be asked: Can the use of cubic splines always give a smaller approximation error in absolute value than the use

of quadratic splines? In general, the answer to this question is negative. Consider an approximation using the interpolation polynomials P_n for the Runge function $f(x) = \frac{1}{1+25\,x^2}$ on the interval [-1, 1] at equidistant nodes. The estimation of the error approximation when the quadratic splines were used includes the third derivative of the function, while the estimation of the error approximation when the cubic splines were used, includes the fourth derivative of the function. Fig. 1 shows the graph of the third derivative of the Runge function, and Fig. 2 shows the graph of the fourth derivative of the Runge function on the interval $[-1, 1]$. It is easy to choose step h of the uniform grid of nodes, in a way, that the approximation of the Runge function with the quadratic splines will give a smaller error in absolute value than the approximation with the right cubic splines (last column in Table 1). Table 1 shows the errors of the approximations with the quadratic polynomial splines, the middle and the right cubic polynomial splines.

Table 1. The errors of the approximation with the quadratic polynomial splines, middle cubic polynomial splines, right cubic polynomial splines, $h = 0.2$

Function	Quadratic polynomial splines	Middle Cubic polynomial splines	Right cubic polynomial splines
$\frac{1}{1+25\,x^2}$	0.0889	0.0200	0.162
$\sin(5x)$	0.0606	0.0215	0.0373

Fig.3 shows the graph of the error of approximation of the Runge function with quadratic splines when the grid step $h = 0.2$ was used.
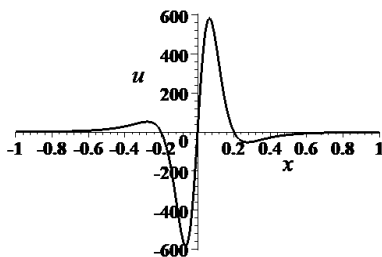
Fig.1: The graph of the third derivative of the Runge function on the interval $[-1, 1]$
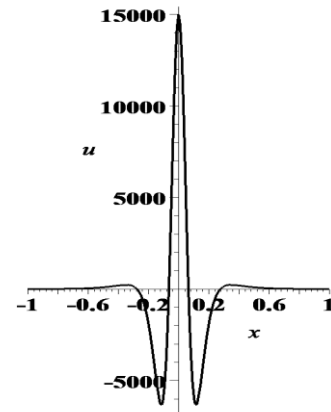

Fig.2: The graph of the fourth derivative of the Runge function on the interval $[-1, 1]$
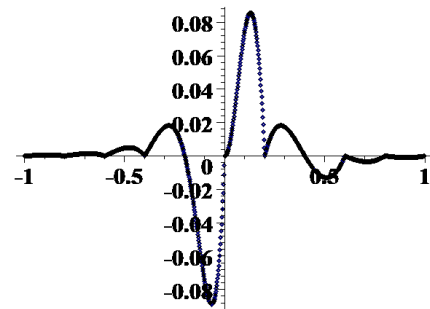

Fig.3: The graph of the error of approximation when the quadratic polynomial splines were used for the approximation of the Runge function with the grid step $h = 0.2$.

Fig. 4. shows the graph of the error of approximation of the function $\sin(5x)$ with the right cubic splines when the grid step $h = 0.2$ was used. Fig. 5. shows the graph of the error of approximation of the function $\sin(5x)$ with the middle cubic splines when the grid step $h = 0.2$ was used.
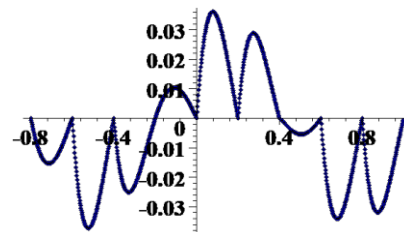

Fig.4: The graph of the approximation error when the right cubic polynomial splines were used for the approximation of the function $\sin(5x),\ h = 0.2$.

Fig. 6. shows the graph of the error of approximation the absolute value of the Runge function with the right cubic splines with the grid step $h = 0.2$. Fig. 7. shows the graph of the error of

approximation of the Runge function with the middle cubic splines with the grid step $h = 0.2$.
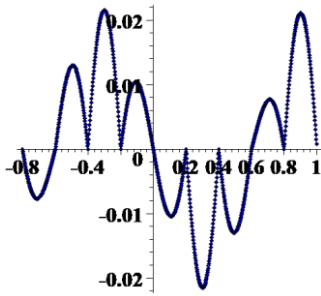

Fig.5: The graph of the approximation error when the middle cubic splines were used for the approximation of the function $\sin(5x)$ with the grid step $h = 0.2$.
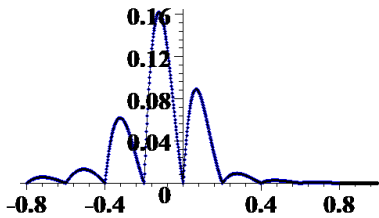

Fig.6: The graph of the approximation error in the absolute value when the right cubic polynomial splines were used for the approximation of the Runge function with the grid step $h = 0.2$.
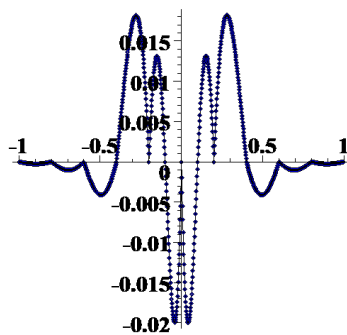

Fig.7:. The graph of the approximation error when the middle cubic polynomial splines were used for the approximation of the Runge function with the grid step $h = 0.2$.

2.2 Non-polynomial Splines

Let us now take $\varphi_1(x) = 1$, $\varphi_2(x) = x$, $\varphi_3(x) = \sin(x)$, $\varphi_4(x) = \cos(x)$. The system of equations for determining the non-polynomial basis functions $w_j(x), x \in [x_j, x_{j+1}]$, takes the form:

$$\sum_{k=k_{j0}}^{k_{jm}} w_k(x) = 1,$$

$$\sum_{k=k_{j0}}^{k_{jm}} x_k w_k(x) = x,$$

$$\sum_{k=k_{j0}}^{k_{jm}} \cos(x_k) w_k(x) = \cos(x),$$

$$\sum_{k=k_{j0}}^{k_{jm}} \sin(x_k) w_k(x) = \sin(x).$$

Let us take $k_{j0} = j - 1$, $k_{jm} = j + 2$. In this case we obtain (when $x = x_j + th$, $t \in [0, 1], x \in [x_j, x_{j+1}]$):

$$w_j(x_j + th) = ((1 - t)\sin(3h) + (1 + t)\sin(h)$$
$$+ 2\sin(h(t - 2)) + (t - 2)\sin(2h)$$
$$+ \sin(h(1 + t)) - 3\sin(h(t - 1)))/$$
$$(5\sin(h) + \sin(3h) - 4\sin(2h)),$$

$$w_{j+1}(x_j + th) = (t \sin(3h) + (2 - t)\sin(h)$$
$$+ 3\sin(th) - \sin(th - 2h)$$
$$- (t + 1)\sin(2h) - 2\sin(h + th))/$$
$$(5\sin(h) + \sin(3h) - 4\sin(2h)),$$

$$w_{j+2}(x_j + th) = (\sin(h + th) + \sin(th - h)$$
$$- 2\sin(th) - t\sin(2h) + 2t\sin(h))/(5\sin(h)$$
$$+ \sin(3h) - 4\sin(2h)),$$

$$w_{j-1}(x_j + th) = (-\sin(th - 2h) - \sin(th)$$
$$+ 2\sin(th - h) + (2 - 2t)\sin(h)$$
$$+ (t - 1)\sin(2h))/$$
$$(5\sin(h) + \sin(3h) - 4\sin(2h)).$$

These splines are useful in the middle of the interval $[a, b]$. Fig. 8 shows the graph of the error of approximation of the function $\sin(5x)$ with the middle non-polynomial splines with the grid step $h = 0.2$.
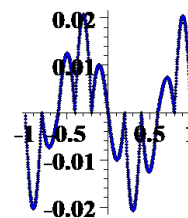

Fig.8: The graph of the approximation error when the middle non-polynomial splines were used for the approximation of the function $\sin(5x)$ with the grid step $h = 0.2$.

Table 2 shows the errors of the approximations with the middle cubic polynomial splines, the middle non-polynomial splines, and the right non-polynomial splines.

Table 2. The errors of the approximation with the middle cubic polynomial splines, middle non-polynomial splines, $h = 0.2$

Function	Middle Cubic polynomial splines	Middle non-polynomial splines	Right non-polynomial splines
$\dfrac{1}{1+25\,x^2}$	0.0200	0.0197	0.161
$\sin(5x)$	0.0215	0.0207	0.0372

As it is known, the integro-differential splines (see [9]) can provide a smaller approximation error. Let $\varphi_0 = 1, \varphi_1 = x, \varphi_2 = x^2, \varphi_3 = x^3$. We obtain the basic splines from the approximation relations

$$\varphi_i(x_j)w_j(x) + \varphi_i(x_{j+1})w_{j+1}(x) +$$
$$\int_{x_j}^{x_{j+1}} \varphi_i(t)dt\, w_j^{<0,1>}(x) + \varphi_i(x_{j-1})w_{j-1}(x) =$$
$$\varphi_i(x), \quad i = 0, 1, 2, 3.$$

On the interval $[x_j, x_{j+1}]$, we construct the polynomial approximation in the form:

$$U_j^{IPol}(x) = u(x_j)w_j(x) + u(x_{j+1})w_{j+1}(x) +$$
$$\int_{x_j}^{x_{j+1}} u(t)dt\, w_j^{<0,1>}(x) + u(x_{j-1})w_{j-1}(x).$$

Considering that $x \in [x_j, x_{j+1}], h = x_{j+1} - x_j, t \in [0,1]$, it is easy to see that the basis functions can be written in the form

$$w_j(x_j + th) = \frac{(t-1)(8t-3)(t+1)}{3},$$

$$w_{j+1}(x_j + th) = \frac{t(10t-7)(t+1)}{6},$$

$$w_j^{<0,1>}(x_j + th) = \frac{4t(t-1)(t+1)}{h},$$

$$w_{j-1}(x_j + th) = \frac{t(2t-1)(t-1)}{6}.$$

Similarly, we can construct a non-polynomial approximation using integro-differential polynomial-trigonometric splines. Let $\varphi_0 = 1, \varphi_1 = x, \varphi_2 = \sin(x), \varphi_3 = \cos(x)$. On the interval

$[x_j, x_{j+1}]$, we construct the polynomial approximation in the form

$$U_j^{Itrig}(x) = u(x_j)w_j(x) + u(x_{j+1})w_{j+1}(x) +$$
$$\int_{x_j}^{x_{j+1}} u(x)dx\, w_j^{<0,1>}(x) + u(x_{j-1})w_{j-1}(x).$$

Considering that $x \in [x_j, x_{j+1}], h = x_{j+1} - x_j, t \in [0,1]$, it is easy to see that the basis functions can be written in the form:

$$w_j(x_j + th) = (2 + 2t + 4\cos(th)$$
$$+ \cos(2h)(2 - 2t) + h\sin(h + th)$$
$$+ \sin(2h)(h - 2th) + 3h\sin(th - h)$$
$$- 4\cos(th - h) - 4\cos(h))/(6 - 8\cos(h)$$
$$+ 2\cos(2h) - 2h\sin(h) + h\sin(2h)),$$

$$w_{j+1}(x_j + th) = -((h - 2th)\sin(h)$$
$$+ \cos(h)(2 + 4t) - 2\cos(th - h)$$
$$- h\sin(th + h) + 3h\sin(th) + 2\cos(th)$$
$$- 2t\cos(2h) - 2t - 2)/(2\cos(2h) -$$
$$- 2h\sin(h) + h\sin(2h) - 8\cos(h) + 6),$$

$$w_j^{<0,1>}(x_j + th) = \frac{2t\sin(h) - 2\sin(th)}{h\sin(h) - 2 + 2\cos(h)}.$$

Table 3 shows the errors of the approximations with the polynomial integro-differential splines, and non-polynomial integro-differential splines.

Table 3. The errors of the approximation with the polynomial integro-differential splines, and non-polynomial integro-differential splines, $h = 0.2$

Function	Polynomial integro-differential splines	Non-polynomial integro-differential splines
$\dfrac{1}{1+25\,x^2}$	0.0184	0.0183
$\sin(5x)$	0.00307	0.00295

Fig. 9 shows the graph of the error of approximation of the Runge function with the integro-differential polynomial splines with the grid step $h = 0.2$. Fig. 10 shows the graph of the error of approximation of the Runge function with the integro-differential non-polynomial splines with the grid step $h = 0.2.$
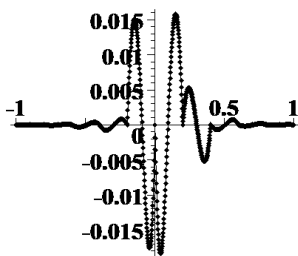
Fig.9: The graph of the approximation error when the integro-differential polynomial splines were used for the approximation of the Runge function with the grid step $h = 0.2$.
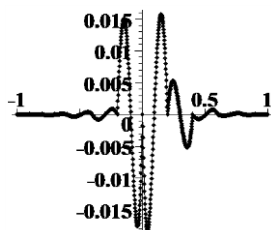

Fig.10: The graph of the approximation error when the integro-differential trigonometrical-polynomial splines were used for the approximation of the Runge function with the grid step $h = 0.2$.

Next, we apply spline approximations to solve the Volterra integral equations of the second kind.

3 Problem Solution of the Volterra Integral Equation

Let n be an integer, and a, b be real. Suppose that a grid of nodes $\{x_j\}$ with step $h = \frac{b-a}{n}$, is constructed on the interval $[a, b]$. Thus, $x_j = a + jh, j = 0, \ldots, n$. Let us consider the numerical solution of the Volterra equation of the second kind. The linear Volterra equation of the second kind has the form:

$$u(x) + \int_a^x K(x, s)u(s)ds = f(x), \quad x, s \in [a, b],$$

where f is a given function, K, f are continues functions and $u(x)$ is an unknown function that needs to be obtained.

3.1 The Numerical Solution

Let us consider the numerical solution of the Volterra equation of the second kind using splines of the fourth order of approximation. We approximate the function $u(x)$ with fourth-order splines. In this case, after discarding the approximation error, we obtain the approximate values of the solution to the integral equation. Let us denote them by $\tilde{u}(x_k)$. First, we have to solve a

system of linear algebraic equations, then we successively determine the approximate values of the unknowns $\tilde{u}(x_k)$.

In the case of a grid consisting of four nodes, the system of equations has the form:

$$\tilde{u}(x_0) = f(x_0),$$

$$\tilde{u}(x_1) + \tilde{u}(x_0) \int_{x_0}^{x_1} K(x_1, s)\, \omega_0^R(s)ds +$$

$$+\tilde{u}(x_1) \int_{x_0}^{x_1} K(x_1, s)\, \omega_1^R(s)ds$$

$$+\tilde{u}(x_2) \int_{x_0}^{x_1} K(x_1, s)\, \omega_2^R(s)ds$$

$$+\tilde{u}(x_3) \int_{x_0}^{x_1} K(x_1, s)\, \omega_3^R(s)ds = f(x_1),$$

$$\tilde{u}(x_2) + \tilde{u}(x_0) \int_{x_0}^{x_1} K(x_2, s)\, \omega_0^R(s)ds +$$

$$+\tilde{u}(x_1) \int_{x_0}^{x_1} K(x_2, s)\, \omega_1^R(s)ds$$

$$+\tilde{u}(x_2) \int_{x_0}^{x_1} K(x_2, s)\, \omega_2^R(s)ds$$

$$+\tilde{u}(x_3) \int_{x_1}^{x_2} K(x_2, s)\, \omega_3^R(s)ds +$$

$$+ \tilde{u}(x_0) \int_{x_1}^{x_2} K(x_2, s)\, \omega_0^M(s)ds +$$

$$+\tilde{u}(x_1) \int_{x_1}^{x_2} K(x_2, s)\, \omega_1^M(s)ds$$

$$+\tilde{u}(x_2) \int_{x_1}^{x_2} K(x_2, s)\, \omega_2^M(s)ds$$

$$+\tilde{u}(x_3) \int_{x_1}^{x_2} K(x_2, s)\, \omega_3^M(s)ds = f(x_2),$$

$$\tilde{u}(x_3) + \tilde{u}(x_0) \int_{x_0}^{x_1} K(x_3, s)\, \omega_0^R(s)ds +$$

$$+\tilde{u}(x_1) \int_{x_0}^{x_1} K(x_3, s)\, \omega_1^R(s)ds$$

$$+\tilde{u}(x_2)\int_{x_0}^{x_1} K(x_3,s)\,\omega_2^R(s)ds$$

$$+\tilde{u}(x_3)\int_{x_1}^{x_2} K(x_3,s)\,\omega_3^R(s)ds$$

$$+\tilde{u}(x_0)\int_{x_1}^{x_2} K(x_3,s)\,\omega_0^M(s)ds +$$

$$+\tilde{u}(x_1)\int_{x_1}^{x_2} K(x_3,s)\,\omega_1^M(s)ds$$

$$+\tilde{u}(x_2)\int_{x_1}^{x_2} K(x_3,s)\,\omega_2^M(s)ds$$

$$+\tilde{u}(x_3)\int_{x_1}^{x_2} K(x_3,s)\,\omega_3^M(s)ds$$

$$+\tilde{u}(x_0)\int_{x_2}^{x_3} K(x_3,s)\,\omega_0^l(s)ds +$$

$$+\tilde{u}(x_1)\int_{x_2}^{x_3} K(x_3,s)\,\omega_1^l(s)ds$$

$$+\tilde{u}(x_2)\int_{x_2}^{x_3} K(x_3,s)\,\omega_2^l(s)ds$$

$$+\tilde{u}(x_3)\int_{x_2}^{x_3} K(x_3,s)\,\omega_3^l(s)ds = f(x_3).$$

The advantages of the proposed method include the ability to calculate the exact integral $\int_{x_i}^{x_{i+1}} K(x,s)\,\omega_0^R(s)ds$ (without error). However, in case of difficulties with calculating the integral, we can apply a quadrature formula that provides the order of approximation $m, m \geq 4$.

3.2 Numerical Examples
Problem 1. Now we take the equation from paper [6]:

$$u(x) = \exp(-x) + x\exp(x)$$
$$- \int_0^x \exp(x+t)\,u(t)dt\,, x \in [0,1].$$

The exact solution of the integral equation is $u(x) = \exp(-x)$. Figs. 11 and 12 show the errors of the solution of Problem 1 with cubic polynomial splines when $n = 32, 64$, Digits=18.
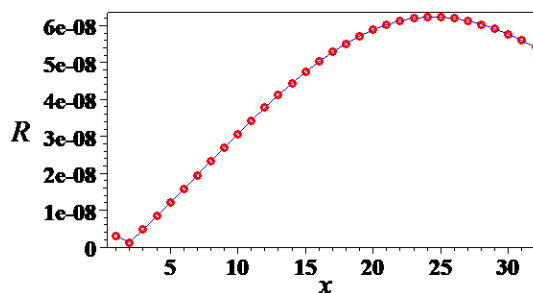

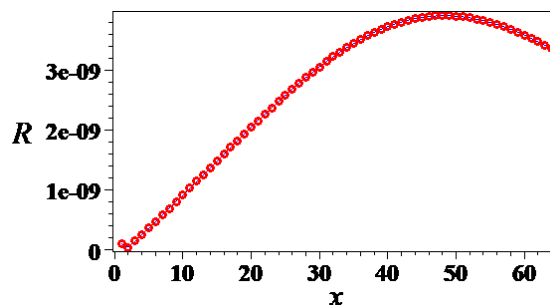Fig.11: The error of the solution of Problem 1 with cubic polynomial splines when $n = 32$

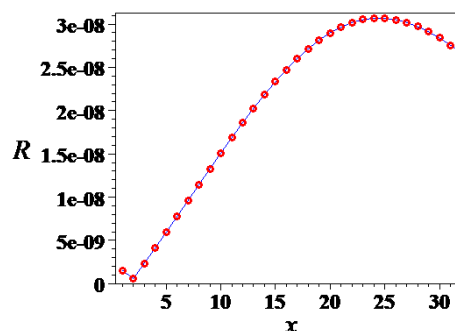Fig.12: The error of the solution of Problem 1 with the cubic polynomial splines when n = 64

Fig.13: The error of the solution of problem 1 with the non-polynomial splines when n = 32

Figs. 13 and 14 show the errors of the solution of Problem 1 with non-polynomial splines when $n = 32, 64$ (Digits=25).
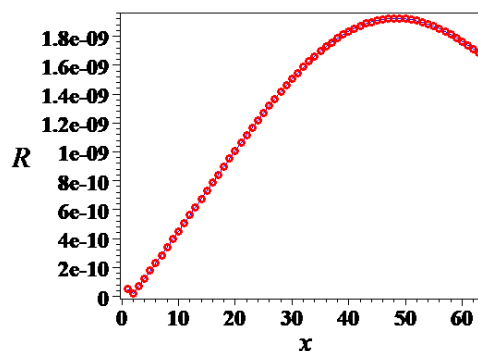

Fig.14: The error of the solution of Problem 1 with the non-polynomial splines when n = 64

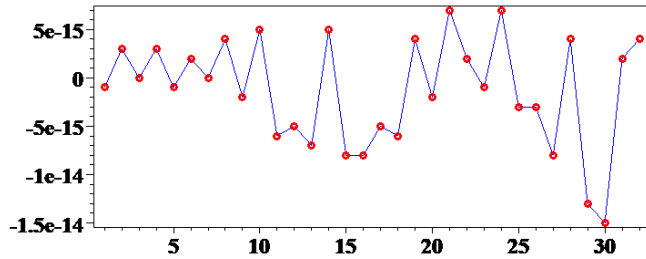Note that the trapezoidal method for this problem will give an exact result (see Fig. 15).

Fig.15: The error of the solution of Problem 1 with the non-polynomial splines when n = 32

Problem 2. We take the equation

$$u(x) = g(x) - \int_0^x (x-t)\cos(x-t)u(t)dt,$$
$$x \in [0,1].$$

The exact solution of the equation is $u(x) = 2\cos(\sqrt{3}x + 1)/3$. The function $g(x)$ is as follows:

$$g(x) = \frac{2}{3}\cos(\sqrt{3}x + 1) - \frac{\sqrt{3}}{3}x\cos(x)\sin(1)$$

$$+ \frac{2}{3}\cos(x)\cos(1) - \frac{1}{3}x\sin(x)\cos(1)$$

$$- \frac{\sqrt{3}}{3}\sin(x)\sin(1) - \frac{2}{3}\cos(\sqrt{3}\,x)\cos(1)$$

$$+ \frac{2}{3}\sin(\sqrt{3}\,x)\sin(1).$$

Fig. 16 shows the errors of the solution of Problem 2 with the cubic polynomial splines when $n = 32$.
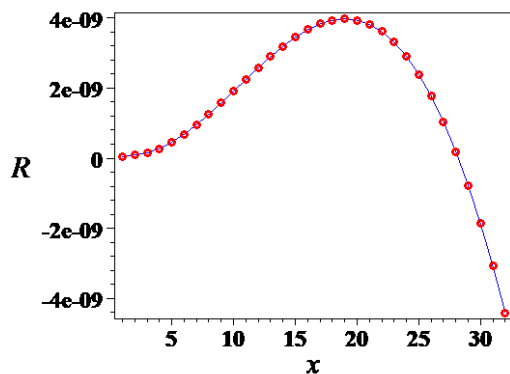

Fig.16: The error of the solution of Problem 2 with the polynomial splines when n = 32

3.2 Simpson Quadrature Rule Application

Now consider the results of applying the Simpson quadrature rule to the solution of the Volterra integral equation. We will construct the solution on a uniform grid of nodes. We denote

$$S(x,j) = \frac{x_{j+1} - x_{j-1}}{6}(K(x, x_{j-1})u_{j-1}$$

$$+ 4K(x, x_j)u_j + K(x, x_{j+1})u_{j+1}).$$

In the beginning, as always, we have

$$u(x_0) = f(x_0).$$

Further we will use a cubic interpolation spline at the nodes x_0, x_2, x_3, x_4 in the equation

$$u(x_1) + \int_0^{x_1} K(x_1, s)u(s)ds = f(x_1).$$

We use the cubic interpolation of the form:

$$U_1^R(x) = u(y_j)\omega_j^R(x) + u(y_{j+1})\omega_{j+1}^R(x)$$

$$+ u(y_{j+2})\omega_{j+2}^R(x) + u(y_{j+3})\omega_{j+3}^R(x),$$

$$\omega_j^R(x) = \frac{(x - y_{j+1})(x - y_{j+2})(x - y_{j+3})}{(y_j - y_{j+1})(y_j - y_{j+2})(y_j - y_{j+3})},$$
$$x \in [y_j, y_{j+1}],$$

$$\omega_{j+1}^R(x) = \frac{(x - y_j)(x - y_{j+2})(x - y_{j+3})}{(y_{j+1} - y_j)(y_{j+1} - y_{j+2})(y_{j+1} - y_{j+3})},$$
$$x \in [y_j, y_{j+1}],$$

$$\omega_{j+2}^R(x) = \frac{(x - y_j)(x - y_{j+1})(x - y_{j+3})}{(y_{j+2} - y_j)(y_{j+2} - y_{j+1})(y_{j+2} - y_{j+3})},$$
$$x \in [y_j, y_{j+1}],$$

$$\omega_{j+3}^R(x)$$
$$= \frac{(x - y_j)(x - y_{j+1})(x - y_{j+2})}{(y_{j+3} - y_j)(y_{j+3} - y_{j+1})(y_{j+3} - y_{j+2})}.$$
$$x \in [y_j, y_{j+1}].$$

In this case, we take $y_j = x_0, y_{j+1} = x_2, y_{j+2} = x_3, y_{j+3} = x_4$. Thus, we will apply the relation

$$V(x) = u(y_0)\omega_0^R(x) + u(y_2)\omega_2^R(x)$$
$$+ u(y_3)\omega_3^R(x) + u(y_4)\omega_4^R(x)$$

on the interval $[x_0, x_2]$. We have now

$$V(x_1) = u(y_0)\omega_0^R(x_1) + u(y_2)\omega_2^R(x_1)$$
$$+ u(y_3)\omega_3^R(x_1) + u(y_4)\omega_4^R(x_1).$$

In the following equations, we apply Simpson's rule:

$$u(x_2) + \int_0^{x_2} K(x_1, s)u(s)ds = f(x_2),$$

$$u(x_3) + \int_0^{x_1} K(x_3, s)u(s)ds$$
$$+ \int_{x_1}^{x_3} K(x_3, s)u(s)ds = f(x_3),$$

$$u(x_4) + \int_0^{x_2} K(x_4, s)u(s)ds$$
$$+ \int_{x_2}^{x_4} K(x_4, s)u(s)ds = f(x_4).$$

Therefore to obtain a solution at the points $x_j, j = 1, 2, 3, 4$, we will have to solve the system of equations:

$$u_0 = f(x_0),$$

$$u_1 + V(x_1) = f(x_1),$$

$$u_2 + S(x_2, 1) = f(x_2),$$

$$u_3 + V(x_3) + S(x_3, 2) = f(x_3),$$

$$u_4 + S(x_4, 1) + S(x_4, 3) = f(x_4).$$

Further, the next value of u_j is obtained by solving equations for odd j of the form

$$u_j + V(x_j) + \sum_{k=1}^{[j/2]} S(x_j, 2k) = f(x_j).$$

Here $[m/2]$ computes the integer quotient of m divided by 2.

And for even j we solve equations of the form

$$u_j + \sum_{k=1}^{j/2} S(x_j, 2k - 1) = f(x_j).$$

Figure 17 shows the error in solving the Volterra equation of Problem 1 by the proposed method. Thus, the application of the Simson formula gives a very good result if, in addition, on one grid interval, splines of the fourth order of approximation are used.
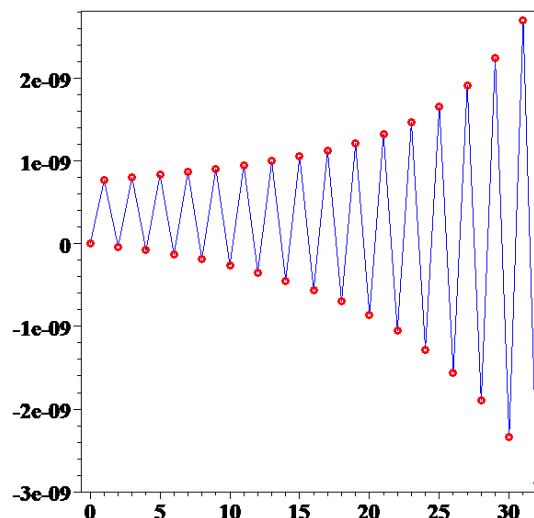

Fig.17: The graph of the error in solving the Volterra equation of Problem 1 when the Simpson method was used (32 nodes)

A different approach to the construction of calculation formulas can be also applied. We can use the Newton-Cotes formula:

$$W_3(x) = \frac{(x_3 - x_0)}{8}(K(x, x_0)u_0 + 3K(x, x_1)u_1 + 3K(x, x_2)u_2 + K(x, x_3)u_3)$$

and the spline interpolation:

$$V_1(x) = \int_{x_0}^{x_1} K(x, s)\big(u_0 w_0(s) + u_1 w_1(s) + u_2 w_2(s) + u_3 w_3(s)\big)ds.$$

The traditional approach is to apply the Simpson formula. We will also use it:

$$\int_{x_{j-1}}^{x_{j+1}} K(x, s)u(s)ds \approx S_j(x),$$

where

$$S_j(x) = \frac{x_{j+1} - x_{j-1}}{6}\big(K(x, x_{j-1})u_{j-1} + 4K(x, x_j)u_j + K(x, x_{j+1})u_{j+1}\big).$$

First, we need to define the values of the u_1, u_2, u_3, u_4. Denote

$$W_3(x) = \frac{x_3 - x_0}{8}(K(x, x_0)u_0 + 3K(x, x_1)u_1 + 3K(x, x_2)u_2 + K(x, x_3)u_3)$$

Using

$$u_0 = f(x_0),$$

and solving the system of equations

$$u_1 + V_1(x_1) = f(x_1),$$

$$u_2 + S_1(x_2) = f(x_2),$$

$$u_3 + W_3(x_3) = f(x_3),$$

$$u_4 + S_1(x_4) + S_3(x_4) = f(x_4),$$

we obtain u_1, u_2, u_3, u_4. Now, applying the Simpson rule, we are able to successively find the values of $u_j, j = 5, \ldots, n$. Further, the next value of u_j is obtained by solving equations for odd j of the form

$$u_j + W(x_j) + \sum_{k=1}^{[j/2]} S(x_j, 2k) = f(x_j).$$

Here $[m/2]$ computes the integer quotient of m divided by 2. And for even j we solve equations of the form:

$$u_j + \sum_{k=1}^{j/2} S(x_j, 2k - 1) = f(x_j).$$

In Fig.18 the errors of the solution are shown on the interval $[0,1]$ when the Simpson method and the Newton-Cotes were used (32 nodes). In Fig.19 the errors of the solution are shown on the interval $[x_5,1]$ when 32 nodes were used.
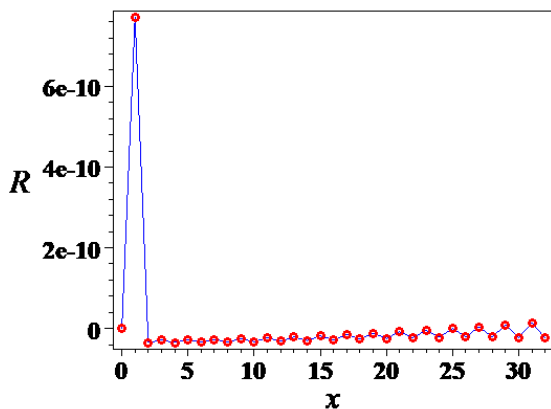

Fig.18: The graph of the error in solving the Volterra equation of Problem 1 when the Simpson method and the Newton-Cotes were used (32 nodes)
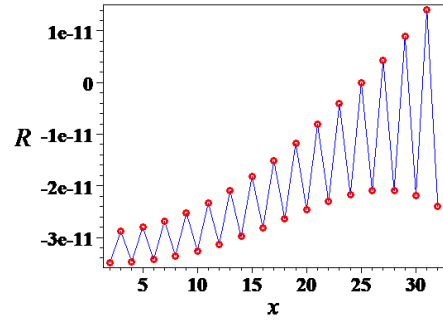

Fig.19: The graph of the error in solving the Volterra equation of Problem 1 when the Simpson and the Newton-Cotes were used (32 nodes)

Figures 20-21 show the results of solving Problem 1 when the Simpson method and the Newton-Cotes were used. In Fig.20 the errors of the solution are shown on the interval $[0,1]$ when 64 nodes were used. In Fig.20 the errors of the solution are shown on the interval $[u_5,1]$ when 64 nodes were used.
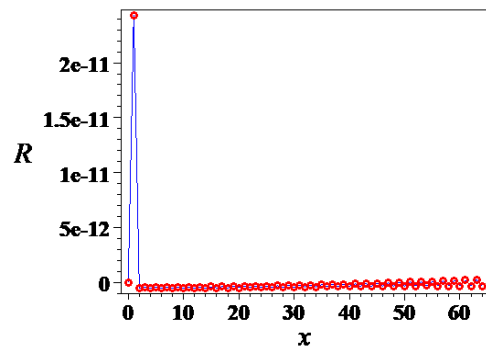

Fig.20: The graph of the error in solving the Volterra equation of Problem 1 when the Simpson method and the Newton-Cotes were used (64 nodes)
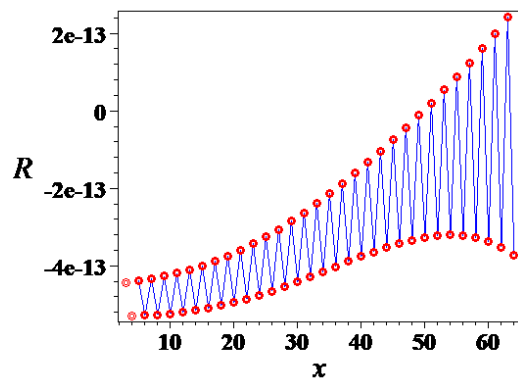

Fig.21: The graph of the error in solving the Volterra equation of problem 1 when the Simpson method and the Newton-Cotes were used (64 nodes)

Using the left and right integro-differential polynomial splines we can construct the numerical

scheme to solve the Volterra equation of the second kind.

Denote

$$W(x,j) = \sum_{k=0}^{2} u_{j+k} \int_{x_j}^{x_{j+1}} K(x,s) w_{j+k}^{R}(s) ds +$$
$$\int_{x_{j+1}}^{x_{j+2}} u(s) ds \int_{x_j}^{x_{j+1}} K(x,s) w_j^{R<1,2>}(s) ds,$$

$$W_1(x,j)$$
$$= \sum_{k=-1}^{1} u_{j+k} \int_{x_j}^{x_{j+1}} K(x,s) w_{j+k}^{L}(s) ds$$
$$+ \int_{x_j}^{x_{j+1}} u(s) ds \int_{x_j}^{x_{j+1}} K(x,s) w_j^{L<0,1>}(s) ds.$$

As usual we have $u(x_0) = f(x_0)$. Next, we have the system of equations

$$u(x_1) + \int_0^{x_1} K(x_1,s) u(s) ds = f(x_1),$$

$$u(x_2) + \int_0^{x_1} K(x_2,s) u(s) ds$$
$$+ \int_{x_1}^{x_2} K(x_2,s) u(s) ds = f(x_2).$$

Using the right and the left polynomial splines, we obtain

$$\int_0^{x_1} K(x_1,s) u(s) ds = W(x_1, 0).$$

$$\int_{x_1}^{x_2} K(x_2,s) u(s) ds = W_1(x_2, 1).$$

Thus, we have to solve the system of equations in order to obtain $u(x_1)$, $u(x_2)$:

$$u(x_1) + W(x_1, 0) = f(x_1),$$

$$u(x_2) + W(x_2, 0) + W_1(x_2, 1) = f(x_2).$$

Next we can obtain $u(x_j)$, $j = 3, ..., n - 1$, solving the equations

$$u(x_j) + W(x_2, 0) + \sum_{k=1}^{j-1} W_1(x_j, k) = f(x_j),$$

$$j = 3, ..., n - 1.$$

Figs. 22 and 23 show the graph of the error of Problem 1 when the integro-differential polynomial splines were used.
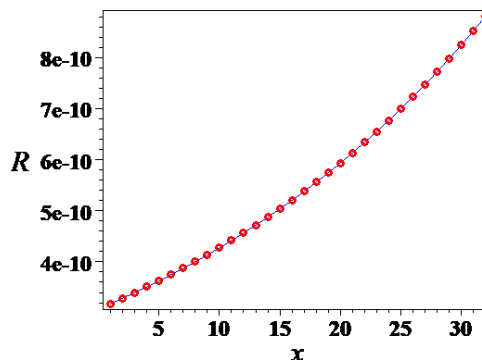

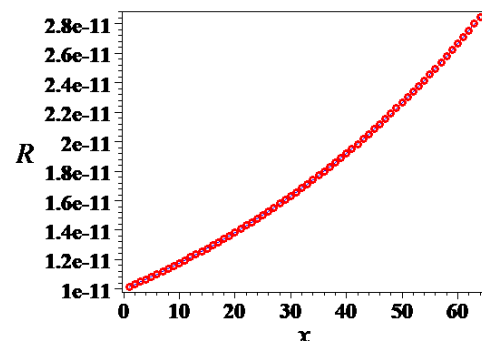Fig.22: The error of the solution of Problem 1 (Digits = 25, 32 nodes)

Fig.23: The error of the solution of Problem 1 (Digits = 25, 32 nodes)

Comparing the results of numerical experiments for Problem 1 shown in Figs. 11-14 and Figs. 22-23, we see that the integro-differential splines give a smaller error.

4 Conclusion

This paper considered the use of polynomial and non-polynomial splines of the fourth order of approximation. The results of the numerical experiments have shown the correspondence of theoretical estimates to practical experiments and numerical stability of the proposed methods with the Polynomial-trigonometrical splines gave less error, but more signs in the mantissa are required for the calculations.

In some cases, the values of the integrals can be known. In this case, the integro-differential splines can be applied. The integro-differential splines allow us to get a smaller error both when approximating and when solving integral equations.

I.G. Burova, A.G. Doronina, D.E. Zhilin

The application of the integro-differential splines will continue to be investigated in details in the next papers.

References:

[1] Q.Huang, M.Wang, Superconvergence of interpolated collocation solutions for weakly singular Volterra integral equations of the second kind, *Computational and Applied Mathematics*, Vol. 40, No 3, paper 71, 2021.

[2] A.F.Al-Shimmary, A.K.Hussain, S.K.Radhi, Numerical Solution of Volterra Integro-Differential Equation Using 6thOrder Runge-Kutta Method, *Journal of Physics: Conference Series*, Vol. 1818, No 1, paper 012183, 2021.

[3] M.Mohammad, C.Cattani, A collocation method via the quasi-affine biorthogonal systems for solving weakly singular type of Volterra-Fredholm integral equations, *Alexandria Engineering Journal*, Vol. 59, No 4, 2020, pp. 2181-2191.

[4] J.Ma, H.Liu, Fractional collocation boundary value methods for the second kind Volterra equations with weakly singular kernels, *Numerical Algorithms,* Vol. 84, No 2, 2020, pp. 743-760.

[5] S.Soradi-Zeid, Efficient radial basis functions approaches for solving a class of fractional optimal control problems, *Computational and Applied Mathematics,* Vol. 39, No 1, paper 2 , 2020.

[6] M. Asif, I. Khan, N. Haider, Q. Al-Mdallal, Legendre multi-wavelets collocation method for numerical solution of linear and nonlinear integral equations. *Alexandria Engineering Journal*, Vol. 59, 2020, pp. 5099-5109.

[7] Derakhshan, M., Zarebnia, M. On the numerical treatment and analysis of two-dimensional Fredholm integral equations using quasi-interpolant, *Computational and Applied Mathematics,* Vol. 39, No 2, paper 106, 2020.

[8] I.G.Burova, On left integro-differential splines and Cauchy problem, *International Journal of Mathematical Models and Methods in Applied Sciences,* Vol. 9, 2015, pp. 683-690.

[9] I.G.Burova, Application local polynomial and non-polynomial splines of the third order of approximation for the construction of the numerical solution of the volterra integral equation of the second kind, *WSEAS Transactions on Mathematics,* Vol. 20, 2021, pp. 9-23.