Smooth Polynomial and Non-polynomial Splines of the Second Order of Approximation

I. G. BUROVA, ZHEKAI LAN
The Department of Computational Mathematics,
St. Petersburg State University,
7-9 Universitetskaya Embankment, St.Petersburg,
RUSSIA

Abstract: - Continuous local polynomial splines are widely used in approximations, as well as in solving differential and integral equations.. In this paper, we consider the construction of continuously differentiable splines of the second-order of approximation. We construct continuously differentiable polynomial and non-polynomial splines by smoothing continuous local approximations. The construction of smoothed non-polynomial approximations is discussed using the example of constructing smooth trigonometric splines. Theoretical results are given. Results with numerical experiments are presented. Among the numerical experiments, the approximation on a non-uniform grid of nodes is considered, as well as the influence of introducing additional errors into the values of the function given at the grid nodes.

Key-Words: - polynomial splines, non-polynomial splines, smooth approximations, continuous local approximations, interpolation, trigonometric splines

Received: April 14, 2026. Revised: May 14, 2026. Accepted: June 2, 2026. Published: July 27, 2026.

1 Introduction

Currently, the application areas of polynomial and non-polynomial splines are expanding, encompassing ever-new application areas. Traditionally, splines are used to solve problems arising from mathematical modeling problems, including solving integral equations and boundary value problems. Below, we mention several recent publications that have used splines to numerically solve various emerging problems.

In paper [1], the nonlinear splines were used to forecast the construction costs index (CCI), which is an important macroeconomic indicator.

In paper [2], quadratic polynomial splines are used for free-form constructions on a planar Euclidean domain.

In paper [3], a new numerical method, based on a non-polynomial cubic spline algorithm, for solving a class of stochastic partial differential equations is presented.

Paper [4], introduces a novel and developed numerical technique that solves nonlinear time fractional differential equations. It combines finite difference and non-polynomial spline methods while using comfortable descriptions of fractional derivatives.

In paper [5], the non-polynomial B-splines are explicitly expressed in terms of non-polynomial divided differences of truncated power function are expressed.

In paper [6], the quintic spline collocation method, based on non-polynomial quintic spline functions, was used to construct the numerical solution of fourth-order fractional boundary value problems.

In paper [7], the approximate solutions to the mixed boundary value problem were developed by using non-polynomial spline functions.

In paper [8], a new general form for the non-polynomial cubic spline function is introduced and used to approximate the solution of linear, nonlinear, and fuzzy Fredholm integral equations of the second kind.

Paper [9] introduces the fractional non-polynomial spline method as a novel scheme for solving the time-fractional Korteweg-de Vries (KdV) equation.

In paper [10], a fourth-order exponential spline-based scheme is developed to solve the coupled system of the benchmark problem.

In paper [11], the exponential cubic B-spline differential quadrature method was used to find the numerical solutions of the nonlinear Schrodinger (NLS) equation.

In paper [12], two different numerical techniques with the trigonometric B-spline and the exponential B-spline are employed to solve the Volterra partial integro-differential equation (PIDE) with a weak singular kernel.

Paper [13], uses the quartic trigonometric B-spline for solving the Lane–Emden–Fowler equation.

Paper [14], the quintic UAT tension B-spline collocation method was used for solving the one- and multi-dimensional cubic Ginzburg-Landau (CGL) equation.

In paper [15], Troesch's problem was solved using the collocation method based on the cubic trigonometric B-Spline technique.

In paper [16], a new differential quadrature method (DQM) based on the Unified and Extended Trigonometric B-spline (UETB-spline) functions for the numerical approximation of 3D wave equations was proposed.

In paper [17], bivariate spline quasi-interpolants were used for solving the Fredholm integral equation of the second kind.

Thus, the task of processing experimental results arises when solving various problems. Many numerical methods for solving problems in mathematical physics involve calculating the framework of an approximate solution. Next, the obtained function values can be connected with a line or surface. This line (or a surface) can have certain specified properties. Some of these properties can include interpolation conditions, continuous properties, and/or continuous differentiability properties (in the case of surface construction, this means differentiability properties along a given direction).

2 Problem Formulation and Problem Solution

Suppose $\{x_j\}$ is a set of ordered nodes on an interval $[a,b]$, such that $a = x_0 < \cdots < x_j < \cdots < x_n = b$. Suppose we are given the values f_j of a function $f(x)$ at nodes x_j, $j = 0,1,\dots,n$. In the simplest problem, the set of nodes $\{x_j\}$ is given with step $h = (b-a)/n$. Suppose the functions $\varphi_0(x)$, $\varphi_1(x)$ form a Chebyshev system on the interval $[x_j, x_{j+1}]$.

Basis splines on the interval $[x_j, x_{j+1}]$ are found from the condition

$$U_k^h(x) = u(x), \qquad u(x) = \varphi_0(x),\ \varphi_1(x) \quad (1)$$

Thus, we arrive at a system of approximation identities

$$\varphi_0(x_j)\omega_j(x) + \varphi_0(x_{j+1})\omega_{j+1}(x) = \varphi_0(x),$$
$$\varphi_1(x_j)\omega_j(x) + \varphi_1(x_{j+1})\omega_{j+1}(x) = \varphi_1(x), \quad (2)$$
$$x \in [x_j, x_{j+1}].$$

Here $\omega_j(x)$, $\omega_{j+1}(x)$ are the basis splines to be obtained, $supp\ \omega_j = [x_{j-1}, x_{j+1}]$.

$$\omega_j(x) = \frac{\varphi_0(x)\varphi_1(x_{j+1}) - \varphi_1(x)\varphi_0(x_{j+1})}{\varphi_1(x_{j+1})\varphi_0(x_j) - \varphi_1(x_j)\varphi_0(x_{j+1})} \quad (3)$$

$$\omega_{j+1}(x) = \frac{\varphi_0(x)\varphi_1(x_j) - \varphi_1(x)\varphi_0(x_j)}{\varphi_1(x_j)\varphi_0(x_{j+1}) - \varphi_1(x_{j+1})\varphi_0(x_j)} \quad (4)$$

Next we regard the construction smooth splines using the local continuous polynomial and non-polynomial splines of the second order of approximation.

2.1 Polynomial Case

To construct a polynomial approximation of the second order of approximation we construct a basis spline $w_j(x)$ such that supp $w_j(x) = [x_{j-1}, x_{j+1}]$.

The formula of the polynomial basis spline is the following:

$$\omega_j^p(x) = \begin{cases} \dfrac{x - x_{j+1}}{x_j - x_{j+1}}, & x \in [x_j, x_{j+1}], \\ \dfrac{x - x_{j-1}}{x_j - x_{j-1}}, & x \in [x_{j-1}, x_j], \end{cases} \quad (5)$$
$$\omega_j^p(x) = 0,\ x \notin [x_{j-1}, x_{j+1}].$$

We denote $f_j = f(x_j)$. On each grid interval $[x_j, x_{j+1}]$ we construct a linear spline approximation as follows:

$$S_j(x) = f_j\, w_j(x) + f_{j+1}\, w_{j+1}(x), \quad (6)$$

where $w_j(x), w_{j+1}(x)$ are the basis splines. These splines are constructed separately for each grid interval and can be obtained by solving a system of approximation relations.

In the polynomial case, the basis splines are as follows:

$$w_j(x) = \frac{x - x_{j+1}}{x_j - x_{j+1}}, \quad (7)$$

$$w_{j+1}(x) = \frac{x - x_j}{x_{j+1} - x_j}. \quad (8)$$

The plots of basis splines are given in Figure 1.

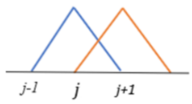

Fig. 1: The plots of basis splines
Source: created by the authors

The error of approximation with the splines is as follows, [18]:

$$\parallel f - S_j \parallel_{C[x_j,x_{j+1}]} \leq \frac{1}{8} h^2 \parallel f'' \parallel_{C[x_j,x_{j+1}]} . \quad (9)$$

Using the approximation on interval $[x_j, x_{j+1}]$ we construct the piece-wise function $S(x)$ on interval $[a,b]$ as follows:

$$S(x) = \begin{cases} S_0(x), x \in [x_0, x_1], \\ \quad \dots \\ S_{n-1}(x), x \in [x_{n-1}, x_n]. \end{cases} \quad (10)$$

According to our construction, approximation (10) is a continuous function on the interval $[a,b]$. Our aim is to construct a smooth approximation on $[a,b]$. We will show that after adding one component to approximation (6), we could construct a smooth approximation. We construct the smooth approximation on the interval $[x_j, x_{j+1}]$ in the form:

$$Z_j(x) = f_j\, w_j(x) + f_{j+1}\, w_{j+1}(x) + a_j w_j(x) w_{j+1}(x), \quad x \in [x_j, x_{j+1}]. \quad (11)$$

Here a_j is a parameter to be obtained such that the first derivative of $Z(x)$:

$$Z'(x) = \begin{cases} Z_0^{\prime(x)}, x \in [x_0, x_1], \\ \quad \dots \\ Z'_{n-1}(x), x \in [x_{n-1}, x_n], \end{cases} \quad (12)$$

is a continuous function. Here we denote $Z(x)$ a piecewise function on the interval $[a,b]$.

Thus, to receive a formula with a continuous first derivative, we calculate the first derivative of formula (11):

$$Z_j'(x) = f_j\, w_j'(x) + f_{j+1}\, w_{j+1}'(x) + a_j \left(w_j(x) w_{j+1}'(x) + w_j'(x) w_{j+1}(x) \right). \quad (13)$$

In the interval $[x_{j-1}, x_j]$ we have a relation with an unknown parameter a_{j-1}:

$$Z_{j-1} = f_{j-1}\, w_{j-1}(x) + f_j\, w_j(x) + a_{j-1} w_{j-1}(x) w_j(x), \quad x \in [x_{j-1}, x_j]. \quad (14)$$

So, we have the relation for the first derivative:

$$Z_{j-1}'(x) = f_{j-1}\, w_{j-1}'(x) + f_j\, w_j'(x) + a_{j-1} \left(w_{j-1}(x) w_j'(x) + w_{j-1}'(x) w_j(x) \right), \quad (15)$$

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

Thus, to fulfill the relation

$$Z_{j-1}'(x_j) = Z_j'(x_j) \quad (16)$$

at each node $x_j, j = 1, \dots n-1$, the following expression should be fulfilled in the polynomial case:

$$a_j = a_{j-1}(x_{j+1} - x_j)/(x_{j-1} - x_j) + (f_j - f_{j-1})(x_{j+1} - x_j)/(x_j - x_{j-1}) + f_j - f_{j+1}. \quad (17)$$

Now, we can put $a_0 = 0$ and construct the smooth approximation for the function $f(x)$:

$$Z_j(x) = f_j\, w_j(x) + f_{j+1}\, w_{j+1}(x) + a_j w_j(x) w_{j+1}(x), \quad x \in [x_j, x_{j+1}]. \quad (18)$$

We assume a uniform grid is constructed on $[a,b]$. Thus, on the interval $[x_0, x_1]$ we have:

$$Z_0(x) = f_0\, w_0(x) + f_1\, w_1(x). \quad (19)$$

On the interval $[x_1, x_2]$, we have the approximation in the form:

$$Z_1(x) = f_1\, w_1(x) + f_2\, w_2(x) + A_{12},$$
$$A_{12} = \left((f_1 - f_0) + (f_1 - f_2) \right) w_1(x) w_2(x). \quad (20)$$

Expanding the functions f_0, f_2 using Taylor's formula in the neighborhood of the point x_1, we have:

$$f_1 - f_0 = -h\, f'(x_1) + O(h^2),$$
$$f_1 - f_2 = h\, f'(x_1) + O(h^2). \quad (21)$$

Thus, we obtain the relation:

$$|A_{12}| = |(f_1 - f_0) + (f_1 - f_2)||w_1(x) w_2(x)| \leq K h^2 \max_{x \in [x_0, x_2]} |f''(x)|. \quad (22)$$

Next, on the interval $[x_2, x_3]$, we have:

$$Z_2(x) = f_2\, w_2(x) + f_3\, w_3(x) + A_{23},$$
$$A_{23} = \left((f_0 - f_1) + 2(f_2 - f_1) + (f_2 - f_3) \right) w_2(x) w_3(x). \quad (23)$$

Similarly, expanding the functions f_0, f_2, using Taylor's formula in the local area of the point x_1, and expanding the functions f_1, f_3, in the local area of the point x_2, we have:

$$|(f_0 - f_1) + (f_2 - f_1)| \leq K_1 h^2 \max_{[x_0, x_2]} |f''|, \quad (24)$$

$$|(f_2 - f_3) + (f_2 - f_1)| \leq K_2 h^2 \max_{[x_1, x_3]} |f''|, \quad (25)$$

$K_1 = \text{const}, K_2 = \text{const}$, and we obtain that

$$|A_{23}| \leq K_3 h^2 \max_{[x_0, x_3]} |f''|, \; K_3 = \text{const}. \quad (26)$$

It is not difficult to see that the following theorem is valid.

Theorem 1. *To estimate the approximation of a function $f \in C^2[x_j, x_{j+1}]$, the following relation holds:*

$$\| f - S_j \|_{C[x_j,x_{j+1}]} \leq \frac{1}{8} h^2 \; \| f'' \|_{C[x_j,x_{j+1}]} + h^2 \, K_j \, \| f'' \|_{C[x_0,x_{j+1}]}, \text{ where } K_j = \text{const.} \quad (27)$$

Proof. Expanding f_{j+1}, f_{j-1}, using the Taylor formula in the local areas round the points x_{j+1} and x_{j-1}, we get the relations: $|(f_j - f_{j-1}) + (f_j - f_{j+1})| \leq k_j h^2 \max_{[x_{j-1},x_{j+1}]} |f''(x)|$, $k_j = \text{const}$.

Note that on the interval $[x_j, x_{j+1}]$ we can use x, $x \in [x_j, x_{j+1}]$, in the form $x = x_j + th, t \in [0,1]$, so we have the relations: $w_j(x_j + th) = 1 - t$, $w_{j+1}(x_j + th) = t$, and therefore, on the interval $[x_j, x_{j+1}]$ the next estimate holds: $|w_j(x_j + th) w_{j+1}(x_j + th)| \leq \frac{1}{4}$.

Using the relation $|a_j \, w_j(x) w_{j+1}(x)| \leq K_j h^2$ when $x \in [x_j, x_{j+1}]$ and applying estimate (9), we obtain the desired statement.

The theorem is proved.

Corollary 1. When using a non-uniform grid of nodes, the error estimate will include a term dependent on the first derivative of the function and with an error order of $O(h)$, where $h = \max_i h_i$.

Corollary 2. When introducing additional errors into the function values at grid nodes in the case of smoothing splines, the error accumulates linearly as the grid interval numbers increase.

Consider an example of smoothing a continuously differentiable function $f_1(x)$ consisting of three parts as follows:

$$f_1(x) = \begin{cases} -\left(x - \frac{1}{2}\right)^2, & x \in \left[-1, -\frac{1}{2}\right], \\ 2x, & x \in \left[-\frac{1}{2}, \frac{1}{2}\right], \\ \left(x + \frac{1}{2}\right)^2, & x \in \left[\frac{1}{2}, 1\right]. \end{cases} \quad (28)$$

The plot of the function $f_1(x)$ is shown in Figure 2. The plot of the first derivative of the function $f(x)$ is shown in Figure 3.

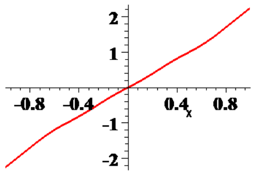

Fig. 2: The plot of the function $f_1(x)$
Source: created by the authors

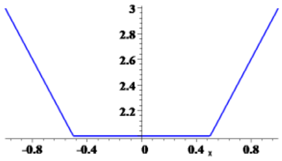

Fig. 3: The plot of the first derivative of the function $f_1(x)$
Source: created by the authors

The plot of the error of the smoothed polynomial approximation of the function $f_1(x)$ is given in Figure 4.

The plot of the error of the continuous polynomial approximation of the function $f_1(x)$ is given in Figure 5.

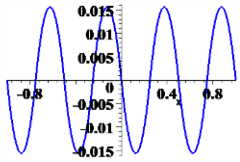

Fig. 4: The plot of the error of the smoothed polynomial approximation of the function $f_1(x)$
Source: created by the authors

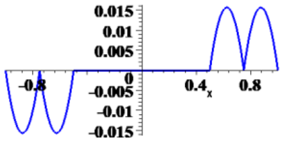

Fig. 5: The plot of the error of the continuous polynomial approximation of the function $f_1(x), n = 7$.
Source: created by the authors

Now consider an example of smoothing a continuous function $f_2(x)$:

$$f_2(x) = \begin{cases} -1, & x \in \left[-1, -\frac{1}{2}\right], \\ 2x, & x \in \left[-\frac{1}{2}, \frac{1}{2}\right], \\ 1, & x \in \left[\frac{1}{2}, 1\right]. \end{cases} \quad (29)$$

Using the values of the function $f_2(x)$ at 16 equally spaced nodes after constructing a smoothing approximation, we obtain the function shown in Figure 6.

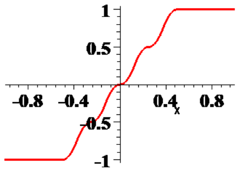

Fig. 6: The plot of the smoothed polynomial approximation of the function $f_2(x)$
Source: created by the authors

The plot of the error of the smoothed polynomial approximation of the function $f_2(x)$ is given in Figure 7.

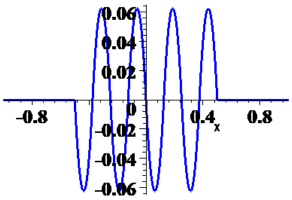

Fig. 7: The plot of the error of the smoothed polynomial approximation of the function $f_2(x)$
Source: created by the authors

The plot of the error of the continuous polynomial approximation of the function $f_2(x)$ is given in Figure 8.

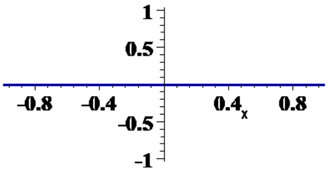

Fig. 8: The plot of the error of the continuous polynomial approximation of the function $f_2(x)$, $n = 16$.
Source: created by the authors

Now we examine the effect of extra experimental errors on the smoothing spline result. Suppose the errors arise from data rounding in computer memory, or the errors arise due to the experiments. For definiteness, suppose the quantity of nodes is given with $n = 15$ and the test function is the next: $f_3(x) = \frac{1}{1+25\,x^2}$. This function is known as the Runge function. The Runge function is often used as a test function. We add extra errors eps with values not more than $eps = 0.000017$ into the Runge function's values at odd grid nodes. Figure 9 shows a plot of the approximation error when using smoothed splines ($n = 60$). Figure 10 shows a plot of the approximation error when using linear polynomial splines ($n = 60$). The plots on Figure 9 and Figure 10 show that the smoothed approximation error increases linearly with increasing grid interval number if an additional error is introduced into the values of the function at grid nodes.

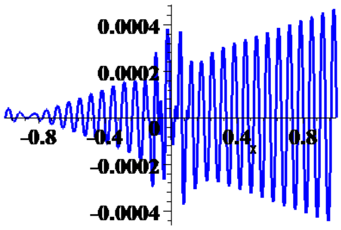

Fig. 9: The plot of the approximation error when using smoothed splines ($n = 60$, with extra errors)
Source: created by the authors

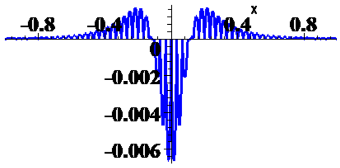

Fig. 10: The plot of the approximation error when using linear polynomial splines ($n = 60$, with extra errors)
Source: created by the authors

For comparison, in Figure 11, we present the plot of the approximation error without adding extra errors into the Runge function's values.

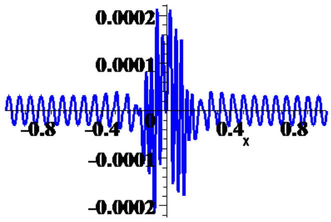

Fig. 11: The plot of the approximation error when using smoothed splines ($n = 60$, without extra errors)
Source: created by the authors

The plot of the error of approximation of the first derivative of the Runge function when using smooth polynomial splines is given in Figure 12.

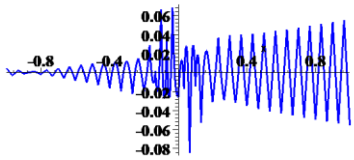

Fig. 12: The plot of the error of approximation of the first derivative of the Runge function when using smooth polynomial splines ($n = 60$, with extra errors)
Source: created by the authors

It should be noted that smoothing splines are useful when solving integral equations.

By replacing the unknown function in the integral part using spline approximations, we reduce the solution of the integral equation to the solution of a system of algebraic equations. The solution to this system of equations (linear or nonlinear) is the framework of an approximate solution. Smoothing splines can then be used to construct a continuously differentiable solution of the integral equation.

The smoothing splines are useful when solving boundary value problems with grid methods or variational methods. When we have obtained the framework of an approximate solution, then the smoothing splines can be used to construct a continuously differentiable solution of the boundary value problem.

The method can be used for constructing a smooth approximation on both uniform and non-uniform grids. For the examples below in the next section, a non-uniform grid of nodes was used, constructed using the roots of the Chebyshev polynomial of the first kind on the interval $[-1,1]$. The Chebyshev polynomial of the first kind has the form $T_m(x) = \cos(m \arccos(x))$.

The roots of $T_m(x)$ when $m = 16$ are as follows:
$x[0] \approx -0.9952$, $x[1] \approx -0.9569$, $x[2] \approx -0.8819$, $x[3] \approx -0.7730$, $x[4] \approx -0.6344$, $x[5] \approx -0.4714$, $x[6] \approx -0.2903$, $x[7] \approx -0.09802$, $x[8] \approx 0.09802$, $x[9] \approx 0.2903$, $x[10] \approx 0.4714$, $x[11] \approx 0.6344$, $x[12] \approx 0.7730$, $x[13] \approx 0.8819$, $x[14] \approx 0.9569$, $x[15] \approx 0.9952$.

2.2 Non-Polynomial Case

For constructing smooth approximations using non-polynomial splines, we will use the construction scheme described in the previous section.

In the trigonometrical case, the basis functions are as follows:

$$w_j(x) = \frac{\sin(x - x_{j+1})}{\sin(x_j - x_{j+1})}, x \in [x_j, x_{j+1}], \quad (30)$$

$$w_{j+1}(x) = \frac{\sin(x - x_j)}{\sin(x_{j+1} - x_j)}, x \in [x_j, x_{j+1}]. \quad (31)$$

It should be mentioned that on an equidistant set of node with step h the basis functions are as follows: $w_j(x_j + th) = 1 - t + O(h^2), w_{j+1}(x_j + th) = t + O(h^2)$, where $t \in [0,1]$.

The continuous approximation with the trigonometric splines on the grid interval $[x_j, x_{j+1}]$ we construct in the form:

$$S_j^T(x) = f_j\, w_j(x) + f_{j+1}\, w_{j+1}(x). \quad (32)$$

The error of the continuous approximation with the splines is as follows, [18]: $\| f - S_j^T \|_{C[x_j,x_{j+1}]} \leq Kh^2 \| f'' + f \|_{C[x_j,x_{j+1}]}$.
Recall that we find the basis splines using approximation identities, applying the Chebyshev system of functions $\varphi_0(x) = \sin(x)$, $\varphi_1(x) = \cos(x)$.

The smooth approximation for the function $f(x)$ we construct in the form:

$$Z_j(x) = f_j\, w_j(x) + f_{j+1}\, w_{j+1}(x) + a_j w_j(x) w_{j+1}(x), \quad x \in [x_j, x_{j+1}]. \quad (33)$$

Thus, to fulfill the relation $Z'_{j-1}(x_j) = Z'_j(x_j)$ at each node $x_j, j = 1, 2, \dots, n-1$, the following expression should be fulfilled in the trigonometrical case:

$$a_j = a_{j-1} \frac{\sin(-x_{j+1} + x_j)}{\sin(x_j - x_{j-1})} + F, \quad (34)$$

where $F = f_j \cos(x_{j+1} - x_j) - f_{j+1} + f_{j-1} \sin(x_j - x_{j+1}) / \sin(x_j - x_{j-1}) - f_j \cos(x_{j+1} - x_j) \sin(x_j - x_{j+1}) / \sin(x_j - x_{j-1})$.

The formula above is constructed on a nonuniform set of nodes. When the nodes are equidistant ones with step h, then the relation can be written in more simple form $a_j = -a_{j-1} + 2 f_j \cos(h) - f_{j+1} - f_{j-1}$.

The approximation with the smooth trigonometric splines on the interval $[x_1, x_2]$ can be written in the form:

$$Z_1(x) = f_1\, w_1(x) + f_2\, w_2(x) + A_{12}, \quad (35)$$

where $A_{12} = (2 f_1 \cos(h) - f_0 - f_2) w_1(x) w_2(x)$

Using the relation $\cos(h) = 1 - \frac{h^2}{2} + O(h^3)$;

We get:

$$A_{12} = \big((f_1 - f_0) + (f_1 - f_2)\big) w_1(x) w_2(x) + O(h^2).$$

Thus, $|A_{12}| \leq K_1 h^2 \max\limits_{[x_0, x_2]} |f''|$.

The approximation with the smooth trigonometric splines on the interval $[x_2, x_3]$ can be written in the form:

$$Z_2(x) = f_2\, w_2(x) + f_3\, w_3(x) + A_{23}, \quad (36)$$

where $A_{23} = (-2 f_1 \cos(h) + 2 f_2 \cos(h) + f_0 + f_2 - f_1 - f_3) w_1(x) w_2(x)$.

Thus $|A_{23}| \leq K_2 h^2 \max_{[x_0,x_3]} |f''|$.
It is not difficult to see that the following theorem is valid.

Theorem 2. *To estimate the approximation of a function $f \in C^2[x_j, x_{j+1}]$, the following relation holds:*

$$\| f - S_j \|_{C[x_j,x_{j+1}]} \leq K h^2 \ \| f'' + f \|_{C[x_j,x_{j+1}]} + h^2 \ K_1 \ \| f'' \|_{C[x_0,x_{j+1}]}, \quad (37)$$

where $K, K_1 = $ const.

3 The Results of Numerical Experiments

Numerical experiments were conducted using the Maple computer algebra package. Figure 13, Figure 14, Figure 15, Figure 16, Figure 17, Figure 18, Figure 19, Figure 20, Figure 21, Figure 22, Figure 23, Figure 24, Figure 25, Figure 26, Figure 27, and Figure 28 show the results of numerical calculations. We take the Runge function $f(x) = \frac{1}{1+25x^2}$ on interval [-1,1]. The set of equidistant or non- equidistant nodes $x_j, j = 0,1,\dots,n, n = 15, were$ constructed on the interval [-1 ,1], and the values of the function $f(x)$ at these points x_j were used.

Figure 13 shows the plot of the Runge function on the interval [-1 ,1].

Figure 14 shows the plot of the error of approximation $(R(x) = f(x) - S(x))$ of the smoothed approximation for the Runge function when using nodes taken as roots of the Chebyshev polynomial ($n = 15$).

Figure 15 shows the plot of the error of the piecewise linear approximation for the Runge function when using nodes taken as roots of the Chebyshev polynomial ($n = 15$).

Figure 16 shows the plot of the approximation of the first derivative of the smoothed approximation when using nodes taken as roots of the Chebyshev polynomial ($n = 15$).

Figure 17 shows the plot of the error of approximation of the first derivative of the smoothed approximation when using nodes taken as roots of the Chebyshev polynomial ($n = 15$).

Figure 18 shows the plot of the error of approximation of the first derivative of the Runge function with piecewise linear approximation when using nodes taken as roots of the Chebyshev polynomial ($n = 15$).

Figure 19 shows the plot of the error of the smoothed approximation for the Runge function when using equidistant nodes ($n = 15$).

Figure 20 shows the plot of the error of the piecewise linear approximation for the Runge function when using equidistant nodes ($n = 15$).

Figure 21 shows the plot of the error of approximation of the first derivative of the smoothed approximation of the Runge function when using equidistant nodes ($n = 15$).

Figure 22 shows the plot of the error of approximation of the first derivative of the Runge function with the smoothed approximation when using equidistant nodes ($n = 15$).

Figure 23 shows the plot of the error of the smoothed approximation of the Runge function when using equidistant nodes ($n = 15$).

Figure 24 shows the plot of the error of the piecewise trigonometric approximation of the Runge function when using equidistant nodes ($n = 15$).

Figure 25 shows the plot of the error of the approximation of the first derivative of the Runge function with smooth splines when using equidistant nodes ($n = 15$).

Figure 26 shows the plot of the error of the approximation of the Runge function with smooth splines when using nodes taken as roots of the Chebyshev polynomial ($n = 15$).

Figure 27 shows the plot of the error of the approximation of the first derivative of the Runge function with the piecewise trigonometric approximation when using nodes taken as roots of the Chebyshev polynomial ($n = 15$).

Figure 28 shows the plot of the error of approximation of the first derivative of the Runge function with the smooth approximation when using nodes taken as roots of the Chebyshev polynomial ($n = 15$).

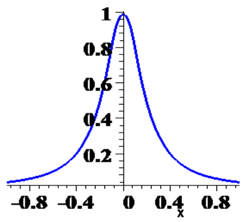

Fig. 13: The plot of the Runge function
Source: created by the authors

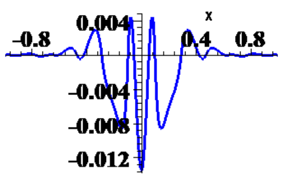

Fig. 14: The plot of the error of the smoothed polynomial approximation for the Runge function when using nodes taken as roots of the Chebyshev polynomial ($n = 15$)
Source: created by the authors

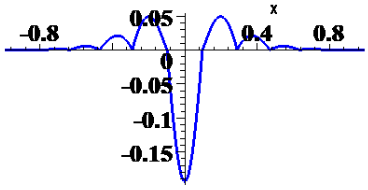

Fig. 15: The plot of the error of the piecewise linear polynomial approximation for the Runge function when using nodes taken as roots of the Chebyshev polynomial ($n = 15$)
Source: created by the authors

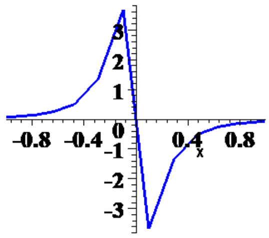

Fig. 16: The plot of the first derivative of the Runge function ($n = 15$)
Source: created by the authors

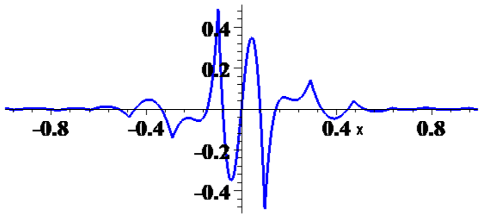

Fig. 17: The plot of the error of approximation of the first derivative of the smoothed polynomial approximation when using nodes taken as roots of the Chebyshev polynomial ($n = 15$)
Source: created by the authors

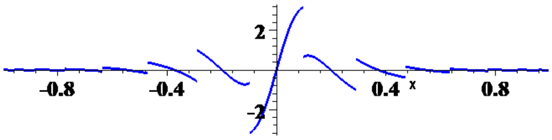

Fig. 18: The plot of the error of approximation of the first derivative of the Runge function with piecewise linear polynomial approximation when using nodes taken as roots of the Chebyshev polynomial ($n = 15$)
Source: created by the authors

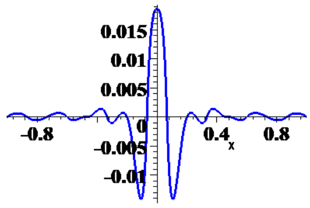

Fig. 19: The plot of the error of the smoothed polynomial approximation for the Runge function when using equidistant nodes ($n = 15$)
Source: created by the authors

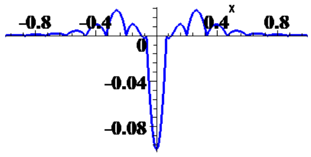

Fig. 20: The plot of the error of the piecewise linear polynomial approximation for the Runge function when using equidistant nodes ($n = 15$)
Source: created by the authors

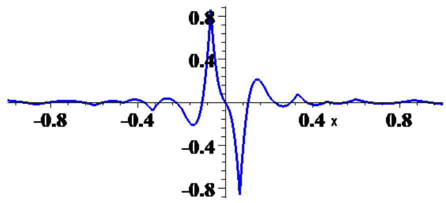

Fig. 21: The plot of the error of approximation of the first derivative of the smoothed polynomial approximation of the Runge function when using equidistant nodes ($n = 15$)
Source: created by the authors

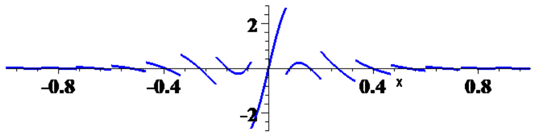

Fig. 22: The plot of the error of approximation of the first derivative of the Runge function with the piecewise linear polynomial approximation when using equidistant nodes ($n = 15$)
Source: created by the authors

Now we show the results of the trigonometric approximations.

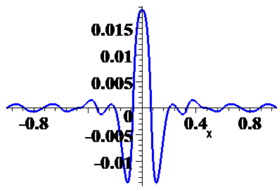

Fig. 23: The plot of the error of the smoothed trigonometric approximation of the Runge function when using equidistant nodes ($n = 15$)
Source: created by the authors

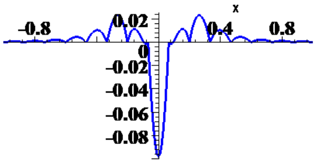

Fig. 24: The plot of the error of the piecewise trigonometric approximation of the Runge function when using equidistant nodes ($n = 15$)
Source: created by the authors

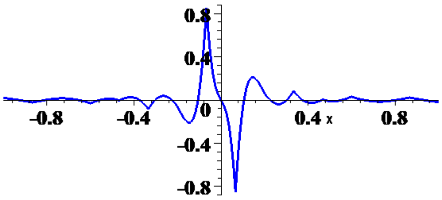

Fig. 25: The plot of the error of approximation of the first derivative of the Runge function with the smooth trigonometric approximation when using equidistant nodes ($n = 15$)
Source: created by the authors

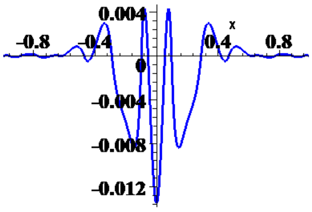

Fig. 26: The plot of the error of approximation of the Runge function with the smooth trigonometric approximation when using nodes taken as roots of the Chebyshev polynomial ($n = 15$)
Source: created by the authors

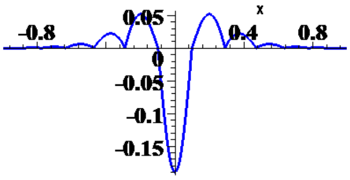

Fig. 27: The plot of the error of the approximation of the first derivative of the Runge function with the piecewise trigonometric approximation when using nodes taken as roots of the Chebyshev polynomial ($n = 15$)
Source: created by the authors

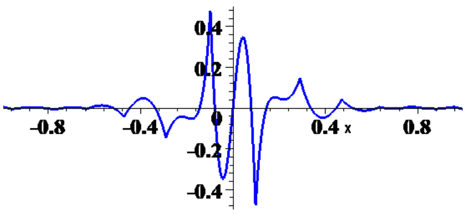

Fig. 28: The plot of the error of approximation of the first derivative of the Runge function with the smooth approximation when using nodes taken as roots of the Chebyshev polynomial ($n = 15$)
Source: created by the authors

The plot of the error of approximation of the function 3sin($2\pi x$)+4cos($3\pi x$) with the linear approximation when using an equidistant set of nodes ($n = 15$), is given in Figure 29.

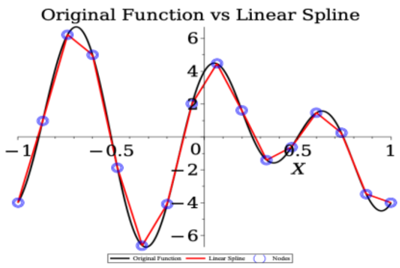

Fig. 29: The plot of the error of approximation of the function 3sin($2\pi x$)+4cos($3\pi x$) with the linear approximation when using an equidistant set of nodes ($n = 15$)
Source: created by the authors

The plot of the error of approximation of the function 3sin($2\pi x$)+4cos($3\pi x$) with the linear approximation when using an equidistant set of nodes, ($n = 15$) is given on Figure 30.

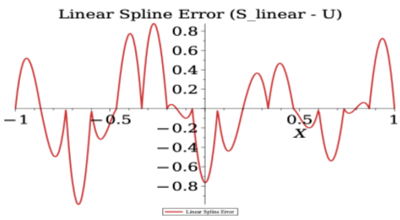

Fig. 30: The plot of the error of approximation of the function 3sin($2\pi x$)+4cos($3\pi x$) with the linear approximation when using an equidistant set of nodes ($n = 15$)
Source: created by the authors

The plot of approximation of the function 3sin($2\pi x$)+4cos($3\pi x$) with the smooth polynomial approximation when using an equidistant set of nodes, ($n = 15$), is given in Figure 31.

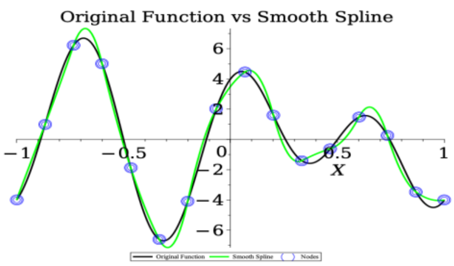

Fig. 31: The plot of approximation of the function 3sin($2\pi x$)+4cos($3\pi x$) with the smooth polynomial

approximation when using an equidistant set of nodes ($n = 15$)
Source: created by the authors

The plot of the error of approximation of the function 3sin(2πx)+4cos(3πx) with the smooth polynomial approximation when using an equidistant set of nodes ($n = 15$), is given in Figure 32.

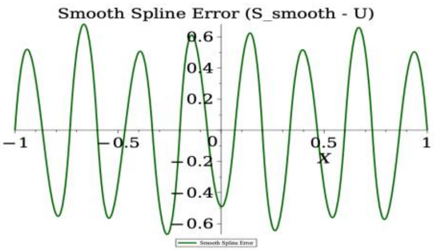

Fig. 32: The plot of the error of approximation of the function 3sin(2πx)+4cos(3πx) with the smooth polynomial approximation when using an equidistant set of nodes ($n = 15$)
Source: created by the authors

The plot of the approximation of the function e^{3x}-10e^{x}+e^{-2x} with the linear polynomial approximation when using an equidistant set of nodes ($n = 15$) is given in Figure 33.

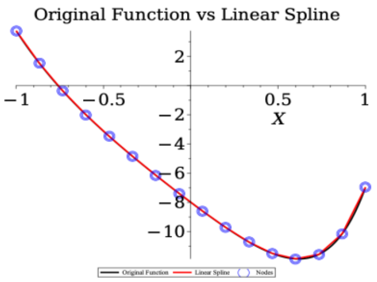

Fig. 33: The plot of approximation of the function e^{3x}-10e^{x}+e^{-2x} with the linear polynomial approximation when using an equidistant set of nodes n ($n = 15$)
Source: created by the authors

The plot of the error of approximation of the function e^{3x}-10e^{x}+e^{-2x} with the linear polynomial approximation when using an equidistant set of nodes ($n = 15$) is given in Figure 34.

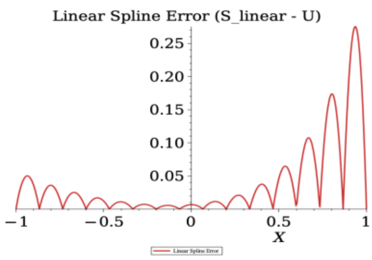

Fig. 34: The plot of the error of approximation of the function e^{3x}-10e^{x}+e^{-2x} with the linear polynomial approximation when using an equidistant set of nodes ($n = 15$)
Source: created by the authors

The plot of the approximation of the function e^{3x}-10e^{x}+e^{-2x} with the smooth polynomial approximation when using an equidistant set of nodes ($n = 15$) is given in Figure 35.

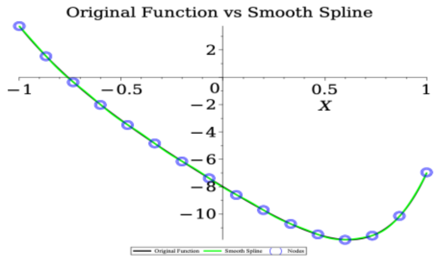

Fig. 35: The plot of approximation of the function e^{3x}-10e^{x}+e^{-2x} with the smooth polynomial approximation when using an equidistant set of nodes ($n = 15$)
Source: created by the authors

The plot of the error of approximation of the function e^{3x}-10e^{x}+e^{-2x} with the smooth polynomial approximation when using an equidistant set of nodes ($n = 15$) is given in Figure 36.

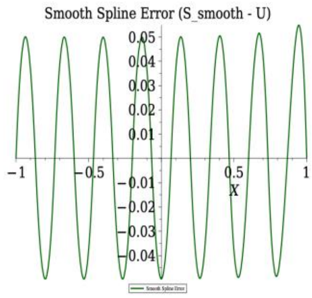

Fig. 36: The plot of the error of approximation of the function e^{3x}-10e^{x}+e^{-2x} with the smooth polynomial approximation when using an equidistant set of nodes ($n = 15$)
Source: created by the authors

The numerical results presented in Figure 1, Figure 2, Figure 3, Figure 4, Figure 5, Figure 6, Figure 7, Figure 8, Figure 9, Figure 10, Figure 11, Figure 12, Figure 13, Figure 14, Figure 15, Figure 16, Figure 17, Figure 18, Figure 19, Figure 20, Figure 21, Figure 22, Figure 23, Figure 24, Figure 25, Figure 26, Figure 27, Figure 28, Figure 29, Figure 30, Figure 31, Figure 32, Figure 33, Figure 34, Figure 35, and Figure. 36 demonstrate the results of the application of the proposed continuous and smooth spline approximations.

Table 1 and Table 2 show the approximation results for the Runge function when continuous and smooth splines were used. Table 3 and Table 4 below show the error of the first derivative of the Runge function when continuous and smooth splines were used. The interval $[-1,1]$ was considered, and a grid of uniform and non-uniform nodes ($n = 15$) was constructed. The roots of the Chebyshev polynomial were used as an example of how irregular grid nodes are used. The Tables show the maximum approximation errors for the grid intervals.

Table 1. The actual errors of approximations of the Runge function when polynomial approximations were used

$f(x)$	Equidistant nodes		Chebyshev nodes	
	cont.spl	smooth spl.	cont.spl.	smooth spl
$\frac{1}{1+25x^2}$	0.10	0.019	0.19	0.014

Source: created by the authors.

Table 2. The actual errors of approximations of the Runge function when trigonometric approximations were used

$f(x)$	Equidistant nodes		Chebyshev nodes	
	cont.spl.	smooth	cont.spl.	smooth spl.
$\frac{1}{1+25x^2}$	0.1	0.02	0.19	0.013

Source: created by the authors.

Table 3. The actual errors of approximations of the first derivatives of the Runge function when polynomial approximations were used

$f(x)$	Equidistant nodes		Chebyshev nodes	
	cont.spl	smooth spl.	cont.spl	smooth spl
$\frac{1}{1+25x^2}$	2.70	0.85	3.2	0.50

Source: created by the authors.

Table 4. The actual errors of approximations of the first derivatives of some of the Runge functions when trigonometric approximations were used

$f(x)$	Equidistant nodes		Chebyshev nodes	
	cont.spl.	smooth spl.	cont.spl.	smooth spl.
$\frac{1}{1+25x^2}$	2.64	0.85	3.1	0.5

Source: created by the authors.

Next, we consider how the approximation errors change when refining the grid of nodes.

Table 5, Table 6, Table 7, and Table 8 present the maximums (in absolute value) of the errors of approximations of functions $f(x)$ on a refined uniform grid of nodes for N = 8, 16, 32, 64, and 128 when the linear splines were used. Calculations were done in Maple when Digits=15, $N = n + 1$, $[a, b] = [-1,1]$.

The following notations are used in Table 5, Table 6, Table 7, and Table 8: $f_1(x) = x, f_2(x) = x^7 - 5x^5 + 9x^4 + 2x^2 - 4x + 3, f_3(x) = 3\sin 2\pi x + 4\cos 3\pi x$, $f_4(x) = e^{3x}$-10 e^x+e^{-2x}, $f_5(x) = \frac{1}{1+25x^2}$.

Table 5 shows the maximums (in absolute value) of the errors of linear polynomial approximations of functions $f(x)$.

Table 5. The maximums (in absolute value) of the errors of linear polynomial approximations of functions f(x)

$f(x)$	N = 8	N =16	N =32	N =64	N =128
$f_1(x)$	0.1 10^{-14}	0.1 10^{-14}	0.1 10^{-14}	0.1 10^{-14}	0.1 10^{-14}
$f_2(x)$	1.04	0.26	0.71 10^{-1}	0.17 10^{-1}	0.46 10^{-2}
$f_3(x)$	2.92	0.86	0.22	0.55 10^{-1}	0.13 10^{-1}
$f_4(x)$	0.90	0.22	0.61 10^{-1}	0.14 10^{-1}	0.39 10^{-2}
$f_5(x)$	0.33	0.2210^{-1}	0.60 10^{-2}	0.15 10^{-2}	0.38 10^{-3}

Source: created by the authors.

Table 6 presents the maximums (in absolute value) of the errors of smooth polynomial approximations of functions $f(x)$ on a refined uniform grid of nodes for N = 8, 16, 32, 64, and 128.

Table 6. The maximums (in absolute value) of the errors of smooth polynomial approximations of functions f(x)

$f(x)$	N = 8	N =16	N =32	N =64	N =128
$f_1(x)$	0.1 10^{-14}	0.1 10^{-14}	0.1 10^{-14}	0.1 10^{-14}	0.1 10^{-14}
$f_2(x)$	1.04	0.29	0.78 10^{-1}	0.20 10^{-1}	0.51 10^{-2}
$f_3(x)$	2.67	0.60	0.16	0.42 10^{-1}	0.11 10^{-1}
$f_4(x)$	0.20	0.44 10^{-1}	0.12 10^{-1}	0.31 10^{-2}	0.79 10^{-3}
$f_5(x)$	0.14	0.18 10^{-1}	0.21 10^{-2}	0.14 10^{-3}	0.14 10^{-4}

Source: created by the authors.

Table 7 presents the maximums (in absolute values) of the errors of linear trigonometric approximations of functions $f(x)$ on a refined uniform grid of nodes for N = 8, 16, 32, 64, and 128.

Table 7. The maximums (in absolute values) of the errors of linear trigonometric approximations of functions f(x)

$f(x)$	$N = 8$	$N = 16$	$N = 32$	$N = 64$	$N = 128$
$f_1(x)$	$0.69\ 10^{-2}$	$0.17\ 10^{-2}$	$0.43\ 10^{-3}$	$0.11\ 10^{-3}$	$0.28\ 10^{-4}$
$f_2(x)$	1.17	0.30	$0.79\ 10^{-1}$	$0.19\ 10^{-1}$	$0.51\ 10^{-2}$
$f_3(x)$	2.90	0.85	0.21	$0.55\ 10^{-1}$	$0.13\ 10^{-1}$
$f_4(x)$	0.73	0.21	$0.57\ 10^{-1}$	$0.13\ 10^{-1}$	$0.37\ 10^{-2}$
$f_5(x)$	$0.59\ 10^{-2}$	$0.50\ 10^{-1}$	$0.18\ 10^{-1}$	$0.51\ 10^{-2}$	$0.12\ 10^{-2}$

Source: created by the authors.

Table 8 presents the maximums (in absolute values) of the errors of smooth trigonometric approximations of functions $f(x)$ on a refined uniform grid of nodes for N = 8, 16, 32, 64, and 128.

Table 8. The maximums (in absolute value) of the errors of smooth trigonometric approximations of functions f(x)

$f(x)$	$N = 8$	$N = 16$	$N = 32$	$N = 64$	n =128
$f_1(x)$	$0.69\ 10^{-2}$	$0.18\ 10^{-2}$	$0.47\ 10^{-3}$	$0.12\ 10^{-3}$	$0.30\ 10^{-4}$
$f_2(x)$	1.17	0.33	$0.88\ 10^{-1}$	$0.23\ 10^{-1}$	$0.58\ 10^{-2}$
$f_3(x)$	2.67	0.59	0.16	$0.41\ 10^{-1}$	$0.11\ 10^{-1}$
$f_4(x)$	0.18	$0.50\ 10^{-1}$	$0.14\ 10^{-1}$	$0.35\ 10^{-2}$	$0.90\ 10^{-3}$
$f_5(x)$	$0.41\ 10^{-1}$	$0.69\ 10^{-1}$	$0.13\ 10^{-2}$	$0.17\ 10^{-3}$	$0.14\ 10^{-4}$

Source: created by the authors.

Consequently, when constructing a smoothed approximation, an oscillation may occur if the nodes are chosen poorly (see the example of approximating the Runge function on a uniform grid of nodes with $n = 16$, Figure 37).

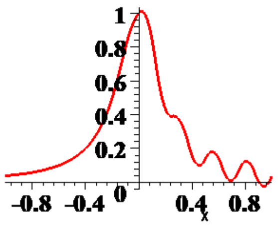

Fig. 37: The plot of the approximation of the Runge function by smoothed splines obtained for n = 16.
Source: created by the authors.

The plot of the error of approximation of the Runge function by smoothed splines obtained for $n = 16$ is shown in Figure 38.

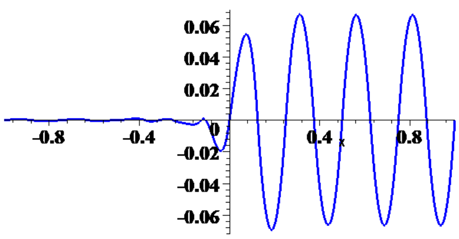

Fig. 38: The Runge function approximation error obtained for n = 16.
Source: created by the authors.

We can change the number of grid nodes (for example, we can increase the set of nodes by 1 node) and put $n = 17$). In this case, the error decreases significantly, and the oscillation disappears (Figure 39).

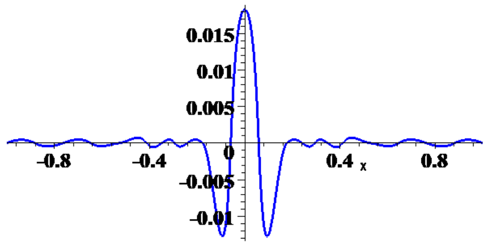

Fig. 39: The plot of the approximation error of the Runge function obtained for n = 17.
Source: created by the authors.

Note that as the number of nodes increases, the error decreases in all cases.

Remark. While working on responses to reviewers' comments, the authors discovered paper, [19], on the same topic, which examines the construction of smooth polynomial splines on an equidistant set of nodes. In paper, [19], the smoothing coefficient was selected based on the solution of an extremal problem. They minimized the norm of the first derivative.

In this paper, when constructing smooth splines, we based our approach on the idea of constructing smooth splines using only local continuous interpolation splines, both as polynomial and as non-polynomial, on both uniform and non-uniform grids. We take the first smoothing coefficient equal to zero. The resulting solution proposed by the authors of our paper is similar to the resulting solution in [19]; our way of construction is simpler and can be easily generalized to the non-polynomial splines on non-uniform grids and splines of arbitrary power.

4 Conclusion

This paper presents the results of theoretical and experimental aspects on approximation by smooth second-order splines using polynomial and non-polynomial splines. The construction of smoothed non-polynomial approximations is discussed using the example of constructing smooth trigonometric splines.

A comparison of the results obtained using piecewise continuous splines with piecewise smooth splines of the second-order splines is performed.

Future studies will consider the construction of the smooth approximations using other types of non-polynomial splines.

Acknowledgement:
The authors are gratefully indebted to the resource center of St. Petersburg University for providing the Maple package.

References:

[1] M. Kubus, L. Mach, P. Misiurski, 2025. Application of the nonlinear splines model to forecast changes in the construction costs index. *Statistics in Transition new series*, *26*(4), pp. 69-87 https://doi.org/10.59139/stattrans-2025-040.

[2] Peters J., Karčiauskas K., What smooth surfaces can be constructed from total degree 2 splines?, *Computer Aided Geometric Design*, Vol. 119, June 2025, 102435, https://doi.org/10.1016/j.cagd.2025.102435.

[3] Fareed A.F., Khattab A.G., Semary M.S., A novel stochastic ten non-polynomial cubic splines method for heat equations with noise term, *Partial Differential Equations in Applied Mathematics,* Vol.10, 2024, 100677. https://doi.org/10.1016/j.padiff.2024.100677.

[4] Yousif M.A, Hamasalh F.K., Conformable non-polynomial spline method: A robust and accurate numerical technique, *Ain Shams Engineering Journal*, Vol.15 (2), 2024, 102415. https://doi.org/10.1016/j.asej.2023.102415.

[5] Zürnacı F., Dişibüyük C., Non-polynomial divided differences and B-spline functions, *Journal of Computational and Applied Mathematics*, Vol.349, 2019, pp. 579–592. https://doi.org/10.1016/j.cam.2018.09.026.

[6] Khalida N., Abbas M., Iqbal M.K., Non-polynomial quintic spline for solving fourth-order fractional boundary value problems involving product terms, *Applied Mathematics and Computation*, Vol.349, 2019, pp.393-407, https://doi.org/10.1016/j.amc.2018.12.066.

[7] Aimi A., Diligenti M., Sampoli M.L., Sestini A., Non-polynomial spline alternatives in isogeometric symmetric Galerkin BEM, *Appl. Numer. Math.*, Vol.116, 2017, pp. 10-23. https://doi.org/10.1016/j.apnum.2016.07.004.

[8] Hammad D.A., Semary M.S., Khattab A.G., Ten non-polynomial cubic splines for some classes of Fredholm integral equations, *Ain Shams Engineering Journal*, Vol.13 (4), 2022, 101666. https://doi.org/10.1016/j.asej.2021.101666.

[9] Yousif M.A., Hamasalh F.K., The fractional non-polynomial spline method: Precision and modeling improvements, *Mathematics and Computers in Simulation*, Vol.218, 2024, pp.512–525. https://doi.org/10.1016/j.matcom.2023.11.033

[10] Sharma S., Sharma N., Exponential spline approach for analyzing n-soliton wave dynamics of good Boussinesq equation, *Physics Letters A*, Vol.553, 2025, 130698. https://doi.org/10.1016/j.physleta.2025.130698.

[11] Rani R., Arora G., Emadifar H., Khademi H., Numerical simulation of one-dimensional nonlinear Schrodinger equation using PSO with exponential B-spline, *Alexandria Engineering Journal,* Vol.79, 2023, pp. 644–651. https://doi.org/10.1016/j.aej.2023.08.050.

[12] Kaur M., Kapoor M., Numerical simulation of Volterra PIDE with singular kernel via modified cubic exponential and uniform algebraic trigonometric tension B-spline DQM, *Mathematics and Computers in Simulation,* Vol.226, 2024, pp. 438–451. https://doi.org/10.1016/j.matcom.2024.07.025

[13] Ali K.A., Mehanna M.S., Wazwaz A.M., Shaalan M.A., Solve third order Lane–Emden–Fowler equation by Adomian decomposition method and quartic trigonometric B-spline method, *Partial Differential Equations in Applied Mathematics*, Vol.10, 2024, 100676 https://doi.org/10.1016/j.padiff.2024.100676.

[14] Shi J., Rahman K., Deng J., Split-step quintic uniform algebraic trigonometric tension B-spline collocation method for cubic Ginzburg-Landau equations, *Computers and Mathematics with Applications,* Vol.196, 2025, 395–414. https://doi.org/10.1016/j.camwa.2025.08.003.

[15] El-shenawy A., El-Gamel M., Reda D., Troesch's problem: A numerical study with cubic trigonometric B-spline method, *Partial Differential Equations in Applied Mathematics*, Vol.10, 2024, 100694. https://doi.org/10.1016/j.padiff.2024.100694.

[16] Msmali A.H., Tamsir T., Ahmadin A.A.H, Unified and extended trigonometric B-spline DQM for the numerical treatment of three-dimensional wave equations, *Ain Shams Engineering Journal*, Vol.15 (2), 2024, 102382 https://doi.org/10.1016/j.asej.2023.102382.

[17] Barrera D., Elmokhtari F., Sbibih D., Two methods based on bivariate spline quasi-interpolants for solving Fredholm integral equations, *Applied Numerical Mathematics*, Vol.127, 2018, pp. 78–94. https://doi.org/10.1016/j.apnum.2017.12.016.

[18] Burova I.G., On Left Integro-Differential Splines and Cauchy Problem, *Int. J. Math. Models Methods Appl. Sci*., Vol. 9, 2015, pp. 683–690, [Online]. https://www.naun.org/main/NAUN/ijmmas/2015/b582001-015.pdf (Accessed Date: January 10, 2026).

[19] *(text in Russian)* Ingtem J.G., Minimal-norm-derivative spline function in interpolation and approximation, *Bulletin of Moscow University. Series 15. Computational Mathematics and Cybernetics*, Vol. 32, 2008, pp. 201-213. https://doi.org/10.3103/S0278641908040031.

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

- I. G. Burova was responsible for the theoretical results.
- Zhekai Lan carried out the simulation and the optimization.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself

No funding was received for conducting this study.

Conflict of Interest

The authors have no conflicts of interest to declare that are relevant to the content of this article.