DOI: 10.46300/9106.2020.14.61
Volume 14, 2020

Approximations with polynomial, trigonometric, exponential splines of the third order and boundary value problem

I.G.Burova, E.F.Muzafarova
St. Petersburg State University
7/9 Universitetskaya nab., St.Petersburg, 199034 Russia
Russia, burovaig@mail.ru, i.g.burova@spbu.ru

Received: June 11, 2020, Revised: August 3, 2020. Accepted: August 7, 2020. Published: August 13, 2020.

Abstract—This paper is devoted to the construction of local approximations of functions of one and two variables using the polynomial, the trigonometric, and the exponential splines. These splines are useful for visualizing flows of graphic information. Here, we also discuss the parallelization of computations. Some attention is paid to obtaining two-sided estimates of the approximations using interval analysis methods. Particular attention is paid to solving the boundary value problem by using the polynomial splines and the trigonometric splines of the third and fourth order approximation. Using the considered splines, formulas for a numerical differentiation are constructed. These formulas are used to construct computational schemes for solving a parabolic problem. Questions of approximation and stability of the obtained schemes are considered. Numerical examples are presented.

Keywords— Boundary value problem, exponential splines, interpolation, interval estimation, polynomial splines, trigonometric splines, exponential splines.

I. Introduction

QUITE often, interval analysis is used to verify the result. In papers [1]–[4] an overview on applications of interval arithmetic is given and verification methods for solving linear systems of equations, nonlinear systems, the algebraic eigenvalue problem and initial value problems are discussed. In paper [3] interval estimation is used in epidemiological analysis.

It is an important task to determine the upper and lower boundaries of solutions (see [5]-[6]). Two-sided estimates allow the verification to solve the problem. In paper [5] an approach for solving non-linear systems of equations is proposed. This approach is based on the Interval-Newton and Interval-Krawczyk operators and B-splines. The proposed algorithm is making great benefits of the geometric properties of B-spline functions to avoid unnecessary computations. For eigenvalue problems of self-adjoint differential operators, a universal framework is proposed to give explicit lower and upper bounds for the eigenvalues (see [6]).

An important aspect of solving the problem is to improve the calculation accuracy (see [7]-[8]). The interval analysis technique and radial point interpolation method are adopted in [7] in order to improve the calculation accuracy and reduce the computational cost. The corresponding formulations of the structural acoustic system for the interval response analysis are deduced in [7] too. When processing flows of graphic information, great importance is often given to the compression and subsequent recovery of this information. The problems of data compression and visualization using radial basis functions are discussed in [8]. When solving various problems, various splines are very often used. Splines have proven themselves well in solving problems of approximation and functions, in visualizing the results of calculations. They often provide a solution with less error. A great contribution to the development of spline theory was made by J.H.Ahlberg, E.N.Nilson, J.L.Walsh, Carl de Boor, and other mathematicians. Currently, polynomial splines are widely used. Non-polynomial splines are less well known, but often provide a smaller approximation error. It should be noted that Prof. Yu.K.Demyanovich devotes a lot of attention to the study of quadratic polynomial splines of the Lagrangian type (see [9]). In paper [15], methods are given for constructing splines of generalized smoothness in the case of splines of the Lagrangian type on a differentiable manifold. In paper [16] methods for constructing splines of generalized smoothness are developed on a manifold and an application of the results obtained to splines of the Hermitian type of the first height is given.

Recently, many authors have been trying to improve computational schemes for solving partial differential equations. When solving this problem, splines are very often used. Explicit formulae were developed to obtain the different derivatives of the linear partial differential equations in paper [17]. In paper [18], when solving the heat conduction problem, two types of basis functions are considered: B-spline and exponential B-spline combined with Bernstein polynomials. Paper [19] presents a numerical algorithm for using radial basis function-generated finite differences to solve partial differential equations on S2 using polyharmonic splines with added polynomials defined in a 2D plane. In paper [20] the hybrid spline difference method is used to solve the one-dimensional heat transfer equations. A novel multistep method based on the non-uniform rational basis spline curves is developed in [21] for the solution of a system of nonlinear differential equations.

This paper discusses the issues of visualizing results, reducing counting time, and verification of calculations. This paper continues the series of papers on approximation with local polynomial and non-polynomial splines and interval estimation (see [10] – [14]). This paper focuses on the polynomial, the trigonometric and the exponential splines of the third order approximation. To construct the approximation, we need the values of the function at the grid nodes and the basis formulas splines. To construct the interval estimation of the approximation of the function or its first derivative, we need the function values at the grid nodes and the rules for working with real intervals. The proposed splines can be used to construct numerical methods for solving partial differential equations. In Section 6, we consider the application of polynomial and trigonometric splines to the solution of boundary value problems. It should be noted that different approaches lead to numerical differentiation formulas with different properties

II. The left and the right splines

In some cases, the use of the trigonometric approximations is preferable to the polynomial approximations. Here we compare these two types of approximations. To approximate functions on a finite grid of nodes, we will use the left and right splines.

Suppose a, b are real numbers. We will apply left splines near the right end of the finite interval $[a,b]$. Right splines will be applied near the left end of the finite interval $[a,b]$. Let the set of nodes x_j be such that $a < \ldots < x_{j-1} < x_j < x_{j+1} < \ldots < b$. We construct an approximation of function $f(x), f \in C^{(3)}([a,b])$ with local splines, in which the support of the bases spline consists of three adjacent intervals. When approximating a function on a finite interval near the left and right boundaries of the interval $[a,b]$ we will use the approximation $F^L(x), G^L(x), Q^L(x), x \in [x_j, x_{j+1}]$, with the left or $F^R(x), G^R(x), Q^R(x), x \in [x_j, x_{j+1}]$ with the right continuous splines:

$$F^L(x) = f(x_{j-1})w_{j-1}^L(x) + f(x_j)w_j^L(x) + f(x_{j+1})w_{j+1}^L(x),$$

$$F^R(x) = f(x_j)W_j^R(x) + f(x_{j+1})W_{j+1}^R(x) + f(x_{j+2})W_{j+2}^R(x),$$

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

$$G^R(x) = f(x_j)v_j^R(x) + f(x_{j+1})v_{j+1}^R(x) + f(x_{j+2})v_{j+2}^R(x).$$

$$Q^L(x) = f(x_{j-1})\alpha_{j-1}^L(x) + f(x_j)\alpha_j^L(x) + f(x_{j+1})\alpha_{j+1}^L(x),$$

$$Q^R(x) = f(x_j)\alpha_j^R(x) + f(x_{j+1})\alpha_{j+1}^R(x) + f(x_{j+2})\alpha_{j+2}^R(x).$$

Approximations using the trigonometric splines will be denoted by $F^L(x), F^R(x)$. Approximations using the polynomial splines will be denoted by $G^L(x)$, $G^R(x)$. Approximations using the exponential splines will be denoted by $Q^L(x)$, $Q^R(x)$.

The set of interpolation with the local left and right splines are called boundary minimal splines.

In paper [14] it is shown that the left trigonometric basis be written as follows:

$$w_j^L(x) = \frac{\cos\left(x - \frac{x_{j-1}}{2} - \frac{x_{j+1}}{2}\right) - \cos\left(\frac{x_{j-1}}{2} - \frac{x_{j+1}}{2}\right)}{2\sin\left(\frac{x_j}{2} - \frac{x_{j+1}}{2}\right)\sin\left(\frac{x_{j+1}}{2} - \frac{x_j}{2}\right)},$$

$$w_{j+1}^L(x) = \frac{\cos\left(\frac{x_j}{2} - \frac{x_{j-1}}{2}\right) - \cos\left(\frac{x_j}{2} + \frac{x_{j-1}}{2} - x\right)}{2\sin\left(\frac{x_{j+1}}{2} - \frac{x_{j-1}}{2}\right)\sin\left(\frac{x_{j+1}}{2} - \frac{x_j}{2}\right)},$$

$$w_{j-1}^L(x) = \frac{\cos\left(\frac{x_j}{2} - \frac{x_{j+1}}{2}\right) - \cos\left(\frac{x_j}{2} + \frac{x_{j+1}}{2} - x\right)}{2\sin\left(\frac{x_{j-1}}{2} - \frac{x_j}{2}\right)\sin\left(\frac{x_{j-1}}{2} - \frac{x_{j+1}}{2}\right)}.$$

We consider that $supp\, w_j^L(x) = [x_{j-2}, x_{j+1}]$. The left trigonometric basis splines we obtain from the system of equation $F^L(x) = f(x)$, when $f(x) = 1, \sin(x), \cos(x)$, $x \in [x_j, x_{j+1}]$.

In paper [14] it is shown that the left polynomial basis splines can be written as follows:

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

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

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

We consider that $supp\, \omega_j^L(x) = [x_{j-2}, x_{j+1}]$. The left polynomial basis splines we obtain from the system of equation $G^L(x) = f(x)$, when $f(x) = 1, x, x^2$, for $x \in [x_j, x_{j+1}]$. Using the notation $x = x_j + th$, $t \in [0,1]$, $x_{j+1} = x_j + h$, $x_{j-1} = x_j - h$, we get

$$\omega_j^L(x_j + th) = -(t-1)(t+1),$$

$$\omega_{j+1}^L(x_j + th) = t(t+1)/2,$$

$$\omega_{j-1}^L(x_j + th) = t(t-1)/2.$$

The relationships $w_j^L(x_j + th) = \omega_j^L(x_j + th) + O(h^2)$, $w_{j+1}^L(x_j + th) = \omega_j^L(x_j + th) + O(h^2), w_{j-1}^L(x_j + th) = \omega_{j-1}^L(x_j + th) + O(h^2)$ establish the relations between left polynomial and trigonometric splines.

We consider that $supp\, W_j^R(x) = [x_{j-1}, x_{j+2}]$. The right trigonometric basis splines we obtain from the system of equation $F^R(x) = f(x)$, when $f(x) = 1, \sin(x), \cos(x)$, for $x \in [x_j, x_{j+1}]$. Formulas for right polynomial and trigonometric splines are given in paper [14]:

$$W_j^R(x) = \frac{\cos\left(\frac{x_{j+1}}{2} - \frac{x_{j+2}}{2}\right) - \cos\left(x - \frac{x_{j+2}}{2} - \frac{x_{j+1}}{2}\right)}{2\sin\left(\frac{x_j}{2} - \frac{x_{j+2}}{2}\right)\sin\left(\frac{x_j}{2} - \frac{x_{j+1}}{2}\right)},$$

$$W_{j+1}^R(x) = \frac{\cos\left(x - \frac{x_{j+2}}{2} - \frac{x_j}{2}\right) - \cos\left(\frac{x_j}{2} - \frac{x_{j+2}}{2}\right)}{2\sin\left(\frac{x_{j+1}}{2} - \frac{x_{j+2}}{2}\right)\sin\left(\frac{x_j}{2} - \frac{x_{j+1}}{2}\right)},$$

$$W_{j+2}^R(x)=\frac{\cos\left(\frac{x_j}{2}-\frac{x_{j+1}}{2}\right)-\cos\left(x-\frac{x_j}{2}-\frac{x_{j+1}}{2}\right)}{2\sin\left(\frac{x_j}{2}-\frac{x_{j+2}}{2}\right)\sin\left(\frac{x_{j+1}}{2}-\frac{x_{j+2}}{2}\right)}.$$

The formulas for the right polynomial basis splines $v_j^R(x)$, $v_{j+1}^R(x), v_{j+2}^R(x)$, $x\in[x_j,x_{j+1}]$, can be written as follows:

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

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

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

When $x=x_j+th,\ t\in[0,1]$, we obtain for $x\in[x_j,x_{j+1}]$:

$$v_j^R(x_j+th)=1-(3/2)t+t^2/2,$$

$$v_{j+1}^R(x_j+th)=2t-t^2,$$

$$v_{j+2}^R(x_j+th)=t^2/2-t/2.$$

The left (or right) exponential basis splines we obtain from the system of equation $Q^{L\alpha}(x)=f(x)$ (or $Q^R(x)=f(x)$), when $f(x)=1,\exp(x),\ \exp(-x)$, for $x\in[x_j,x_{j+1}]$. For the left exponential splines we have:

$$Q^{L\alpha}(x)=f(x_{j-1})\alpha_{j-1}^L(x)+f(x_j)\alpha_j^L(x)+f(x_{j+1})\alpha_{j+1}^L(x),$$

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

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

where

$$A_{j-1}=\exp(x_j-x_{j+1})-\exp(x_{j+1}-x_j)+\exp(x-x_j)$$
$$-\exp(x-x_{j+1})-\exp(x_j-x)+\exp(x_{j+1}-x),$$

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

where

$$A_j=\exp(x-x_{j+1})-\exp(x_{j+1}-x)+\exp(x_{j-1}-x)+$$
$$-\exp(x_{j-1}-x_{j+1})+\exp(x_{j+1}-x_{j-1})-\exp(x-x_{j-1}),$$

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

where

$$A_{j+1}=\exp(x-x_j)-\exp(x_j-x)-\exp(x_{j-1}-x_j)$$
$$+\exp(x_{j-1}-x)-\exp(x-x_{j-1})+\exp(x_j-x_{j-1}).$$

We consider that $supp\ \alpha_j^L(x)=[x_{j-2},x_{j+1}]$. The formula of the basis spline $\alpha_j^L(x)$ when $x\in[x_{j-1},x_j]$, we obtain from $Q^{L\alpha}(x)=f(x)$, when $f(x)=1,\exp(x),\ \exp(-x)$, where $Q^{L\alpha}(x)=f(x_{j-2})\alpha_{j-2}^L(x)+f(x_{j-1})\alpha_{j-1}^L(x)+f(x_j)\alpha_j^L(x)$.

The formula of the basis spline $\alpha_j^L(x)$ when $x\in[x_{j-2},x_{j-1}]$, we obtain from $Q^{L\alpha}(x)=f(x)$, when $f(x)=1,\exp(x)$, $\exp(-x)$, where $Q^{L\alpha}(x)=f(x_j)\alpha_j^L(x)+f(x_{j+1})\alpha_{j+1}^L(x)+f(x_{j+2})\alpha_{j+2}^L(x)$.

Combine the basis spline formulas obtained at these intervals we have a formula on the interval $[x_{j-2},x_{j+1}]$. The plot of the exponential basis spline $\alpha_j^L(x)$, when $h=1,\ j=-1$, is given in Fig 1.

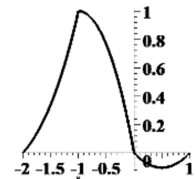

Fig. 1. The plot of the exponential basis spline α_j^L

Denote $f_k=f(x_k)$. When we have the equidistant set of nodes, $x_{j+1}-x_j=x_j-x_{j-1}=h$, then the formula of the first derivative of the approximation at $x=x_j$ is as follows:

$$(Q^L(x))'|_{x=x_j}=(f_{j-1}g_{j-1}+f_jg_j+f_{j+1}g_{j+1}),$$

where $g_j=(\alpha_j^L(x))'|_{x=x_j}=0$,

$$g_{j-1}=(\alpha_{j-1}^L(x))'|_{x=x_j}=\frac{-\exp(h)}{(\exp(h)-1)(\exp(h)+1)},$$

$$g_{j+1}=(\alpha_{j+1}^L(x))'|_{x=x_j}=\frac{\exp(h)}{(\exp(h)-1)(\exp(h)+1)}.$$

We can obtain the formula when $x=x_j+th, t\in[0,1]$:

$$g_{j-1}=-\frac{1}{2h}+\frac{h}{12}+O(h^3),\ g_{j+1}=\frac{1}{2h}-\frac{h}{12}+O(h^3),$$

Thus, when we have the equidistant set of nodes, the formula $f'(x_j)=(Q^{L\alpha}(x))'|_{x=x_j}+O(h)$ can be used.

Remark 1. We can also obtain the left (or right) exponential basis splines from the system of equation $Q^{L\beta}(x)=f(x)$ (or $Q^{R\beta}(x)=f(x)$), when $f(x)=1,\exp(-x),\ \exp(-2x)$, $x\in[x_j,x_{j+1}]$. For the left exponential splines we have:

$$Q^{L\beta}(x)=f(x_{j-1})\beta_{j-1}^L(x)+f(x_j)\beta_j^L(x)+f(x_{j+1})\beta_{j+1}^L(x),$$

where

$$\beta_{j-1}^L(x)=\frac{B_{j-1}\exp(2x_{j+1})\exp(2x_j)\exp(2x_{j-1})}{\exp(x_{j+1}-x_{j-1})\exp(x_j-x_{j-1})\exp(x_j-x_{j+1})},$$

$$B_{j-1}=\exp(-x_j-2x_{j+1})-\exp(-x_{j+1}-2x_j)$$
$$+\exp(-x-2x_j)-\exp(-x-2x_{j+1})$$
$$+\exp(-x_{j+1}-2x)-\exp(-x_j-2x),$$

$$\beta_j^L(x)=\frac{B_j\exp(2x_{j+1})\exp(2x_j)\exp(2x_{j-1})}{\exp(x_{j+1}-x_{j-1})\exp(x_j-x_{j-1})\exp(x_j-x_{j+1})},$$

where $B_j=\exp(-x-2x_{j+1})-\exp(-x_{j+1}-2x)$

$$-\exp(-x_{j-1}-2x_{j+1})+\exp(-x_{j+1}-2x_{j-1})$$
$$-\exp(-2x_{j-1}-x)+\exp(-2x-x_{j-1}),$$

$$\beta_{j+1}^L(x) = \frac{B_{j+1}\exp(2x_{j+1})\exp(2x_j)\exp(2x_{j-1})}{\exp(x_{j+1}-x_{j-1})\exp(x_j-x_{j-1})\exp(x_j-x_{j+1})},$$

where $B_{j+1} = \exp(-x-2x_j) - \exp(-x_j-2x)$

$$-\exp(-x_{j-1}-2x_j) + \exp(-x_{j-1}-2x)$$

$$-\exp(-2x_{j-1}-x) + \exp(-2x_{j-1}-x_j).$$

The plot of the exponential basis spline $\beta_j^L(x)$, when $h=1, j=-1$, is given in Fig 2.

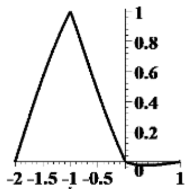

Fig. 2. The plot of the exponential basis spline β_j^L

When we have the equidistant set of nodes, when $x_{j+1}-x_j = x_j - x_{j-1} = h$, then the formula of the first derivative of the approximation at $x = x_j$ is as follows:

$$Q^{L\beta}(x))'|_{x=x_j} = (f_{j-1}g_{j-1} + f_j g_j + f_{j+1}g_{j+1}),$$

where $g_j = (\beta_j^L(x))'|_{x=x_j} = -1$,

$$g_{j-1} = (\beta_{j-1}^L(x))'|_{x=x_j} = \frac{-1}{(\exp(h)-1)(\exp(h)+1)},$$

$$g_{j+1} = (\beta_{j+1}^L(x))'|_{x=x_j} = \frac{\exp(2h)}{(\exp(h)-1)(\exp(h)+1)}.$$

We can obtain that when $x = x_j + th, t \in [0,1]$:

$$g_{j-1} = -\frac{1}{2h} + \frac{1}{2} + O(h),\ g_{j+1} = \frac{1}{2h} + \frac{1}{2} + O(h).$$

Thus, this variant of the approximation of the first derivative with $Q^{L\beta}$ is not very good and is not recommended for the calculation of the first derivative of the function.

Remark 2. We can also obtain the left (or right) exponential basis splines from the system of equation $Q^{L\gamma}(x) = f(x)$(or $Q^{R\gamma}(x) = f(x)$), when $f(x) = 1, \exp(x), \exp(2x)$, $x \in [x_j, x_{j+1}]$. For the left exponential splines we have:

$$Q^{L\gamma}(x) = f(x_{j-1})\gamma_{j-1}^L(x) + f(x_j)\gamma_j^L(x) + f(x_{j+1})\gamma_{j+1}^L(x),$$

where

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

where

$$C_{j-1} = \exp(x_j+2x_{j+1}) - \exp(x_{j+1}+2x_j)$$

$$+\exp(x+2x_j) - \exp(x+2x_{j+1})$$

$$-\exp(x_j+2x) + \exp(x_{j+1}+2x),$$

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

where

$$C_j = \exp(x+2x_{j+1}) - \exp(2x+x_{j+1})$$

$$+\exp(2x+x_{j-1}) - \exp(x_{j-1}+2x_{j+1})$$

$$+\exp(x_{j+1}+2x_{j-1}) - \exp(2x_{j-1}+x),$$

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

where

$$C_{j+1} = \exp(2x+x_j) - \exp(x+2x_j)$$

$$+\exp(x_{j-1}+2x_j) - \exp(2x+2x_{j-1})$$

$$+\exp(x+2x_{j-1}) - \exp(2x_{j-1}+x_j)\ .$$

The plot of the exponential basis spline $\gamma_j^L(x)$, when $h=1, j=-1$, is given in Fig 3.

When we have the equidistant set of nodes, when $x_{j+1}-x_j = x_j - x_{j-1} = h$, then the formula of the first derivative of the approximation at $x = x_j$ is as follows:

$$Q^{L\gamma}(x))'|_{x=x_j} = (f_{j-1}g_{j-1} + f_j g_j + f_{j+1}g_{j+1}),$$

where $g_j = (\gamma_j^L(x))'|_{x=x_j} = 1$,

$$g_{j-1} = (\gamma_{j-1}^L(x))'|_{x=x_j} = \frac{-\exp(2h)}{(\exp(h)-1)(\exp(h)+1)},$$

$$g_{j+1} = (\gamma_{j+1}^L(x))'|_{x=x_j} = \frac{1}{(\exp(h)-1)(\exp(h)+1)}.$$

We can obtain that when $x = x_j + th, t \in [0,1]$.

$$g_{j-1} = -\frac{1}{2h} - \frac{1}{2} + O(h),\ g_{j+1} = \frac{1}{2h} + \frac{1}{2} + O(h).$$

Thus, this variant of the approximation of the first derivative of the function $f(x)$ with $(Q^{L\gamma})'$ is not very good and is not recommended for the calculation of the first derivative of the function.

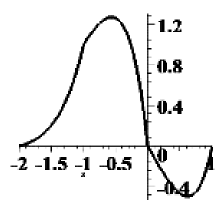

Fig. 3. The plot of the basis spline γ_j^L

Theorem 1. Let function $f(x)$ be such that $f \in C^{(3)}([\alpha,\beta]), [\alpha,\beta] \subset [a,b]$. The set of nodes such that $x_{j+1} - x_j = x_j - x_{j-1} = h$. Then for $x \in [x_j, x_{j+1}]$ we have

1) $\|f - G^L\|_{[x_j, x_{j+1}]} \le K_1 h^3 \|f'''\|_{[x_{j-1}, x_{j+1}]}$,
2) $\|f - G^R\|_{[x_j, x_{j+1}]} \le K_1 h^3 \|f'''\|_{[x_j, x_{j+2}]}$,
3) $\|f - F^L\|_{[x_j, x_{j+1}]} \le K_2 h^3 \|f''' + f''\|_{[x_{j-1}, x_{j+1}]}$,
4) $\|f - F^R\|_{[x_j, x_{j+1}]} \le K_2 h^3 \|f''' + f''\|_{[x_j, x_{j+2}]}$,
5) $\|f - Q^{L\alpha}\|_{[x_j, x_{j+1}]} \le K_3 h^3 \|f''' - f'\|_{[x_{j-1}, x_{j+1}]}$,

where $K_1 = \frac{0.385}{3!} \approx 0.0642, K_2 = 0.0835, K_3 = 0.12$.

Proof. The method of the proof of the statements 1)-5) is given in [12]. The proofs of the statements 1)-4) can be seen in paper

[14]. In short we explain the estimation 5) when the basis functions are constructed from the condition $f = Q^{L\alpha}$, $f = 1, \exp(x), \exp(-x)$. Function $f(x)$ can be written, as follows:

$$f(x) = \int_{x_j}^{x} (f''' - f')(-2 + e^{t-x} + e^{x-t})dt.$$

The method of the construction this representation of $f(x)$ can be seen in [12]. Using this formula we construct the estimation 5).

The plots of the errors of approximations of functions and the first derivatives of these functions are given in Figures 4-7. Fig.4 (left) shows the error of the approximation of function $f = \sin(3x)$ which was obtained with the use the exponential spline $Q^{L\alpha}$. Fig.4 (right) shows the error of the approximation of the first derivative of the function $f = \sin(3x)$ which was obtained with the use the exponential spline $Q^{L\alpha}$. Here $h = 0.1$.

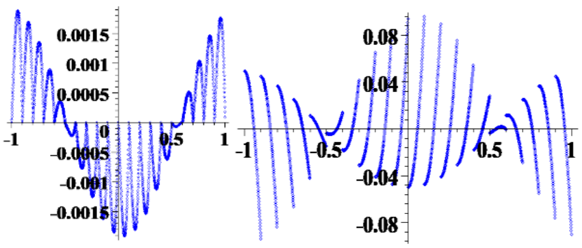

Fig.4. The error of the approximation of function $f = sin(3x)$ (left), and the error of the approximation of $f'(x)$ obtained with the use the exponential spline $Q^{L\alpha}$ (right)

Fig.5 (left) shows the error of the approximation of function $f = \sin(3x)$ which was obtained with the use the exponential spline $Q^{L\beta}$. Fig.5 (right) shows the error of the approximation of the first derivative of the function $f = \sin(3x)$ which was obtained with the use the exponential spline $Q^{L\beta}$. Here $h = 0.1$.

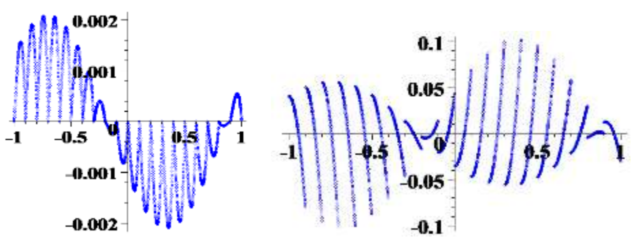

Fig.5. The error of the approximation of function $f = \sin(3x)$ (left) and the error of the approximation of $f'(x)$ with the use the exponential spline $Q^{L\beta}$(right)

Fig.6 (left) shows the error of the approximation of function $f = \sin(3x)$ which was obtained with the use the exponential spline $Q^{L\gamma}$. Fig.6 (right) shows the error of the approximation of the first derivative of the function $f = \sin(3x)$ which was obtained with the use the exponential spline $Q^{L\gamma}$. Here $h = 0.1$.

Table 1 shows the theoretical and actual errors of approximation with the trigonometrical splines in the interval $[-1,1]$ with the grid step of $h = 0.1$. Table 2 shows the theoretical and actual errors of approximation with the exponential splines $Q^{L\alpha}$ in the interval $[-1,1]$ with the grid step of $h = 0.1$.

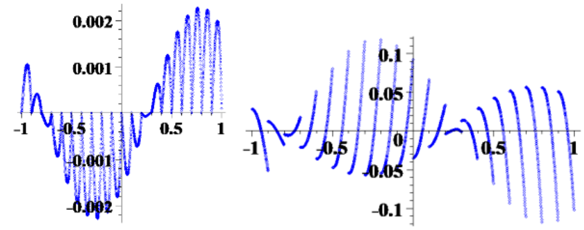

Fig.6. The error of the approximation of function $f = sin(3x)$ (left) and the error of the approximation of $f'(x)$ with the use the exponential spline $Q^{L\gamma}$(right)

Table 3 shows the theoretical and actual errors of approximation with the polynomial splines in the interval $[-1,1]$ with the grid step of $h = 0.1$. Table 4 shows the actual errors of approximation with the exponential splines $Q^{L\beta}$ and $Q^{L\gamma}(x)$ in the interval $[-1,1]$ with the grid step of $h = 0.1$.

Table 1. The theoretical and actual errors of approximation with the trigonometrical splines

$f(x)$	actual err.	theoret. err.
$\frac{\sin x}{1+25x^2}$	$0.71 \cdot 10^{-2}$	$0.13 \cdot 10^{-1}$
$\sin(\frac{2x}{25})\cos(\frac{2}{25}+\frac{x}{2})$	$0.12 \cdot 10^{-5}$	$0.16 \cdot 10^{-5}$
$\sin(\frac{2x}{25})\cos(2x+\frac{1}{50})$	$0.56 \cdot 10^{-4}$	$0.74 \cdot 10^{-4}$

Table 2. The theoretical and actual errors of approximation with the exponential splines $Q^{L\alpha}$

f	actual err.	theoret. err.
$\frac{\sin(x)}{1+25x^2}$	$0.72 \cdot 10^{-2}$	$0.18 \cdot 10^{-1}$
$\sin(\frac{2x}{25})\cos(\frac{2}{25}+\frac{x}{2})$	$0.90 \cdot 10^{-5}$	$0.17 \cdot 10^{-4}$
$\sin(\frac{2x}{25})\cos(2x+\frac{1}{50})$	$0.71 \cdot 10^{-4}$	$0.12 \cdot 10^{-3}$

Table 3. The theoretical and actual errors of approximation with the polynomial splines

$f(x)$	Actual err.	Theoret. err.
$\frac{\sin x}{1+25x^2}$	$0.72 \cdot 10^{-2}$	$0.97 \cdot 10^{-2}$
$\sin(\frac{2x}{25})\cos(\frac{2}{25}+\frac{x}{2})$	$0.39 \cdot 10^{-5}$	$0.39 \cdot 10^{-5}$
$\sin(\frac{2x}{25})\cos(2x+\frac{1}{50})$	$0.61 \cdot 10^{-4}$	$0.62 \cdot 10^{-4}$

Table 4. The actual errors of approximation with the exponential splines $Q^{L\gamma}$ and . $Q^{L\beta}$

$f(x)$	actual err. $Q^{L\gamma}$	actual err. $Q^{L\beta}$
$\frac{\sin x}{1+25x^2}$	$0.71 \cdot 10^{-2}$	$0.71 \cdot 10^{-2}$
$\sin(\frac{2x}{25})\cos(\frac{2}{25}+\frac{x}{2})$	$0.15 \cdot 10^{-4}$	$0.13 \cdot 10^{-4}$
$\sin(\frac{2x}{25})\cos(2x+\frac{1}{50})$	$0.95 \cdot 10^{-4}$	$0.86 \cdot 10^{-4}$

III. Interval extension

As is known, the task of interval estimation is to find the narrowest possible estimation interval.

For interval estimation of approximation with splines, we will use operations on intervals (see, for example, book [1]). Interval result over real intervals $A = [a_1, a_2]$ and $B = [b_1, b_2]$ can be obtained using the formulas:

1. $A + B = [a_1 + b_1, a_2 + b_2]$,
2. $A - B = [a_1 - b_2, a_2 - b_1] = A + [-1,1] \cdot B$,
3. $A \cdot B = \min\{a_1b_1, a_1b_2, a_2b_1, a_2b_2\}, \max\{a_1b_1, a_1b_2, a_2b_1, a_2b_2\}]$,
4. $A : B = [a_1, a_2,] \cdot [1/b_2, 1/b_1], 0 \notin B$.

For a unary operation we use the rule:

5. $r(A) = \left[\min_{x\in A}(r(x)), \max_{x\in A}(r(x))\right]$, where $r(A)$ is the unary operation.

Theorem 1 helps us to choose the correct length $h = x_{j+1} - x_j$ of the interval $[x_j, x_{j+1}]$.

Suppose we know the values of function $f(x)$ at nodes $\{x_k\}$. Using formulas of trigonometrical splines and the technique of interval analysis [1] we can construct the upper and lower boundaries for every interval $Y_j = [x_j, x_{j+1}]$. Thus, we avoid the calculations of approximation $f(x)$ in many points of every interval $[x_j, x_{j+1}]$ if we need to know the boundaries of the interval, where the function $f(x)$ varieties. In order to obtain the boundaries of variety $f(x)$ we construct the approximation $F(x), x \in Y_j$ and consider $F(Y_j)$.

In order to get the narrowest estimation interval we consider formulas for the left and the right basis trigonometric splines. First, we consider the estimate of the lower bound of the estimating interval of the basis spline $w_{j-1}(x)$.

Let x_{j-1}^{max} be the maximum

$$x_{j-1}^{max} = \max_{x\in[x_j,x_{j+1}]}(\cos\left(\frac{x_j}{2} - x + \frac{x_{j+1}}{2}\right).$$

Then the upper boundary of $w_{j-1}(x)$ will be the following

$$w_{j-1}^{MA} = 2\sin(x_j/2 - x_{j+1}/2)/\sin(x_j - x_{j-1}) - \sin(x_j - x_{j-1}) - \sin(x_j - x_{j-1})x_{j-1}^{max} + w_{j-1}^{A},$$

where $w_{j-1}^{A} = \sin(x_{j+1} - x_j)/\sin(x_j - x_{j-1}) - \sin(x_{j+1} - x_{j-1}) - \sin(x_j - x_{j+1})$.

After calculating the upper boundaries of $w_{j-1}(x), w_j(x)$ and $w_{j+1}(x)$ we can calculate the upper boundary of $F(x)$. Now the upper boundary of $F(x)$ will be the following:

$$F^{MAX} = f(x_{j-1})w_{j-1}^{MA} + f(x_j)w_j^{MA} + f(x_{j+1})w_{j+1}^{MA}.$$

A program was developed in the MAPLE environment to visualize the interval estimation of the variation of a function and its first derivative. To obtain an interval estimate of the function or its first derivative, values of the function in grid nodes are required. The program uses trigonometric basic splines. Directional machine rounding is not used in this version of the program.

Note that in the case of applying a similar method of interval estimation using polynomial quadratic splines resulting evaluating the interval is wider than in the case of trigonometric splines. As shown in Alefeld's book [1], a polynomial of the second degree $x^2 + b^{(1)}x + b^{(0)}$ should be reduced to $(x + a^{(1)})^2 + a^{(0)}$, where $a^{(1)} = b^{(1)}/2$, $a^{(0)} = b^{(0)} - (b^{(1)})^2/4$. Fig. 7 shows the estimation interval for the function $x^2/625$ after applying polynomial splines. Fig. 8 shows the estimation interval for the function $x^2/625$ after applying trigonometric splines.

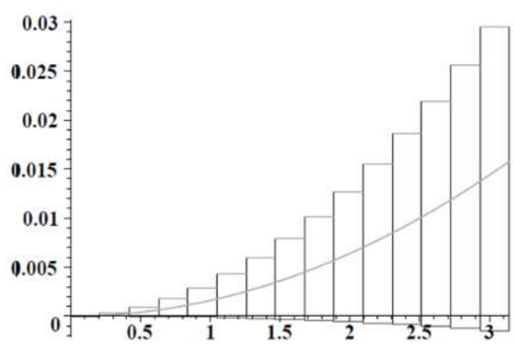

Fig. 7. The estimation interval for the function $x^2/625$ after applying polynomial splines

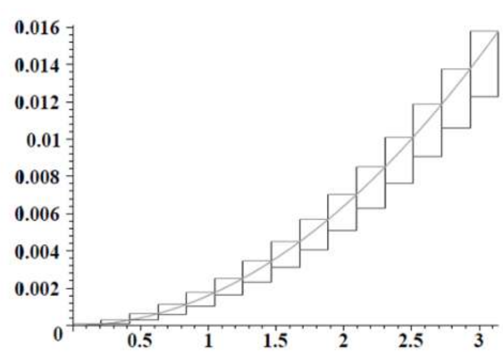

Fig. 8. The estimation interval for the function $x^2/625$ after applying trigonometric splines

IV. Functions of Several Variables and Interval Estimation

Suppose that a function of several variables is specified at grid nodes.

First consider a function of two variables. Suppose, for example, that the function is $S(x, y) = \sin(2xy)\cos(2x + y/2)$ (see Fig. 9). We fix one of the variables and use the proposed interval estimation technique.

Let us put $y = 1/25$ in $S(x, y) = \sin((2xy)\cos(2x + y/2)$. Now we can determine the interval estimation of the function $Q_1(x) = S(x, 1/25)$. The plots of $Q_1(x)$and the result of the interval estimation are given in Fig. 10. The plot of the error of the approximation of the function $Q_1(x) = S(x, 1/25)$ is given in Fig. 11.

Let us put $x = 1/25$ in $S(x, y) = \sin(2xy)\cos(2x + y/2)$. Now we can determine the interval estimation of the function $Q_2(y) = S(1/25, y)$. The plots of $Q_2(y)$ and the result of the interval estimation are given in Fig. 12.

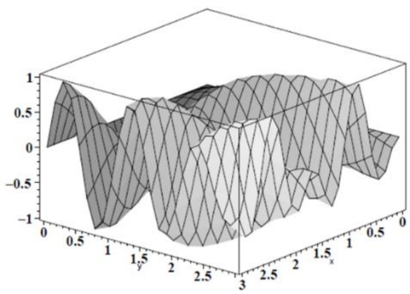

Fig. 9. The plot of the function $S(x, y) = \sin((2xy)\cos(2x + y/2)$

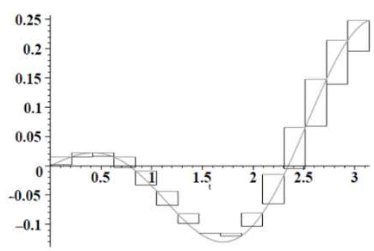

Fig. 10. The plot of the function $Q_1(x) = S(x, 1/25)$and the result of the interval estimation

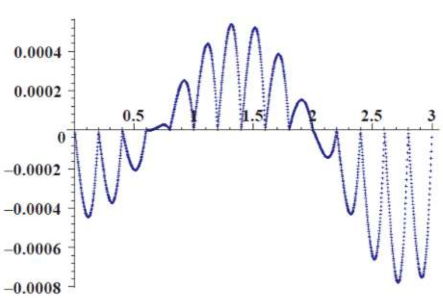

Fig. 11. The plot of the error of the approximation of the function $Q_1(x) = S(x, 1/25)$

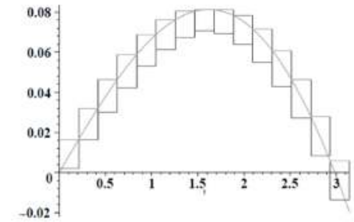

Fig. 12. The plot of the function $Q_2(y) = S(1/25, y)$ and the result of the interval estimation

Let us put $y = 1$ in $S(x, y) = \sin(2xy)\cos(2x + y/2)$. Now we can determine the interval estimation of the function $Q_3(x) = S(x, 1)$. The plots of $Q_3(x)$and the result of the interval estimation are given in Fig. 13. The plot of the error of the approximation of the function $Q_3(x) = S(x, 1)$ is given in Fig. 14.

Let us put $x = 1$ in $S(x, y) = \sin(2xy)\cos(2x + y/2)$. Now we can determine the interval estimation of the function $Q_4(y) = S(1, y)$. The plots of $Q_4(y)$ and the result of the interval estimation are given in Fig. 15.

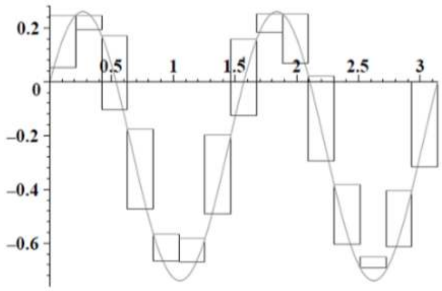

Fig. 13. The plot of the function $Q_3(x) = S(x, 1)$ and the result of the interval estimation

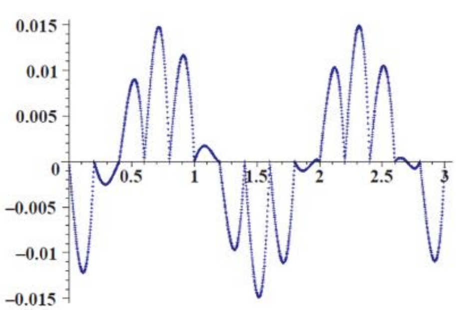

Fig. 14. The plot of the error of the approximation of the function $Q_3(x) = S(x, 1)$

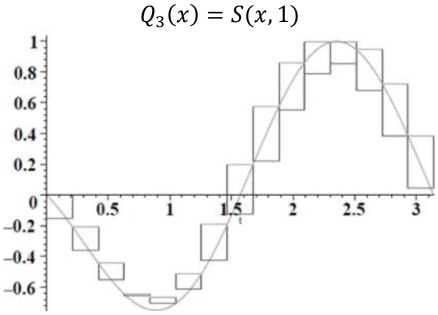

Fig. 15. The plot of the function $Q_4(y) = S(1, y)$ and the result of the interval estimation

Let us put $y = 1/2$ in $S(x, y) = \sin(2xy)\cos(2x + y/2)$. Now we can determine the interval estimation of the function

DOI: 10.46300/9106.2020.14.61

$Q_5(x) = S(x, 1/2)$. The result of the interval estimation are given in Fig. 16.

Let us put $x = 1/2$ in $S(x, y) = \sin(2xy)\cos(2x + y/2)$. Now we can determine the interval estimation of the function $Q_6(y) = S(1/2, y)$. The result of the interval estimation are given in Fig. 17.

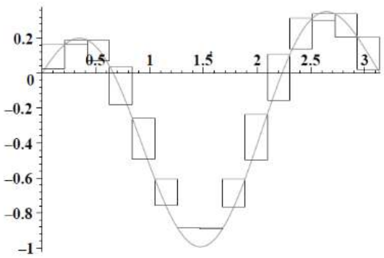

Fig. 16. The plot of the function $Q_5(x) = S(x, 1/2)$. and the result of the interval estimation

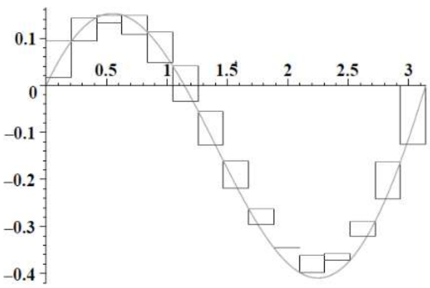

Fig. 17. The plot of the function $Q_6(y) = S(1/2, y)$ and the result of the interval estimation

V. Paralleling Calculations at Approximation with Third-Order Splines

Basic splines of the third order of approximation, convenient for interpolating the functions of one variable, were considered in detail in Sections 2-4.

In this section, we discuss the construction of the interpolation of a function of two variables in a rectangular region on a plane. Consider the approximation of the function of the two variables in domain D. Suppose that two families of parallel lines are constructed in domain D. $x_0 + ih, i = 0, \pm 1, \dots, y_0 + kh_1, k = 0, \pm 1, \dots, h_1, h > 0$.

Let the values of the function of two variables at the grid nodes (the intersection points of these lines) be known. Let us discuss the construction of the approximation of a function in this domain and the parallelization of this process. Applying the direct (tensor) product, we can obtain the formulas for the basis splines of two variables. The formula of the right polynomial basis spline $\Omega^R_{j,k} = \Omega^R_{j,k}(x_j + th, y_k + t_1 h)$, when $supp\,\Omega^R_{j,k} = [x_{j-2}, x_{j+1}] \times [y_{k-2}, y_{k+1}]$, on a uniform grid with step h, has the form, which is different for different position of the small square:

$$\Omega^R_{j,k}(z) = (t-1)(t-2)(t_1-1)(t_1-2)/4, \quad 0 \le t \le 1, \quad 0 \le t_1 \le 1,$$

$$\Omega^R_{j,k}(z) = -(t-1)(t+1)(t_1-1)(t_1-2)/2, \quad -1 \le t \le 0, \quad 0 \le t_1 \le 1,$$

$$\Omega^R_{j,k}(z) = (t+1)(t+2)(t_1-1)(t_1-2)/4, \quad -2 \le t \le -1, \quad 0 \le t_1 \le 1,$$

$$\Omega^R_{j,k}(z) = (t+1)(t+2)(t_1+1)(t_1+2)/4, \quad -2 \le t \le -1, \quad -2 \le t_1 \le -1,$$

$$\Omega^R_{j,k}(z) = (t-1)(t-2)(t_1+1)(t_1+2)/4, \quad 0 \le t \le 1, \quad -2 \le t_1 \le -1,$$

$$\Omega^R_{j,k}(z) = -(t-1)(t+1)(t_1+1)(t_1+2)/2, \quad -1 \le t \le 0, \quad -2 \le t_1 \le -1,$$

$$\Omega^R_{j,k}(z) = (t+1)(t-1)(t_1+1)(t_1-1), \quad -1 \le t \le 0, \quad -1 \le t_1 \le 0,$$

$$\Omega^R_{j,k}(z) = -(t-1)(t-2)(t_1+1)(t_1-1)/2, \quad 0 \le t \le 1, \quad -1 \le t_1 \le 0,$$

$$\Omega^R_{j,k}(z) = -(t+1)(t+2)(t_1+1)(t_1-1)/2, \quad -2 \le t \le -1, \quad -1 \le t_1 \le 0,$$

where $z = (x_j + th,\ y_k + t_1 h)$. The image of this spline is shown in Fig. 18.

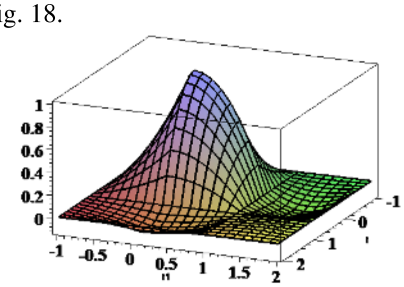

Fig. 18. The plot of the right basis function $\Omega^R_{j,k}(z)$.

The formula of the left polynomial basis spline $\Omega^L_{j,k} = \Omega^L_{j,k}(x_j + th, y_k + t_1 h)$, when $supp\,\Omega^L_{j,k} = [x_{j-1}, x_{j+2}] \times [y_{k-1}, y_{k+2}]$, on a uniform grid with step h along the axes has the form:

$$\Omega^L_{j,k}(z) = (t+1)(t-1)(t_1+1)(t_1-1), \quad 0 \le t \le 1, \quad 0 \le t_1 \le 1,$$

$$\Omega^L_{j,k}(z) = -(t+1)(t+2)(t_1-1)(t_1+1)/2, \quad -1 \le t \le 0, \quad 0 \le t_1 \le 1,$$

$$\Omega^L_{j,k}(z) = -(t+1)(t+2)(t_1-1)(t_1+1)/2, \quad -1 \le t \le 0, \quad 0 \le t_1 \le 1,$$

$$\Omega^L_{j,k}(z) = (t-1)(t-2)(t_1-1)(t_1-2)/4, \quad 1 \le t \le 2, \quad 1 \le t_1 \le 2,$$

$$\Omega^L_{j,k}(z) = -(t-1)(t+1)(t_1-1)(t_1-2)/2, \quad 0 \le t \le 1, \quad 1 \le t_1 \le 2,$$

$$\Omega^L_{j,k}(z) = (t+1)(t+2)(t_1-1)(t_1-2)/4, \quad -1 \le t \le 0, \quad 1 \le t_1 \le 2,$$

$$\Omega^L_{j,k}(z) = (t+1)(t+2)(t_1+1)(t_1+2)/4, \quad -1 \le t \le 0, \quad -1 \le t_1 \le 0,$$

$$\Omega^L_{j,k}(z) = -(t-1)(t+1)(t_1+1)(t_1+2)/2, \quad 0 \le t \le 1, \quad -1 \le t_1 \le 0,$$

$$\Omega^L_{j,k}(z) = (t-1)(t-2)(t_1+1)(t_1+2)/4,$$

ISSN: 1998-4464

$$1 \le t \le 2, \qquad 1 \le t_1 \le 0,$$

where $z = (x_j + th,\ y_k + t_1 h)$. The image of this spline is shown in Fig. 19.

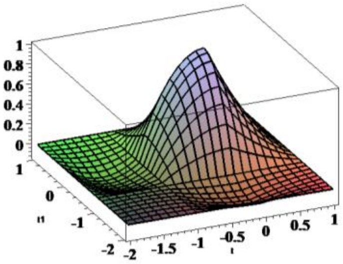

Fig. 19. The plot of the right basis function $\Omega^L_{j,k}$

The basic trigonometric splines of two variables can be similarly constructed. We construct the interpolation of the function $u(x,y)$ separately in each elementary rectangle, with vertices in nodes $(x_j, y_k), (x_{j+1}, y_k), (x_j, y_{k+1}), (x_{j+1}, y_{k+1})$. In the case of using the left basis splines, the approximation has the form:

$$U^L(t,t_1) = u_{j-1,k-1}\omega^L_{j-1}(t)\omega^L_{k-1}(t_1) + u_{j-1,k}\omega^L_{j-1}(t)\omega^L_k(t_1)$$
$$+u_{j-1,k+1}\omega^L_{j-1}(t)\omega^L_{k+1}(t_1) + u_{j,k}\omega^L_j(t)\omega^L_k(t_1)$$
$$+u_{j+1,k}\omega^L_{j+1}(t)\omega^L_k(t_1), +u_{j,k+1}\omega^L_j(t)\omega^L_{k+1}(t_1)$$
$$+u_{j+1,k+1}\omega^L_{j+1}(t)\omega^L_{k+1}(t_1) + u_{j,k-1}\omega^L_j(t)\omega^L_{k-1}(t_1)$$
$$+u_{j+1,k-1}\omega^L_{j+1}(t)\omega^L_{k-1}(t_1), t \in [t_j, t_{j+1}], t_1 \in [t_j, t_{j+1}].$$

The nodes necessary for constructing approximations with the left splines in the lower left rectangle of the region are shown in Fig. 20.

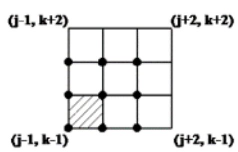

Fig. 20. The nodes that are necessary for constructing the approximation with the left splines $\Omega^L_{j,k}$ in the lower left elementary rectangle

In the case of using the right basis splines, the approximation has the form:

$$U^R(t,t_1) = u_{j+2,k+2}v^R_{j+2}(t)v^R_{k+2}(t_1) + u_{j+2,k}v^R_{j+2}(t)v^R_k(t_1)$$
$$+u_{j+2,k+1}v^R_{j+2}(t)v^R_{k+1}(t_1) + u_{j,k}v^R_j(t)v^R_k(t_1)$$
$$+u_{j+1,k}v^R_{j+1}(t)v^R_k(t_1) + u_{j,k+1}v^R_j(t)v^R_{k+1}(t_1)$$
$$+u_{j+1,k+1}v^R_{j+1}(t)v^R_{k+1}(t_1) + u_{j,k+2}v^R_j(t)v^R_{k+2}(t_1)$$
$$+u_{j+1,k+2}v^R_{j+1}(t)v^R_{k+2}(t_1), \qquad t \in [t_j, t_{j+1}], t_1 \in [t_j, t_{j+1}].$$

Often it is necessary to construct an approximation of a function in large areas. Note that using the locality property, we can construct an approximation simultaneously in several parts of the region. In parallel computing, two schemes are used: “parquet laying” and “Fox's wall”. The "parquet laying" scheme is used where calculations can be carried out, by analogy with parquet laying, independently, starting from any place. The “Fox Wall” scheme differs in that the calculations are carried out in parallel, but sequentially in layers. The next layer cannot be built if there is no previous one. We use the "parquet laying" scheme.

Using the method of geometric parallelism, the construction of the approximation of a function of two variables, if the values of the function are known at the grid nodes on the domain in the plane, can be significantly accelerated. Process data can be arranged in horizontal or vertical stripes (if the domain is rectangular). When distributing this data among the threads in order to construct interpolation, it is also necessary to distribute data in the resulting boundary band. In case of using splines of the third order of approximation, it is necessary to take into account the boundary layer with a width of one grid interval if we use basic splines of only one type (left or right). Thus, when using two-dimensional only left-side or only right-side splines, each thread must additionally distribute the function values in the grid nodes from the boundary strip (Fig. 21, (left)). If we carry out calculations simultaneously (at the same time), using 2 threads and starting from the lower left corner of the rectangular region, using only the left bases splines with two variables, and from the upper right corner (Fig. 21 (right), 22) then we have the acceleration. A feature of this approach is that the nodes in the boundary strip are located only at the vertices of the rectangles located in the strip on the diagonal of the rectangular region (see Fig.22). If using only the right (the left) basis splines from two variables for every thread (process), then additional process data will not be required for each process.

Fig. 21. The distribution of function values in three vertical stripes into three threads (left), the distribution of function values into two triangular regions (two treads) starting from two opposite corners of the domain (right)

To construct a parallel version of the program, we use C and Open MP. It is convenient to use #*pragma omp parallel sections* to parallelize computations. Let us count the number of multiplications and divisions necessary to calculate the approximation of the function of two variables at a point (x,y). It is easy to see that there are about 50 of these operations. To speed up the calculations, each thread should have at least 2000 operations. We constructed a rectangular grid of nodes in the $[0,1] \times [0,1]$ area with a step of $h = 0.01$, at the nodes of which we will calculate the approximate values of the function $f = xy$. We carry out a series of numerical experiments by running the program 10 times, measuring the execution time each time and calculate the average value of the solution time. Acceleration of calculations, that is, the ratio of the running time of a sequential program to the operating time of a parallelized program is 2.91.

Let us count the number of multiplications and divisions necessary to calculate the approximation of the function of two variables at the point (x,y). It is easy to see that there are about

50 of these operations. To speed up the calculations, each thread should have at least 2000 operations. If there are more than 40 grid nodes along each axis then the time of calculations is reduced when we use 2 threads.

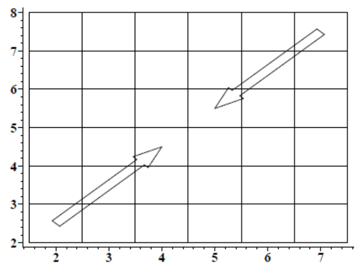

Fig. 22. The two-thread computing parallelization scheme

To calculate the function approach in the lower left corner of the region, we need the function values at the grid nodes, as shown in Fig. 20.

To calculate the function approach in the upper left corner of the region (upper left small rectangle, see 22), the following values of the function in the nodes and the following basic functions are required:

$$\Omega_{j,k}(z) = -(t-1)(t-2)(t_1+1)(t_1+2)/2,\quad 0 \le t \le 1, \quad -1 \le t_1 \le 0,$$

$$\Omega_{j,k}(z) = -(t-1)(t+1)(t_1-1)(t_1-2)/2,\quad -1 \le t \le 0, \quad 1 \le t_1 \le 2,$$

$$\Omega_{j,k}(z) = -(t+2)(t+1)(t_1-1)(t_1-2)/4,\quad -2 \le t \le -1,\ 1 \le t_1 \le 2,$$

$$\Omega_{j,k}(z) = -(t+2)(t+1)(t_1-1)(t_1+1)/2,\quad -2 \le t \le -1, \quad 0 \le t_1 \le 1,$$

$$\Omega_{j,k}(z) = (t-1)(t-2)(t_1-1)(t_1-2)/4,\quad 0 \le t \le 1, \quad 1 \le t_1 \le 2,$$

$$\Omega_{j,k}(z) = (t-1)(t+1)(t_1+1)(t_1-1),$$

$$-1 \le t \le 0, \quad 0 \le t_1 \le 1,$$

$$\Omega_{j,k}(z) = -(t-1)(t+1)(t_1+1)(t_1+2)/2,\quad -1 \le t \le 0, \quad -1 \le t_1 \le 0,$$

$$\Omega_{j,k}(z) = -(t-1)(t-2)(t_1-1)(t_1+1)/2,\quad 0 \le t \le 1, \quad 0 \le t_1 \le 1,$$

$$\Omega_{j,k}(z) = (t+2)(t+1)(t_1+1)(t_1+2)/2,\quad -2 \le t \le -1, -1 \le t_1 \le 0.$$

The image of this basis spline is presented in Fig.23.

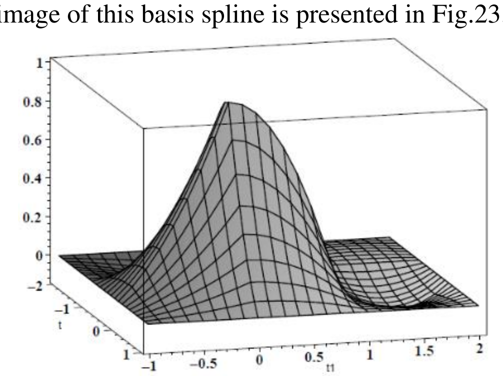

Fig. 23. The plot of the basis function $\Omega_{j,k}$

To calculate the function approach in the upper left corner of the region, the following values of the function in the nodes and the following basic functions are required:

$$U^{RL}(t,t_1) = u_{j,k}v_j^R(t)\omega_k^L(t_1) + u_{j,k+1}v_j^R(t)\omega_{k+1}^L(t_1) + u_{j,k-1}v_j^R(t)\omega_{k-1}^L(t_1) + u_{j+1,k}v_{j+1}^R(t)\omega_k^L(t_1) + u_{j+1,k+1}v_{j+1}^R(t)\omega_{k+1}^L(t_1) + u_{j+1,k-1}v_{j+1}^R(t)\omega_{k-1}^L(t_1) + u_{j+2,k}v_{j+2}^R(t)\omega_k^L(t_1) + u_{j+2,k+1}v_{j+2}^R(t)\omega_{k+1}^L(t_1) + u_{j+2,k-1}v_{j+2}^R(t)\omega_{k-1}^L(t_1), t \in [t_j, t_{j+1}], t_1 \in [t_k, t_{k+1}].$$

Similar formulas can easily be obtained for basis splines and approximations of the function with these basis splines along the main diagonal of the region, starting from the lower left corner of the region (lower left rectangle, see Fig. 22).

VI. Application to the boundary value problem

We apply the approximation with third-order polynomial and non-polynomial splines to the numerical solution of partial differential equations. Let $u = u(x,t)$. We consider the boundary value problem

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial u}{\partial x} + f(x,t)$$

in a rectangular domain $0 \le x \le 1, 0 \le t \le T$ under the boundary conditions: $u|_{x=0} = \varphi(t)$, $u|_{x=1} = \psi(t)$, $u|_{t=0} = u_0$. We construct a grid of nodes $\{(x_j, t_k)\}$, $x_j = jh$, $t_k = k\tau$, $k = 0,1,2 \ldots M, j = 0,1,2, \ldots N$. Let $u_{jk} = u(x_j, t_k)$. We apply the polynomial approximations of partial derivatives of the following type:

$$\frac{\partial u}{\partial t} = \frac{u_{jk+1} - u_{jk}}{\tau} + O(\tau),$$

$$\frac{\partial u}{\partial x} \approx (u_{j-1k+1}g'_{j-1} + u_{jk+1}\, g'_j + u_{j+1k+1}g'_{j+1}),$$

where

$$g'_{j-1} = \frac{x_j - x_{j+1}}{(x_{j-1} - x_j)(x_{j-1} - x_{j+1})},$$

$$g'_j = \frac{1}{(x_j - x_{j-1})} + \frac{1}{(x_j - x_{j+1})},$$

$$g'_{j-1} = \frac{x_j - x_{j-1}}{(x_{j+1} - x_j)(x_{j+1} - x_{j-1})}.$$

On a uniform grid with step h, we obtain $g_j = 0$, $\frac{\partial u}{\partial x} = \left(u_{j-1k+1}g'_{j-1} + u_{j+1k+1}g'_{j+1}\right) + O(h^2)$, where $g'_{j-1} = \frac{-1}{2h}, g'_{j+1} = -g'_{j-1} = \frac{1}{2h}$.

Thus, the use of the cubic polynomial splines on a uniform grid of nodes to construct a formula for numerical differentiation leads to a well-known formula:

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{u_{j-1k+1} - 2u_{jk+1} + u_{j+1k+1}}{h^2}.$$

Thus, we have obtained the well-known difference equation [22], [23]:

$$\frac{u_{jk+1} - u_{jk}}{\tau} = \frac{u_{j-1k+1} - 2u_{jk+1} + u_{j+1k+1}}{h^2}.$$

Now let's see what formula for numerical differentiation can be obtained using the approximation with the trigonometric splines. In the trigonometric case on a non-uniform grid, we can use the formula

$$\frac{\partial u}{\partial x} \approx (u_{j-1k+1}g'_{j-1} + u_{jk+1}\, g'_j + u_{j+1k+1}g'_{j+1}),$$

where

$$g'_{j-1}=\frac{\sin(x_j/2-x_{j+1}/2)}{2\sin(\frac{x_{j-1}}{2}-\frac{x_j}{2})\sin(\frac{x_{j-1}}{2}-\frac{x_{j+1}}{2})},$$
$$g'_{j+1}=\frac{\sin(x_j/2-x_{j-1}/2)}{2\sin(\frac{x_{j-1}}{2}-\frac{x_{j+1}}{2})\sin(\frac{x_j}{2}-\frac{x_{j+1}}{2})},$$
$$g'_j=\frac{\cos(x_j/2-x_{j-1}/2)}{2\sin(\frac{x_j}{2}-\frac{x_{j-1}}{2})}+\frac{\cos(x_j/2-x_{j+1}/2)}{2\sin(\frac{x_j}{2}-\frac{x_{j+1}}{2})}.$$

On a uniform grid with step h, we have $g_j=0$. Thus, we can use the formula:

$$\frac{\partial u}{\partial x}=\left(u_{j-1k+1}g'_{j-1}+u_{j+1k+1}g'_{j+1}\right)+O(h^2),$$

where $g'_{j-1}=\frac{-1}{2\sin(h)}$, $g'_{j+1}=-g'_{j-1}=\frac{1}{2\sin(h)}$.

We also need a formula for the second derivative. Let us construct the formula of the second derivative using approximation $U^M(x)$ of the function $u(x)$ on the grid interval $[x_i,x_{i+1}]$ with the trigonometric spline of the fourth order of approximation (see [11]):

$$U^M(x)=u(x_{i-1})g_{i-1}+u(x_i)g_i+u(x_{i+1})g_{i+1}+u(x_{i+2})g_{i+2},$$

where

$$g_{i-1}=A_{i-1}/B_{i-1},$$
$$A_{i-1}=\sin\left(\frac{x}{2}-\frac{x_i}{2}\right)\sin\left(\frac{x}{2}-\frac{x_{i+1}}{2}\right)\sin\left(\frac{x}{2}-\frac{x_{i+2}}{2}\right),$$
$$B_{i-1}=\sin\left(\frac{x_{i-1}}{2}-\frac{x_i}{2}\right)\sin\left(\frac{x_{i-1}}{2}-\frac{x_{i+1}}{2}\right)\sin\left(\frac{x_{i-1}}{2}-\frac{x_{i+2}}{2}\right),$$
$$g_i=A_i/B_i,$$
$$A_i=\sin\left(\frac{x}{2}-\frac{x_{i-1}}{2}\right)\sin\left(\frac{x}{2}-\frac{x_{i+1}}{2}\right)\sin\left(\frac{x}{2}-\frac{x_{i+2}}{2}\right),$$
$$B_i=\sin\left(\frac{x_i}{2}-\frac{x_{i-1}}{2}\right)\sin\left(\frac{x_i}{2}-\frac{x_{i+1}}{2}\right)\sin\left(\frac{x_i}{2}-\frac{x_{i+2}}{2}\right),$$
$$g_{i+1}=A_{i+1}/B_{i+1},$$
$$A_{i+1}=\sin\left(\frac{x}{2}-\frac{x_{i-1}}{2}\right)\sin\left(\frac{x}{2}-\frac{x_i}{2}\right)\sin\left(\frac{x}{2}-\frac{x_{i+2}}{2}\right),$$
$$B_{i+1}=\sin\left(\frac{x_{i+1}}{2}-\frac{x_{i-1}}{2}\right)\sin\left(\frac{x_{i+1}}{2}-\frac{x_j}{2}\right)\sin\left(\frac{x_{i+1}}{2}-\frac{x_{i+2}}{2}\right),$$
$$g_{i+2}=A_{i+2}/B_{i+2},$$
$$A_{i+2}=\sin\left(\frac{x}{2}-\frac{x_{i-1}}{2}\right)\sin\left(\frac{x}{2}-\frac{x_{i+1}}{2}\right)\sin\left(\frac{x}{2}-\frac{x_i}{2}\right),$$
$$B_{i+2}=\sin\left(\frac{x_{i+2}}{2}-\frac{x_{i-1}}{2}\right)\sin\left(\frac{x_{i+2}}{2}-\frac{x_{i+1}}{2}\right)\sin\left(\frac{x_{i+2}}{2}-\frac{x_i}{2}\right),$$

we receive the formula:

$$\left(U^M(x)\right)''=u(x_{i-1})g''_{i-1}(x)+u(x_i)g''_i(x)+u(x_{i+1})g''_{i+1}(x)+u(x_{i+2})g''_{i+2}(x).$$

When $x_{i-1}=x_i-h,\ x_{i+1}=x_i+h,\ x_{i+2}=x_i+2h,$ it is not difficult to obtain the formula:

$$\left(U^M(x_i)\right)''=u(x_{i-1})g''_{i-1}(x_i)+u(x_i)g''_i(x_i)+u(x_{i+1})g''_{i+1}(x_i)+u(x_{i+2})g''_{i+2}(x_i),$$

where

$$g''_i(x_i)=-\frac{3}{4}-\frac{\cos^2(h/2)}{2\sin^2(h/2)},$$
$$g''_{i-1}(x_i)=\frac{\cos(h/2)}{2\sin\left(\frac{h}{2}\right)\sin\left(\frac{3h}{2}\right)}+\frac{\cos(h)}{2\sin(h)\sin\left(\frac{3h}{2}\right)},$$
$$g''_{i+1}(x_i)=\frac{\cos(h/2)}{2\sin^2\left(\frac{h}{2}\right)}-\frac{\cos(h)}{2\sin(\mathrm{h})\sin\left(\frac{h}{2}\right)}.$$

Example. Let us solve the problem:

$$\frac{\partial u}{\partial t}=\frac{\partial^2 u}{\partial x^2}+f(x,t),\ t\in[0,0.1],x\in[0,1],$$

where $f(x,t)=(2t+0.5)\cos(2x-1)-0.5\cos(1)$, in the domain $[0,1]\times[0,0.1]$, when $u|_{t=0}=0, u|_{x=1}=0$, $u|_{x=0}=0$.

The exact solution of the problem is $u=t\sin(x)\sin(1-x)$. We have constructed the right side of the equation according to the model solution, for debugging the program and calculating the actual errors. Let us construct the grid nodes with a uniform grid of nodes when $N=20, M=30$. Consider the difference equation in the internal nodes of the grid in the polynomial case:

$$\frac{u_{jk+1}-u_{jk}}{\tau}=\frac{u_{j-1k+1}-2u_{jk+1}+u_{j+1k+1}}{h^2}+f\left(x_j,t_{k+1}\right),$$
$$j=1,2,\dots M-1,\qquad k=1,2,\dots N-1,$$
$$u_{j,0}=u_0(jh), j=0,1,\dots M,$$

and

$$u_{0,k}=\varphi(k\tau), k=1,2,\dots N,$$
$$u_{1,k}=\psi(k\tau), k=1,2,\dots N.$$

Along the border of the domain, the values of the solution are known. We carry out calculations from the bottom to the top on the grid layers. As it is known, the implicit scheme in the polynomial case is stable for calculations for any h,τ. Thus, we can use the following implicit scheme using the trigonometric formula for numerical differentiation:

$$u_{j,k+1}=u_{j,k}+\tau\left(u_{j-1,k+1}g''_{j-1}+u_{j+1,k+1}g''_{j+1}+u_{j,k+1}g''_j+f\left(x_j,t_{k+1}\right)\right),$$
$$j=1,2,\dots M-1,\qquad k=1,2,\dots N-1,$$

$$u_{j,0}=u_0(jh), j=0,1,\dots M,$$
$$u_{0,k}=\varphi(k\tau), k=1,2,\dots N,$$
$$u_{1,k}=\psi(k\tau), k=1,2,\dots N.$$

On each layer we need to solve a system of linear algebraic equations. Let us verify that the matrix of this system of equations has a diagonal dominance. We write the system of equations in the form:

$$a_jv_{j-1}+b_jv_j+c_jv_{j+1}=q_j,$$
$$v_0=\varphi((k+1)\tau), v_M=(\psi(k+1)\tau),$$

where $v_j=u_{j,k+1}, q_j=\tau f\left(x_j,t_{k+1}\right)+u_{j,k}$, $b_j=1-\tau g''_j, a_j=-\tau g''_{j-1}, c_j=-\tau g''_{j+1}$.

It is easy to see that for $|\tau|<1$ the inequality holds:

$$|b_j|>|a_j|+|c_j|.$$

Therefore, the system has a unique solution.

Fig.24 shows the error of the solution obtained using the polynomial splines. Fig.25 shows the error of the solution obtained using the trigonometric splines.

Thus, the use of the trigonometric splines gives approximation errors less than in the case of the polynomial splines. The stability will be discussed in the next Section.

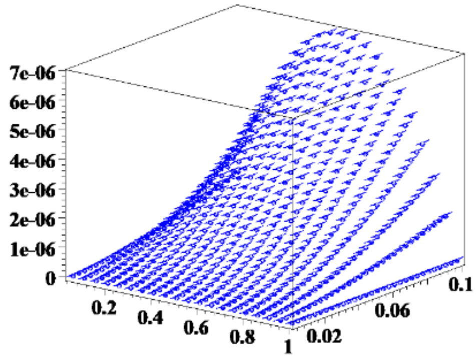

Fig.24.The plot of the errors in absolute value of the solution obtained using polynomial splines

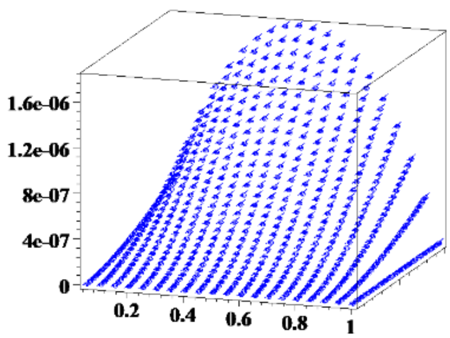

Fig.25.The plot of the errors in absolute value of the solution obtained using trigonometric splines

VII. About Stability

In this section, we discuss aspects related to the convergence of the constructed difference schemes. First of all, we recall some definitions. We have to solve the boundary value problem $Lu = f$ in domain D with the border Γ. Let $D_h = \{M_h\}$ be the set of nodes in $D \cup \Gamma$. Let $u = u(x,t)$ be the solution of the problem. Let the function $u^{(h)}$ be defined only in the set of nodes, so it will be called the mesh function. It is well-known that instead of solving the problem $Lu = f$, we solve the difference scheme $L_h u^{(h)} = f^{(h)}$. Let U_h be the linear normed space with the elements $u^{(h)}$. Let F_h be the linear normed space with the elements $f^{(h)}$. Let $\|\cdot\|_{U_h}$, $\|\cdot\|_{F_h}$ be the norms in the spaces U_h, F_h: $\| u^{(h)} \|_{U_h} = \max_{j,k} |u_{jk}|$,

$\| f^{(h)} \|_{F_h} = \max(\max_j |u_0(jh)|, \max_k |\varphi(k\tau)|, \max_k |\psi(k\tau)|, \max_{j,k} |f(jh,k\tau)|)$.

First, we examine for the stability, the implicit difference scheme:

$$\frac{u_{jk+1}-u_{jk}}{\tau} = u_{j-1,k+1}g''_{j-1} + u_{j+1,k+1}g''_{j+1} + u_{j,k+1}g''_j + f(x_j,t_{k+1}).$$

We first discuss the stability of the initial value problem with respect to the initial data. We will look for a solution to a homogeneous problem (when $f(x_j,t_{k+1}) = 0$) in the form: $u_{jk} = \lambda^k \exp(Ija)$, where I is the imaginary unit, a is real. Now we have the equation:

$$\frac{\lambda - 1}{\tau} = \lambda(g_{j+1}\exp(Ia) + g_{j-1}\exp(-Ia) + g_j).$$

Our aim is to find out for which τ and h the following inequality will satisfy $|\lambda| \le 1 + c\tau$ (von Neumann stability), when $c = const$ does not depend on τ and h. Using the equality $\exp(Ia) - 2 + \exp(-Ia) = -4\sin^2(a/2)$, we get

$$|\lambda| = \left| \frac{1}{1 - \tau g_j + 2\tau\,(2\sin^2\left(\frac{a}{2}\right) - 1)g_{j+1}} \right| \le 1 + c\tau,$$

when $c = 0.0565$. It is not difficult to see that the inequality $|\lambda| \le 1 + c\tau$ holds for any correlation between τ and h. Now we consider the stability of the initial-boundary value problem. Multiply both sides of the difference equation by $-\tau$. We get

$$\tau\left(u_{j-1,k+1}g''_{j-1} + u_{j+1,k+1}g''_{j+1} + u_{j,k+1}g''_j\right) - u_{jk+1} = -\tau f(x_j,t_{k+1}) - u_{jk}.$$

We choose from all the values $u_{j,k+1}$ which in absolute value equals to $|u_{j,k+1}|$ such a value whose index j takes the smallest value $j=j^*$. Let us write the equation corresponding to this value:

$$\tau\left(u_{j^*-1,k+1}g''_{j^*-1} + u_{j^*+1,k+1}g''_{j^*+1} + u_{j^*,k+1}g''_{j^*}\right) - u_{j^*k+1} = -\tau f(x_{j^*},t_{k+1}) - u_{j^*k}.$$

Let $u_{j^*,k+1} > 0$. Consider the right side of the equation

$$\tau\left(u_{j^*-1,k+1}g''_{j^*-1} + u_{j^*+1,k+1}g''_{j^*+1} + u_{j^*,k+1}g''_{j^*}\right) - u_{j^*k+1} = \tau g''_{j^*+1}(u_{j^*+1,k+1} - u_{j^*,k+1}) + \tau g''_{j^*-1}(u_{j^*-1,k+1} - u_{j^*,k+1}) + \tau u_{j^*,k+1}\left(g''_{j^*-1} + g''_{j^*+1} + g''_{j^*}\right) - u_{j^*,k+1} \le -u_{j^*,k+1}.$$

Therefore $-u_{j^*,k+1} \ge -\tau f(x_{j^*},t_{k+1}) - u_{j^*k}$.

Hence, $\max_j |u_{j,k+1}| = u_{j^*,k+1} \le |\tau f(x_{j^*},t_{k+1}) - u_{j^*k}| \le \max_j |u_{j,k+1}| + \tau \max_{j,k} |f(x_j,t_{k+1})|$.

By the definition of stability, the solution of the difference scheme must satisfy the condition $\| u^{(h)} \|_{U_h} \le K \| f^{(h)} \|_{F_h}$ for any $f^{(h)}$. Thus, for any τ and h, the stability condition is satisfied for the difference scheme. Since the difference scheme also approximates the problem, the solution of the difference scheme converges to the solution of the problem.

VIII. Discussion of Results

Section 2 discusses polynomial, trigonometric, and exponential local splines. These splines can be used to approximate functions of one or more variables. In this case, it is necessary to calculate the values of the function at additional nodes between the grid nodes. There can be a lot of these additional points, so it is advisable to parallelize the calculations. If the grid of nodes is rectangular, then it is possible to carry out calculations simultaneously from two opposite vertices of the rectangle. Simultaneous computation on different processors reduces the computation time. This is shown in Section 5. The considered splines are suitable not only for calculating the values of the function at the points between the grid nodes, but also for calculating the first derivative of this function. In particular, we can use these splines to construct formulas for numerical differentiation. Numerical differentiation formulas obtained on the basis of the polynomial, trigonometric, and exponential splines, are considered in the second section on a uniform grid of nodes. These splines approximate the first derivative with the order $O(h^2)$. Using these splines, we can construct a numerical differentiation formula to calculate the second derivative. The constructed formula will give an error $O(h)$. If we need to use the formula for numerical differentiation that approximate the second derivative with the error $O(h^2)$ we need splines with the fourth order of approximation. Thus we need to use cubic polynomial splines or non-polynomial splines which provide the error of the approximation $O(h^4)$. The trigonometric splines with this order of error were used in Section 6. For the convergence of the constructed scheme, not only a good approximation is needed, but also stability as shown in Section 7, the constructed method is stable.

In the proposed method, the same rules should be preserved as in the traditional method. It is necessary not to forget about the fatal error of numerical differentiation and not to select too small the grid step. The next relation should be fulfilled, the rounding error of numbers should be much less than h^2.

IX. Conclusion

In this paper we discuss the approximation with the trigonometric and polynomial splines of the third order. The results of the numerical experiments show that trigonometrical approximation is preferred to polynomial approximation when we approximate a trigonometrical function. To avoid calculation in many points we can use interval extension if we need to know only the upper and the lower boundaries of variation of the function between the nodes. But we have to keep in mind the theorem of approximation. The results of working the program of constructing interval extension are presented. This program uses the trigonometric basis splines. The results of the program that uses the polynomial basis splines are not good, because of the wider interval extension. The parallel calculations during the approximation of the function of two variables in rectangular domain can be done using three threads: beginning from the upper right and lower left corners (using the right or left basis splines) and along the main diagonal (using mix basis splines). In this paper, new computational schemes are constructed to solve a parabolic problem. The schemes are based on approximation of partial derivatives by the trigonometric splines. The examples considered in the paper show that in this case the error in solving the problem turns out to be less than when using the traditional method.

In the future, other numerical schemes will be constructed for solving partial differential equations based on the use of other non-polynomial splines.

References

[1] G.Alefeld, J.Herzberger, "Introduction to Interval Computations," Academic Press, New York, 1983.

[2] [2] G. Alefeld, G. Mayer, "Interval analysis: theory and applications," Journal of Computational and Applied Mathematics, vol. 121, 2000, pp. 421-464.

[3] S. Greenland, "Interval estimation by simulation as an alternative to and extension of confidence intervals," International Journal of Epidemiology, vol. 33, 2004, pp.1389-1397.

[4] P. Eijgenraam, "The solution of initial value problems using interval arithmetic. Formulation and analysis of an algorithm," Thesis, Mathematisch Centrum, Amsterdam, 1981, 185 p.

[5] D.Michel, A.Zidna, "Interval-krawczyk approach for solving nonlinear equations systems in B-spline form," Advances in Intelligent Systems and Computing, vol. 359, 2015, pp. 455-465.

[6] X.Liu, "A framework of verified eigenvalue bounds for self-adjoint differential operators," Applied Mathematics and Computation, vol. 267, 2015, pp. 341-355.

[7] L.Xie, J. Liu, J. Zhang, X.Man, "Evidence-Theory-Based Analysis for Structural-Acoustic Field with Epistemic Uncertainties," InternationalJournal of Computational Methods, vol. 14, no. 2, 2017.

[8] V.Skala, M.Smolik, "A new approach to vector field interpolation, classification and robust critical points detection using radial basis functions, " Advances in Intelligent Systems and Computing, vol. 765, 2019, pp. 109-115.

[9] Yu.K. Dem'yanovich, "General flows and their adaptive decompositions," 11 Wseas Transactions on Mathematics, vol. 17, 2018, pp.28-34.

[10] I.G.Burova, A.A.Vartanova, "Interval Estimation of Polynomial Splines of the Fifth Order, " Proceedings 2017 4th International Conference on Mathematics and Computers in Sciences and in Industry, vol.MCSI 2017, 2018 pp. 293-297.

[11] I.G.Burova, "Construction of trigonometric splines," Vestnik S Petersburg University: Mathematics, vol.37 , no. 2, 2004, pp. 6-11.

[12] 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.

[13] I.G.Burova, E.F.Muzafarova, "Interval estimation using integro-differential splines of the third order of approximation," WSEAS Transactions on Mathematics, vol.18, 2019, pp. 153-160.

[14] I.G.Burova, E.G.Ivanova, V.A.Kostin, A.G.Doronina, "Trigonometric Splines of the Third Order of Approximation and Interval Estimation," WSEAS Transactions on Applied and Theoretical Mechanics, vol. 14, 2019, pp.173-183.

[15] Yu. K. Dem'yanovich, I. D. Miroshnichenko, E. F. Musafarova, "On Splines' Smoothness, WSEAS Transactions on Mathematics," vol. 18, 2019, pp.129-136.

[16] Yuri K. Dem'yanovich, Olga V. Belyakova, Bich T. N. Le, "Generalized Smoothness of the Hermite Type Splines, WSEAS Transactions on Mathematics," vol. 17, 2018, pp.359-368.

[17] F.Ureña, L.Gavete, A.García, J.J.Benito, A.M.Vargas, "Solving second order non-linear hyperbolic PDEs using generalized finite difference method (GFDM)," Journal of Computational and Applied Mathematics, vol.363, 2020, pp.1-21.

[18] T.Kravetc, R. Dalmo, "Finite element application of ERBS extraction," Journal of Computational and Applied Mathematics, 379, 2020, Paper 112947.

DOI: 10.46300/9106.2020.14.61

[19] D.Gunderman, N.Flyer, B.Fornberg, "Transport schemes in spherical geometries using spline-based RBF-FD with polynomials," Journal of Computational Physics, 408, 2020, paper 109256.
[20] A.Deka, D.Datta, "Geometric Size Optimization of Annular Step Fin Array for Heat Transfer by Natural Convection," Modeling and Optimization in Science and Technologies, 16, 2020, pp. 385-402.
[21] Y.Heydarpour, M.Mohammadi-Aghdam, "A New Multistep Technique Based on the Nonuniform Rational Basis Spline Curves for Nonlinear Transient Heat Transfer Analysis of Functionally Graded Truncated Cone," Heat Transfer Engineering, 40 (7), 2019, pp. 588-603.
[22] R.Richtmyer, K.Morton, "Difference methods for initial value problems," N.Y.Interscience-Wiley, 1967.
[23] Randall J. LeVeque, "Finite Difference Methods for Ordinary and Partial Differential Equations. Steady-State and Time-Dependent Problems," University of Washington Seattle, SIAM, Philadelphia, 2007.