Approximation by the Third-Order Splines on Uniform and Non-uniform Grids and Image Processing

I.G. BUROVA
St. Petersburg State University
7/9 Universitetskaya nab.,
St. Petersburg,
RUSSIA

E. F.MUZAFAROVA
Saint Petersburg Lyceum 554
St.Petersburg,
RUSSIA

I. I. NARBUTOVSKIKH
St.Petersburg
RUSSIA

Abstract: - This work is one of a series of papers that is devoted to the further investigation of polynomial splines and trigonometric splines of the third order approximation. Polynomial basis splines are better known and therefore more commonly used. However, the use of trigonometric basis splines often provides a smaller approximation error. In some cases, the use of the trigonometric approximations is preferable to the polynomial approximations. Here we continue to compare these two types of approximation. The Lebesgue functions and constants are discussed for the polynomial splines and the trigonometric splines. The examples of the applications of the splines to image enlargement are given.

Key-Words: Polynomial splines, Trigonometric Splines, Interpolation, Image Processing, Enlarging image, Lebesgue constant

1 Introduction

This paper continues the series of papers on approximation by local polynomial and non-polynomial splines (see [1-3]). This paper focuses on polynomial and trigonometric splines of the third order approximation. To construct the local approximation, we need the values of the function at the grid nodes and the basis splines. It should be noted that Yu.K.Demyanovich devotes a lot of attention to the study of quadratic polynomial splines of the Lagrangian type (see [4]).

In some cases, we may need to apply different types of the local approximation, for example, with the polynomial or trigonometric splines, in different parts of the interval $[a, b]$. In order to find out how to choose nodes when approximating with the right splines on one grid interval and approximating with the left splines on the adjacent grid interval, we will consider Lebesgue constants.

As is known, the stability of the interpolation problem depends on the Lebesgue constant. Lebesgue constant depends only on the grid nodes. Many publications have been devoted to it [5-13]. In paper [15] the results for the Lebesgue constants and the behavior of the Lebesgue functions in view of the optimal interpolation points are given.

Nowadays, splines are quite often used for image processing. Image interpolation plays an important role in converting a low resolution image into a high resolution image. Paper [14] provides a comprehensive study of perdurable image interpolation techniques, such as nearest neighbor, bilinear, Bi-Cubic, cubic spline, and iterative linear interpolation. The usage of a Lagrange polynomial and a piecewise polynomial gives a better fitting curve for the interpolated pixel value algorithms.

In study [15] the image enlargement process is performed by using the Bi-Cubic spline interpolation method, and the result of image try to compare between the original one and picture after enlargement.

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

2 The left and right splines

Suppose a, b are real numbers. For the approximation function $f(x)$ on a finite grid of nodes on interval $[a, b]$ it is desirable to use the values of the function only on this interval. In order for this condition to be fulfilled, we use the approximation with the left or the right splines. The

right splines will be applied near the left end of the finite interval $[a, b]$. The left splines will be applied near the right 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 polynomial or trigonometric splines, in which the support of the bases spline consists of three adjacent intervals. Suppose that $supp\, W_j^R = supp\, v_j^R = [x_{j-2},\ x_{j+1}]$ for the right basis spline and $supp\, w_j^L = supp\, \omega_j^L = [x_{j-1},\ x_{j+2}]$ for the left basis spline. Thus, when approximating a function on a finite interval near the right and left boundaries of the interval $[a, b]$ we will use the approximation $F^L(x), G^L(x),\ x \in [x_j, x_{j+1}]$, with the left or $F^R(x), G^R(x),\ x \in [x_j, x_{j+1}]$, with the right continuous splines:

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

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

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

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

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

Formulas for the left polynomial and trigonometric splines are given in paper [1]. It is shown that the left trigonometric basis splines can be written as follows:

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

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

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

The left polynomial basis splines can be written as follows:

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

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

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

Using the notation $x = x_j + th$, $x_{j+1} = x_j + h$, $x_{j-1} = x_j - h$ we get

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

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

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

In paper [1] it is shown that the right trigonometric basis splines can be written as follows:

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

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

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

In paper [1] it is shown that the right polynomial basis splines can be written as follows:

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

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

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

Approximation Theorem 1 is proved in paper [2].

Theorem 1. Let function $f(x)$ be such that $f \in C^3([a,b])$. 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:
$$\|f - G^L\|_{[x_j,\ x_{j+1}]} \le K_1 h^3 \|f'''\|_{[x_{j-1},\ x_{j+1}]},$$

$$\|f - G^R\|_{[x_j,\ x_{j+1}]} \le K_1 h^3 \|f'''\|_{[x_j,\ x_{j+2}]},$$

$$\|f - F^L\|_{[x_j,\ x_{j+1}]} \le K_2 h^3 \|f''' + f'\|_{[x_{j-1},\ x_{j+1}]},$$

$$\|f - F^R\|_{[x_j, \; x_{j+1}]} \le K_2 h^3 \|f''' + f'\|_{[x_j, \; x_{j+2}]},$$

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

As mentioned earlier, the Lebesgue functions and constants are the valuable numerical instrument for approximation. The Lebesgue functions and constants for quadratic polynomial splines are presented in [3]. We call

$$\lambda_{j,m+1}(x) = \sum_{k=j-l+1}^{j+s} |\omega_k(x)|, \; x \in [x_j, \; x_{j+1}]$$

the Lebesgue function of the basic splines $\omega_j(x)$ with respect to the interval $[x_j, \; x_{j+1}]$. For the left splines we have $l = 2, s = 1, m = l + s$. For the right splines we have $l = 1, s = 2, m = l + s$.

We call

$$\lambda_{j,m+1} = \max_{x \in [x_j, \; x_{j+1}]} \lambda_{j,m+1}(x)$$

the Lebesgue constant of the basic splines $\omega_j(x)$ with respect to the interval $[x_j, \; x_{j+1}]$.

Thus, for the right polynomial splines, the Lebesgue function is the following:

$$\lambda_{j,3}^R(x) = |v_j^R(x)| + |v_{j+1}^R(x)| + |v_{j+2}^R(x)|.$$

Let $E(f)$ denote the best approximation of the function $f(x)$ on various polynomials of the second degree on the interval $[x_j, x_{j+2}]$. Let $P(x)$ denote the polynomial of the best approximation of the second degree for the function $f(x)$ on the interval $[x_j, x_{j+2}]$.

Lemma 1. Let $f \in C ([x_j, x_{j+2}])$. For the error of the approximation with the right polynomial splines of the function $f(x)$ the following inequality is valid:

$$\| f - G^R \|_{[x_j, x_{j+1}]} \le (1 + \lambda_{j,3}^R)E(f).$$

Proof. We have for $x \in [x_j, x_{j+1}]$.

$$|f - G^R| \le |f - P| + |P - G^R| \le E(f) +$$

$$\| \sum_{i=j}^{j+2} (P(x_i) - f(x_i)) v_i^R(x) \| \le (1 + \lambda_{j,3}^R)E(f).$$

The proof is complete.

Remark 1. Let $f \in C ([x_{j-1}, x_{j+1}])$. The similar statement can be proved for the approximation with the left polynomial splines:

$$\| f - G^L \|_{[x_j, x_{j+1}]} \le (1 + \lambda_{j,3}^L)E(f).$$

Let us consider the behavior of the Lebesgue functions and constants in approximation of a function on a non-uniform grid. We denote

$$k = \frac{x_{j+2} - x_{j+1}}{x_{j+1} - x_j}.$$

Using the notation $h = x_{j+1} - x_j, \; x = x_j + th$, $x_{j+1} = x_j + h, \; x_{j+2} = x_j + h + kh$ we get

$$\lambda_{j,3}^R(x_j + th) = |(-t + 1 + k)(t - 1)|k$$
$$+ |t(-t + 1 + k)|k$$
$$+ |t(-t + 1 + k)|$$
$$+ |t(t - 1)|)/((k + 1)/k).$$

The maximum of this expression is achieved when $t = 1/2$ and it has the following form:

$$\lambda_{j,3}^R = \frac{2k + 2k^2 + 1}{2k(k + 1)}.$$

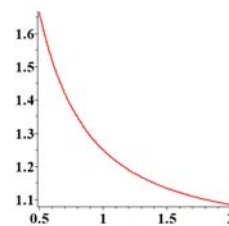The plot of the behavior of the Lebesgue constant $\lambda_{j,3}^R$ on the interval [0.5, 2] is shown in Fig.1.

Fig.1. The behavior of the Lebesgue constant $\lambda_{j,3}^R$ on the interval [0.5, 2].

For the left polynomial splines, the Lebesgue function is the following:

$$\lambda_{j,3}^L(x) = (|(tk + 1)(t - 1)|k + |(tk + 1)(t - 1)|$$
$$+ |(tk + 1)t| + k^2|t(t - 1)|)/(k + 1).$$

The maximum of this expression is achieved when $t = 1/2$ and it has the following form:

$$\lambda_{j,3}^L = \frac{k^2 + 2k + 2}{2(k + 1)}.$$

The plot of the behavior of the Lebesgue constant $\lambda_{j,3}^L$ on the interval [0.5, 2] is shown in Fig.2.
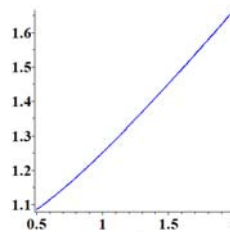

Fig. 2. The behavior of the Lebesgue constant $\lambda_{j,3}^L$ on the interval [0.5, 2]

Let $E(f)$ denote the best approximation of the function $f(x)$ on various trigonometric polynomials of the second degree on the interval $[x_{j-1}, x_{j+1}]$. Let $T(x)$ denote the trigonometric polynomial of the best approximation of the second degree for the function $f(x)$ on the interval $[x_{j-1}, x_{j+1}]$.

Lemma 2. Let $f \in C([x_{j-1}, x_{j+1}])$. For the error of the approximation with the left trigonometric splines of the function $f(x)$ the following inequality is valid:

$$\| f - F^L \|_{[x_j, x_{j+1}]} \le (1 + \lambda_{j,3}^L) E(f).$$

The proof is similar to Lemma 1.

For the left trigonometric splines after using the notation $x = x_j + th$, $x_{j+1} = x_j + h$, $x_{j-1} = x_j - h/k$ we get:

$$\lambda_{j,3}^L(x) = t_1^L + t_2^L + t_3^L,$$

where

$$t_1^L = - \frac{\sin\left(\frac{h(tk+1)}{2k}\right) \sin\left(\frac{th}{2} - \frac{h}{2}\right)}{\sin\left(\frac{h}{2k}\right) \sin\left(\frac{h}{2}\right)},$$

$$t_2^L = \frac{\sin\left(\frac{h(tk+1)}{2k}\right) \sin\left(\frac{th}{2}\right)}{\sin\left(\frac{h(k+1)}{2k}\right) \sin\left(\frac{h}{2}\right)},$$

$$t_3^L = \frac{\sin\left(\frac{th}{2}\right) \sin\left(\frac{th}{2} - \frac{h}{2}\right)}{\sin\left(\frac{h(k+1)}{2k}\right) \sin\left(\frac{h}{2k}\right)}.$$

The maximum of this expression is achieved when $t = 1/2$ and it has the following form:

$$\lambda_{j,3}^L = \frac{4 \sin\left(\frac{h}{2}\right) \sin\left(\frac{h(k+1)}{k}\right) + \sin\left(\frac{h}{k}\right) - \sin(h)}{\sin(h) + \sin\left(\frac{h}{k}\right) - \sin\left(\frac{h(k+1)}{k}\right)}.$$

Let $E(f)$ denote the best approximation of the function $f(x)$ on various trigonometric polynomials of the second degree on the interval $[x_j, x_{j+2}]$. Let $T(x)$ denote the trigonometric polynomial of the best approximation of the second degree for the function $f(x)$ on the interval $[x_j, x_{j+2}]$.

Lemma 3. Let $f \in C([x_j, x_{j+2}])$. For the error of the approximation with the right trigonometric splines of the function $f(x)$ the following inequality is valid:

$$\| f - F^R \|_{[x_j, x_{j+1}]} \le (1 + \lambda_{j,3}^R) E(f).$$

The proof is similar to Lemma 1.

For the right trigonometric splines after using the notation $x = x_j + th$, $x_{j+1} = x_j + h$, $x_{j+2} = x_j + h + hk$ we get:

$$\lambda_{j,3}^R(x) = t_{s1}^R + t_{s2}^R + t_{s3}^R,$$

$$t_{s1}^R = \frac{-\sin\left(-\frac{th}{2} + \frac{h}{2} + \frac{hk}{2}\right) \sin(\frac{th}{2} - \frac{h}{2})}{\sin\left(\frac{h}{2} + \frac{hk}{2}\right) \sin\left(\frac{h}{2}\right)},$$

$$t_{s2}^R = \frac{\sin\left(\frac{th}{2}\right) \sin(-\frac{th}{2} + \frac{h}{2} + \frac{hk}{2})}{\sin\left(\frac{h}{2}\right) \sin(\frac{hk}{2})},$$

$$t_{s3}^R = \frac{\sin\left(\frac{th}{2}\right) \sin\left(\frac{th}{2} - \frac{h}{2}\right)}{\sin\left(\frac{h}{2} + \frac{hk}{2}\right) \sin(\frac{hk}{2})}.$$

The maximum of this expression is achieved when $t = 1/2$ and it has the following form:

$$\lambda_{j,3}^R = \sin\left(\frac{h}{4}\right) \left(\sin\left(\frac{h}{4} + \frac{hk}{2}\right) \sin\left(\frac{hk}{2}\right)\right.$$
$$+ \sin\left(\frac{h}{4} + \frac{hk}{2}\right) \sin\left(\frac{h}{2} + \frac{kh}{2}\right)$$
$$+ \sin\left(\frac{h}{4}\right) \sin\left(\frac{h}{2}\right))$$
$$\left./\left(\sin\left(\frac{h}{2} + \frac{hk}{2}\right) \sin\left(\frac{h}{2}\right) \sin\left(\frac{hk}{2}\right)\right).\right.$$

3 Approximation on non-uniform grids and the Lebesgue constants

Let us consider the approximation by changing one of the types of splines to another type. We get that $k = 1$ by solving a nonlinear equation $\lambda_{j,3}^R = \lambda_{j,3}^L$ for k considering h as a parameter. This means we can use a uniform grid when approximating the function and changing the left splines to the right splines on the interval $[a, b]$. The Lebesgue constant is preserved in the approximation inequalities. The plot of the solution of the nonlinear equation is presented in Fig.3. This means that when approximating the function on the interval $[a, b]$ and moving from left to right splines, we can use the uniform grid.
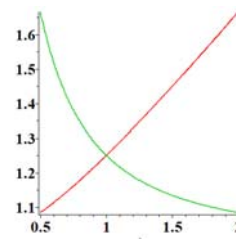

Fig. 3. The plot of the solution of the non-linear equation $\lambda_{j,3}^R = \lambda_{j,3}^L$.

We get the solution $k = 1$ of the nonlinear equation $\lambda_{j,3}^R = \lambda_{j,3}^L$ for the Lebesgue constant of trigonometric splines.

Now we consider the approximation with fifth-order splines and the Lebesgue constants. We construct the approximation with the right basis splines in the form:

$$V_4^R(x) = \sum_{i=j}^{j+4} f(x_i) w_i^R(x), \ x \in [x_j, x_{j+1}],$$
$$j = 0, \dots, n - 5,$$

where

$w_j^R(x)$
$$= \frac{(x - x_{j+1})(x - x_{j+2})(x - x_{j+3})(x - x_{j+4})}{(x_j - x_{j+1})(x_j - x_{j+2})(x_j - x_{j+3})(x_j - x_{j+4})}$$

$w_{j+1}^R(x)$
$$= \frac{(x - x_j)(x - x_{j+2})(x - x_{j+3})(x - x_{j+4})}{(x_{j+1} - x_j)(x_{j+1} - x_{j+2})(x_{j+1} - x_{j+3})(x_{j+1} - x_{j+4})},$$

$w_{j+2}^R(x)$
$$= \frac{(x - x_j)(x - x_{j+1})(x - x_{j+3})(x - x_{j+4})}{(x_{j+2} - x_j)(x_{j+2} - x_{j+1})(x_{j+2} - x_{j+3})(x_{j+2} - x_{j+4})},$$

$w_{j+3}^R(x)$
$$= \frac{(x - x_j)(x - x_{j+1})(x - x_{j+2})(x - x_{j+4})}{(x_{j+3} - x_j)(x_{j+3} - x_{j+1})(x_{j+3} - x_{j+2})(x_{j+3} - x_{j+4})},$$

$w_{j+4}^R(x)$
$$= \frac{(x - x_j)(x - x_{j+1})(x - x_{j+2})(x - x_{j+3})}{(x_{j+4} - x_j)(x_{j+4} - x_{j+1})(x_{j+4} - x_{j+2})(x_{j+4} - x_{j+3})}.$$

We construct the approximation with the left basis splines in the form:

$$V_4^L(x) = \sum_{i=j-3}^{j+1} f(x_i) w_i^L(x), \ x \in [x_j, x_{j+1}],$$
$$j = 3, \dots, n - 1,$$

where

$w_{j-3}^L(x)$
$$= \frac{(x - x_{j-2})(x - x_{j-1})(x - x_j)(x - x_{j+1})}{(x_{j-3} - x_{j-2})(x_{j-3} - x_{j-1})(x_{j-3} - x_j)(x_{j-3} - x_{j+1})},$$

$w_{j-2}^L(x)$
$$= \frac{(x - x_{j-3})(x - x_{j-1})(x - x_j)(x - x_{j+1})}{(x_{j-2} - x_{j-3})(x_{j-2} - x_{j-1})(x_{j-2} - x_j)(x_{j-2} - x_{j+1})},$$

$w_{j-1}^L(x)$
$$= \frac{(x - x_{j-3})(x - x_{j-2})(x - x_j)(x - x_{j+1})}{(x_{j-1} - x_{j-3})(x_{j-1} - x_{j-2})(x_{j-1} - x_j)(x_{j-1} - x_{j+1})},$$

$w_j^L(x)$
$$= \frac{(x - x_{j-3})(x - x_{j-2})(x - x_{j-1})(x - x_{j+1})}{(x_j - x_{j-3})(x_j - x_{j-2})(x_j - x_{j-1})(x_j - x_{j+1})},$$

$w_{j+1}^L(x)$
$$= \frac{(x - x_{j-3})(x - x_{j-2})(x - x_{j-1})(x - x_j)}{(x_{j+1} - x_{j-3})(x_{j+1} - x_{j-2})(x_{j+1} - x_{j-1})(x_{j+1} - x_j)}$$

Theorem 2. The error of the approximation on $[x_j, x_{j+1}]$ with the left polynomial splines with the fifth-order approximation is the following:

$$\left\| V_4^R - f \right\| \leq K \left\| f^{(5)} \right\|_{[x_j, x_{j+4}]} h^5 / 5!, K = 3.632.$$

The error of the approximation with the right polynomial splines with the fifth-order approximation is the following:

$$\left\| V_4^L - f \right\| \leq K \, h^5 / 5! \, \left\| f^{(5)} \right\|_{[x_{j-3}, x_{j+1}]}, K = 3.632.$$

Proof. The proof of the theorem is based on the application of the Lagrangian interpolation remainder formula. For example, we can write

$$V_4^L(x) - f(x) = \frac{f^{(5)}(s)}{5!} V(x),$$
$$\text{where } s \in [x_{j-3}, x_{j+1}],$$
$$V(x) = (x - x_{j-3})(x - x_{j-2})(x - x_{j-1})(x - x_j)(x - x_{j+1}).$$

After calculating the maximum of $|V(x)|, x \in [x_j, x_{j+1}]$, we find the required statement.
The proof is complete.

Similarly to what was considered for third-order splines of approximation, one can obtain the Lebesgue functions $\lambda_{j,5}^R(x)$, $\lambda_{j,5}^L(x)$ and the Lebesgue constants $\lambda_{j,5}^R$, $\lambda_{j,5}^L$ for the fifth-order splines of approximation. Since the expressions turn out to be very large, we show only the graphs of the Lebesgue constants. The plot of the behavior of the Lebesgue constant $\lambda_{j,5}^L$ on the interval [0.1, 3] is shown in Fig.4. The plot of the behavior of the Lebesgue constant $\lambda_{j,5}^R$ on the interval [0.5, 1.2] is shown in Fig.5.
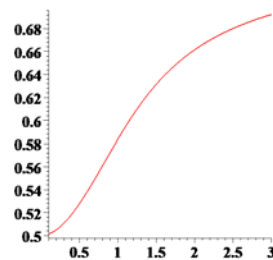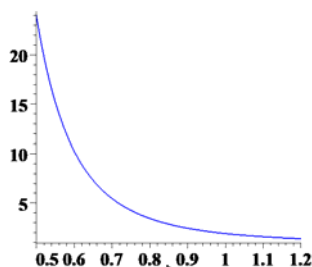

Fig.4. The Lebesgue constant $\lambda_{j,5}^L$

Fig.5. The Lebesgue constant $\lambda_{j,5}^R$

We get $k = 0.96$ by solving a nonlinear equation $\lambda_{j,5}^R = \lambda_{j,5}^L$ for k considering h as a parameter. It follows that when switching from approximation by the right fifth-order splines to approximation by the left fifth-order splines, the same step of a uniform grid can be used. The plot of the behavior of the Lebesgue constants $\lambda_{j,5}^R$ and $\lambda_{j,5}^L$ on the interval $[0.5, 1.2]$ is shown in Fig.6.
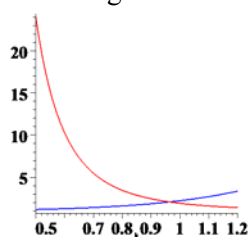

Fig.6.The Lebesgue constants $\lambda_{j,5}^R$ and $\lambda_{j,5}^L$

4 Numerical algorithm of compression the flows

Using the approximation error estimates, it is not difficult to suggest an algorithm for compressing and recovering numerical flows with the preservation of the approximation order. Papers [16,17] show the result of applying the considered splines to enlarge images. Here we consider, in more detail, the differences in approximation by different types of splines. The proposed local splines of the third and fifth orders of approximation can be used to enlarge images when moving from monitors with lower resolution to monitors with higher resolution. One use of the term display resolution applies to fixed-pixel-array displays such as plasma display panels (PDP), liquid-crystal displays (LCD), Digital Light Processing (DLP) projectors, OLED displays, and similar technologies, and is simply the physical number of columns and rows of pixels creating the display (e.g. 1920×1080) (see https://en.wikipedia.org/wiki/Display_resolution). As it is known, the RGB color system (see, for example, https://www.rapidtables.com/web/color/RGB_Color.html), constructs all the colors from the combination of the Red, Green and Blue colors. The red, green and blue use 8 bits each, which have integer values from 0 to 255. Each pixel in the LED

(light emitting diodes) monitor displays colours by combination of red, green and blue LEDs. The tasks of increasing the resolution of images, as well as the compression and restoration of images can be solved using the proposed local splines. Let there be an original image of size [m × n] pixels. We need to implement a solution that would allow us to obtain a new size [k m × k n] at the output from the original image, where k is the magnification factor. We can make the following comparison of points in the source and enlarged images (see. Fig. 7).
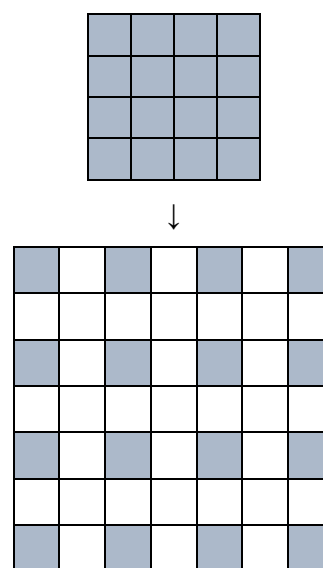

Fig.7.

Next, we perform the calculations in rows that contain color values from the original image. Calculations are carried out for color values at intermediate points. Since the color in computer graphics is a combination of the components of several basic colors, the calculations for each of them are made separately. For calculations on the left side of the line, we can use the left polynomial splines, for the right side we can use the right polynomial splines. Next, we perform the calculations in columns that contain color values from the original image.

4.1 Image Size Increasing Algorithm

Consider the case of increasing the image twice (in width and height) from 6x6 to 12x12. First, copy the source data to a sparse grid of nodes. In the case of increasing the size of the image exactly twice, the problem arises of how to compare the original data with the new image. Let each pixel of the original image has coordinates $(i, j), i \in 1..M, j \in 1..N$, where M is the width, N is the height. We associate

it with a pixel $(2i - 1, \ 2j - 1)$ from the new image. The line to which the information about the colours is added, will then look like the line given in Fig.8:

Fig.8.

and then we will not be able to calculate the colour value in the last pixel. There are two solutions to this problem:

1. If a, b are the width and the height of the original image, x is an integer indicating how many times we want to enlarge the image, then we will take the dimensions of the new image equal: *width* $= (a - 1) \ x + 1$, *height*$= (b - 1)x + 1$. So the line we considered earlier will look like the line given in Fig.9.

Fig.9

With the help of the spline approximation we can calculate the values at intermediate points.

2. Increase the size by x times, but match the pixels as follows in Fig.7 (an example of increasing the size by 2 times). Let the interval [a, b] be taken as [0, 4] (see Fig. 10):

Fig.10.

We consider the set of ordered nodes $\{t_j\}$:

$$t_j = j, j = 0, 1, 2, 3, 4; h \ = \ t_{j+1} - t_j = 1.$$

We calculate at the point $t \in \left[t_j, t_{j+1}\right]$:

$$t = \frac{(t_j + t_{j+1})}{2} = t_j + \ 0.5.$$

Thus, the number of calculations can be significantly reduced. For example, for left polynomial splines of the third order of approximation in this case, it suffices to calculate:

$$w_j(t) = \frac{t - t_{j+1}}{t_j - t_{j+1}} \cdot \frac{t - t_{j+2}}{t_j - t_{j+2}} = 0.375,$$

$$w_{j+1}(t) = \frac{0.5}{1} \cdot \frac{-1.5}{-1} = 0.75,$$

$$w_{j+2}(t) = \frac{0.5}{2} \cdot \frac{-0.5}{1} = -0.125.$$

Next, for the left side, we look for an approximation with the right splines (yellow), for the right side

with the left splines (red colour) as it is shown in Fig.10. As a result of these calculations, the intermediate result will look as follows (see Fig. 11):
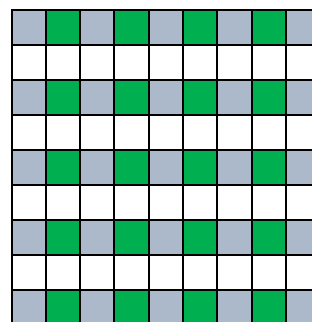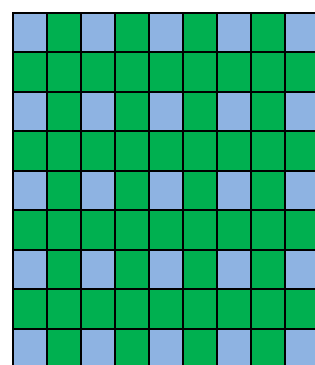

Fig.11.

Fig.12.

After increasing the image in rows, we get the result shown in Fig. 12. Then we repeat the procedure for the columns: for the upper part we are looking for the approximation with the right splines, and for the lower part we are looking for the approximation with the left splines. Thus, we calculate all the colour values in the enlarged image (calculations are carried out for each colour component: red, green, blue and α-channel).

Fig.13. The original image of nature (150x150 pixels)

The results of the enlarging of the images using the splines of the third order of approximation are presented in Figs. 13-15. Fig.13 shows the original image of 150x150 pixels. Fig.14 shows the enlarged

image of 300x300 pixels with polynomial splines. Fig.15 shows the enlarged image of 300x300 pixels with trigonometrical splines.

Fig.14.The enlarged image of nature with polynomial splines (300x300 pixels)

Fig.15. The enlarged image of nature with trigonometrical splines (300x300 pixels)

4.2 Image compression and restoration

A series of experiments on image compression and its subsequent reconstruction using splines were carried out. Image compression was performed by crossing out every even row and column. Fig. 16 shows a comparison of the original and the recovered images. We can see recovered images when the following were used (from left to right):
1) polynomial splines with the third order approximation;

2) trigonometric splines with the third order approximation;
3) polynomial splines with the fifth order approximation;
5) original image.

When we used the polynomial or trigonometric splines of the third order then the quality of the recovered images were almost the same. But the images obtained using the third-order trigonometric splines appeared smoother.

Fig.16.

In the case of polynomial splines of the fifth order, a slightly larger image smoothness is noticeable.

4.3 Estimated Work Time

Java was used to construct a program that enlarges images. For an estimation of time calculations 10 images in high resolution were used. The following sizes have been selected: 3724x5586, 4912x7360, 3456x5184, 6720x4480, 3170x3963, 2730x4096, 4000x6000, 5472x3648, 4000x6000, 3089x2048. The polynomial splines with the fifth order were used for an estimation of time calculations.

Testing was conducted twice. Both times the original image was enlarged 2, 3 and 4 times. The minimum size of the tested image is 6326272 pixels, the maximum size is 36152320. Thus, the number of intermediate pixels for which the calculations were carried out ranges from 18 million (in the case of the minimum image and a 2-fold increase) to 542 million (in the case of the maximum image and a 4-fold increase). Image processing time at the first and second program start is indicated in Table 1.

Table 1

	Test 1	Test 2
image magnification twice	385.49 sec	386.35 sec
image magnification three times	361.28 sec	364.66 sec
4 times magnification	625.85 sec	614.27 sec

Judging by the test results, in this implementation, read-write operations have a significant impact on the image processing time.

5 Conclusion

Our experiments find that local quadratic polynomial splines and the local trigonometric splines give good results for the approximation of the functions. Sometimes, the approximation by the trigonometric splines is preferable to the approximation by the polynomial splines, because it can give the smaller error of approximation. Using splines with the fifth order of approximation we can reduce the number of grid nodes. In this case, the approximation error will not exceed the previous one. When zooming in, both trigonometric and polynomial splines are suitable. There are no significant differences in the magnification of the image, either when using polynomial or when using trigonometric splines.

References:

[1] I.G.Burova, Construction of Trigonometric Splines, *Vestnik St. Petersburg University: Mathematics*, Vol.37, No. 2, 2004, pp. 6-11.

[2] 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.

[3] I.G.Burova, On the Approximation by Quadratic and Cubic Minimal Splines, *Numerical Methods*, Issue 20, 2003, pp. 5-24. (in Russian).

[4] Yu.K. Dem'yanovich, General Flows and their Adaptive Decompositions, *Wseas Transactions on Mathematics*, Vol. 17, 2018, pp.28-34.

[5] V.T. Shevaldin, On Integral Lebesgue Constants of Local Splines with Uniform Knots, *Proceedings of the Steklov Institute of Mathematics,* Vol. 305, 2019, pp. 158-165.

[6] D. Moxey, S.P. Sastry, R.M. Kirby, Interpolation Error Bounds for Curvilinear Finite Elements and Their Implications on Adaptive Mesh Refinement, *Journal of Scientific Computing*, Vol. 78, No. 2, 2019, pp. 1045-1062.

[7] S.I. Novikov, Lebesgue Constants for Some Interpolating L-Splines, *Proceedings of the Steklov Institute of Mathematics*, Vol. 300, 2019, pp. 136-144.

[8] H. Majidian, Automatic Computing of Oscillatory Integrals, *Numerical Algorithms,* Vol. 77, No. 3, 2019, pp. 867-884.

[9] X. Liu, Y. Chen, Convergence Analysis for the Chebyshev Collocation Methods to Volterra Integral Equations with a Weakly Singular Kernel, *Advances in Applied Mathematics and Mechanics*, Vol. 9, No. 6, 2017, pp. 1506-1524.

[10] V.T. Shevaldin, O.Y. Shevaldin, Lebesgue Constant of Local Cubic Splines with Equally Spaced Nodes, *Numerical Analysis and Applications*, Vol. 10, No. 4, pp. 362-367.

[11] V.T. Shevaldin, O.Y. Shevaldin, Upper bounds for Uniform Lebesgue Constants of Interpolation Periodic Sourcewise Representable Splines, *Proceedings of the Steklov Institute of Mathematics*, Vol. 297, 2017, pp. 175-181.

[12] V.T. Shevaldin, E.V. Strelkova, On uniform Lebesgue Constants of Local Exponential Splines with Equidistant Knots, *Proceedings of the Steklov Institute of Mathematics*, Vol. 296, 2017, pp. 206-217.

[13] Bayram Ali Ibrahimoglu, Lebesgue Functions and Lebesgue Constants in Polynomial Interpolation, *Journal of Inequalities and Applications,* Vol. 2016, 2016.

[14] V.Diana Earshia, M.Sumathi, A Comprehensive Study of 1D and 2D Image Interpolation Techniques, *Lecture Notes in Electrical Engineering*, Vol. 500, 2019, pp. 383-391.

[15] D.Abdullah, F.Fajriana, M.Maryana, L.Rosnita, A.P. Utama Siahaan, R.Rahim, P.Harliana, H.Harmayani, Z.Ginting, Erliana, D.Irwansyah, Z.Zulmiardi, M.Khaddafi, F.Milanie, H.Aspan, I.Huda, K.Saddhono, I.Mulyaningsih, R.M.Moonti, P.Parwito, H.Djanggih, A.Amalia, E.Winarno, W.Hadikurniawati, Application of Interpolation Image by using Bi-Cubic Algorithm, *Journal of Physics: Conference Series*, Vol. 1114, No.1, 2018.

[16] I.G.Burova,I.I.Narbutovskikh, E.F.Muzafarova, Image processing and the spline approximation of the third and fifth order**,** *International Journal of Circuits, Systems and Signal Processing,* Vol. 13, 2019, pp. 550-557.

[17] I.G.Burova,E.F.Muzafarova, I.I.Narbutovskikh, Local splines of the second and third order, complex-valued splines and image processing, *International Journal of Circuits, Systems and Signal Processing*, Vol.13, 2019, pp. 419-429．