ISSN 2658-5014 (Print)
2023. Т. 5, № 1 (10). С. 67–75
СИИТ
СИСТЕМНАЯ ИНЖЕНЕРИЯ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ
ISSN 2686-7044 (Online)
http://siit.ugatu.su

UDC 519.16
DOI 10.54708/2658-5014-SIIT-2023-no1-p67

NUMERICAL INVESTIGATION OF THE DIFFERENT INTERPOLATION METHODS

N. M. SHERYKHALINA, A. A. SOKOLOVA, E. R. SHAYMARDANOVA

Abstract. The article is devoted to the approximate solution of some problems of computational mathematics. Practical interpolation for problems solving is discussed. The various interpolation methods as the Lagrange polynomial, the Newton polynomial and the cubic spline are under consideration and the detailed comparative analysis of these methods is carried out. The described methods are applied for restoration of functional dependences on the example of an elementary function. A numerical experiment is demonstrated to restore the functional dependence by the described methods. A mathematical model of the interpolation error is presented. With the help of two polynomials constructed for two different meshes, a polynomial of higher degree is obtained. It is shown that according to the recurrent Aitken ratio the obtained polynomial of a higher degree can be used to estimate the error of the results of the calculated values. The computation results and the error estimations are obtained with great accuracy. The results of the various interpolation methods are compared.

Keywords: interpolation, numerical method, Lagrange polynomial, Newton polynomial, spline, error model.

INTRODUCTION

We have known interpolation since ancient times. It was used by Babylonian and ancient Greek astronomers and mathematicians. A description of linear interpolation can be found in an ancient Chinese mathematical text called The Nine Chapters of the Art of Mathematics, dating from 200 BC to 100 AD. The further development of interpolation is due to such outstanding mathematicians as Newton, Leibniz and Gregory. And their works are still used by mathematicians all over the world.

Today we have a large selection of different interpolation methods. For specific tasks, we can choose the solution that suits best. With the advent of computer, the numerical methods are often applied for solving interpolation problems. Specialists of various professions need to make a large number of calculations with the least error. The term interpolation means the search for intermediate values of a quantity based on some of its known values. In scientific and engineering calculations it is necessary to operate with sets of values obtained by experiment or by random sampling quite often. On the base of these sets we need to construct a function that could receive other obtained values with high accuracy. This is called the curve fitting. Interpolation is a kind of approximation in which the curve of the constructed function passes exactly through the available points.

But interpolation problem is incorrect one. The problem of computing results reliability is under consideration of different authors [1–6].

THE LAGRANGE AND NEWTON POLYNOMIALS

Description of the methods and formulation

Let some function $f(x)$ be given by its values $y_j = f(x_j)$ at a discrete set of points x_j, $j = 0, \ldots, m$. It is required to approximately determine the analytical form of this function and, thus, to be able to calculate its values at intermediate points $x \in (x_j, x_{j+1})$. We will seek the interpolating function in the form of an algebraic polynomial

$$P_n(x) = \sum_{i=0}^{n} a_i x^i. \tag{1}$$

Since the polynomial $P_n(x)$ at nodal points must coincide with the given values of the function, the problem is reduced to solving the system of linear algebraic equations with respect to the unknowns a_i

$$\sum_{i=0}^{n} a_i x_j^i = y_j,\ j = k, \dots, k+n. \quad (2)$$

Let us consider the Lagrange interpolation polynomial. This polynomial was published by Joseph-Louis Lagrange in his work in 1795. The solution of the system can be represented in the form of the Lagrange interpolation polynomial:

$$P_n(x) = L_n(x) = \sum_{j=k}^{k+n} y_j \prod_{\substack{i=k \\ i \neq j}}^{k+n} \frac{x - x_i}{x_j - x_i}. \quad (3)$$

But Lagrange interpolation has one significant drawback. If it is necessary to obtain $L_{n+1}(x)$ by adding the node x_{n+1} to the existing interpolation nodes, all calculations have to be performed. The interpolation polynomial in the Newton's form has not this drawback.

To find the interpolation polynomial in Newton's form we introduce the notation $f_k = f(x_k)$. The expression for the divided difference of the order n is written in the form

$$f(x_k, x_{k+1,} x_{k+n}) = \sum_{j=k}^{k+n} f_j \left(\prod_{\substack{i=k \\ i \neq j}}^{k+n} (x_j - x_i) \right)^{-1}.$$

The Newton interpolation polynomial is the algebraic polynomial:

$$l_n(x) = f(x_k) + (x - x_k) f(x_k, x_{k+1}) + (x - x_k)(x - x_{k+1}) \dots (x - x_{k+n-1}) f(x_k, x_{k+1}, \dots x_{k+n}). \quad (4)$$

This polynomial is identically equal to a polynomial of degree n written at the Lagrange form or at some other form due to the uniqueness of the interpolation polynomial.

The problem of computing results accuracy is under consideration of different authors [7–11]. The following approach can be applied for error estimation. Let's present a mathematical model of the interpolation error. We consider two sets of nodes: $x_j^{(1)}$, $j = 0,..,N_1$ and $x_j^{(2)}$, $j = 0,..,N_2$ and two polynomials are built for them. Then we can write respectively two expressions

$$P_n^{(1)}(x) - f(x) = c \prod_{j=k_1}^{k_1+n} \left(x - x_j^{(1)} \right) + \delta_1(x), \quad P_n^{(2)}(x) - f(x) = c \prod_{j=k_2}^{k_2+n} \left(x - x_j^{(2)} \right) + \delta_2(x).$$

Here c is the value assumed to be independent of the position of the nodes; k_1 and k_2 are the numbers of the initial nodes used by the interpolation formula; $\delta_1(x)$ and $\delta_2(x)$ are small quantity in comparison with the first term. Neglecting small quantities, we solve the system of equations and find the estimate of the interpolation error:

$$P_n^{(1)}(x) - f(x) = \frac{\left(P_n^{(2)}(x) - P_n^{(1)}(x) \right) \Pi_1}{\Pi_2 - \Pi_1}.$$

Here we denote $\Pi_i = \prod_{j=k_i}^{k_i+m} \left(x - x_j^{(i)} \right)$.

Then more accurate value of the function is carried out:

$$f(x) \approx \frac{P_n^{(1)}(x) \Pi_2 - P_n^{(2)}(x) \Pi_1}{\Pi_2 - \Pi_1}.$$

Let us consider the case when the first set consists of the nodes with numbers from k to $k+n$, and the second set with numbers from $k+1$ to $k+n+1$. Then

$$P_n^{(1)}(x) - f(x) \approx \frac{\left[P_n^{(2)}(x) - P_n^{(1)}(x)\right]\prod_{j=k}^{k+n}(x - x_j)}{\prod_{j=k+1}^{k+n+1}(x - x_j) - \prod_{j=k}^{k+n}(x - x_j)} = \left[P_n^{(2)}(x) - P_n^{(1)}(x)\right]\frac{x - x_k}{x_{k+n+1} - x_k},$$

$$f(x) \approx \frac{x_{k+n+1}-x}{x_{k+n+1}-x_k}P_n^{(1)}(x) + \frac{x-x_k}{x_{k+n+1}-x_k}P_n^{(2)}(x) = P_{(n+1)}(x). \quad (5)$$

The function (9) is actually an interpolation polynomial of degree n+1 because:

– $P_{(n+1)}(x)$ is an algebraic polynomial of degree n+1;

– in nodes with numbers from $i = k+1$ to $i = k+n$ both polynomials $P_n^{(1)}(x_i)$ and $P_n^{(2)}(x_i)$, and therefore $P_{(n+1)}(x_i)$, coincide with $f(x_i)$;

– $P_{(n+1)}(x_k) = P_n^{(1)}(x_k) = f(x_k)$;

– $P_{(n+1)}(x_{n+k+1}) = P_n^{(2)}(x_{n+k+1}) = f(x_{n+k+1})$.

The formula (5) is called the recurrent Aitken ratio. The obtained polynomial of a higher degree can be used to estimate the error of the results of the calculated values of both polynomials $P_n^{(1)}(x_i)$ and $P_n^{(2)}(x_i)$.

Numerical experiment

Let $f(x) = \cos(x)$, $x_j = \frac{j}{m}\frac{\pi}{2}, y_j = f(x_j),\ j = 0, \dots, m$ and $m = 14$. The quantity $\Delta_n = |P_n(x) - P_{n+1}(x)|$ represents the error of interpolation; Δ_n^{exact} is the difference between the interpolated and the exact value; $k_\Delta = 1 - \Delta_n^{exact}/\Delta_n$ makes sense of the coefficient of refinement of the interpolated value. In order to construct the similar table, we take two sets of points $x_j^{(1)}$ from 1 to n+1 and $x_j^{(2)}$ from 0 to n. From the table 1 and the table 2 we can see that the values obtained with using the interpolation polynomial in the form of Lagrange and Newton do not differ significantly.

Table 1

Lagrange Polynomial Interpolation

n	$P_n(x)$	Δ_n	Δ_n^{exact}	k_Δ
1	0,997	1,56E-03	0,00157	-0,00554
2	0,986	1,48E-05	1,11E-05	0,247765
3	0,961	6,02E-06	6,07E-06	-0,008
4	0,924	1,34E-07	1,21E-07	0,096955
5	0,875	3,85E-08	3,88E-08	-0,00892
6	0,816	1,38E-09	1,31E-09	0,050273
7	0,746	2,95E-10	2,98E-10	-0,00953
8	0,666	1,48E-11	1,44E-11	0,666
9	0,579	2,47E-12	2,49E-12	0,579
10	0,484	1,64E-13	1,61E-13	0,484
11	0,383	2,09E-14	2,15E-14	0,383
12	0,277	1,89E-15	9,99E-16	0,277
13	0,168	4,50E-15	4,50E-15	0,168
14	0,056		-1,05E-14	0,056

Table 2

Newton Polynomial Interpolation

n	$P_n(x)$	Δ_n	Δ_n^{exact}	k_Δ
1	0,997	1,56E-03	0,00157	-0,00554
2	0,986	1,48E-05	1,11E-05	0,247765
3	0,961	6,02E-06	6,07E-06	-0,008
4	0,924	1,34E-07	1,21E-07	0,096955
5	0,875	3,85E-08	3,88E-08	-0,00892
6	0,816	1,38E-09	1,31E-09	0,050273
7	0,746	2,95E-10	2,98E-10	-0,00953
8	0,666	1,48E-11	1,44E-11	0,666
9	0,579	2,47E-12	2,49E-12	0,579
10	0,484	1,64E-13	1,61E-13	0,484
11	0,383	2,13E-14	2,17E-14	0,383
12	0,277	1,33E-15	1,22E-15	0,277
13	0,168	0	6,38E-16	0,168
14	0,056		1,80E-14	0,056

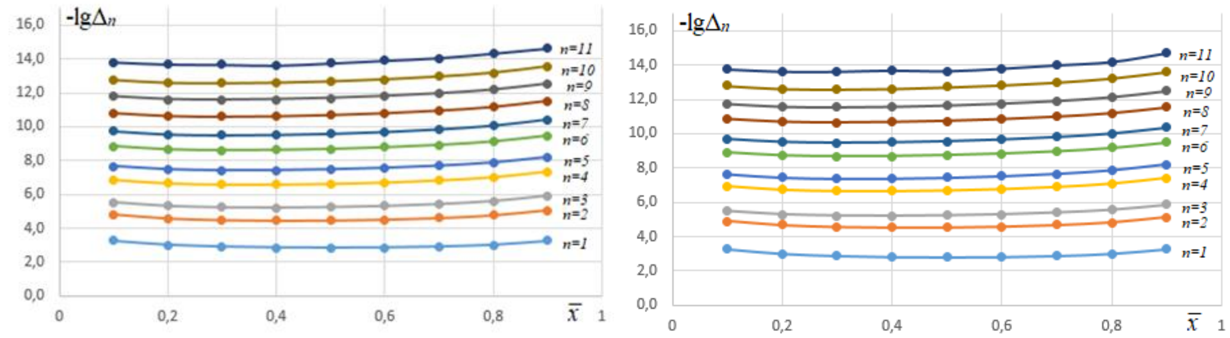
Table 3

Interpolation by Lagrange Polynomial with Two Points Sets

n	$P_n(x)$	Δ_n	Δ_n^{exact}	k_Δ
1	0,998	1,48E-05	8,65E-06	0,414343
2	0,986	3,61E-06	3,66E-06	-0,01229
3	0,961	5,75E-08	4,82E-08	0,162379
4	0,924	1,28E-08	1,30E-08	-0,01431
5	0,875	3,75E-10	3,43E-10	0,084261
6	0,816	6,81E-11	6,91E-11	-0,01554
7	0,746	2,96E-12	2,81E-12	0,050146
8	0,666	4,36E-13	4,43E-13	-0,017
9	0,579	2,59E-14	2,50E-14	0,034
10	0,484	2,94E-15	2,61E-15	0,113
11	0,383	3,33E-16	4,44E-16	-0,333
12	0,277	4,44E-16	3,89E-16	0,125
13	0,168		1,94E-15	
14	0,056		8,20E-15	

One can see from table 3, that interpolation using the Lagrange interpolation polynomial with two sets of points accelerated the refinement of the value.

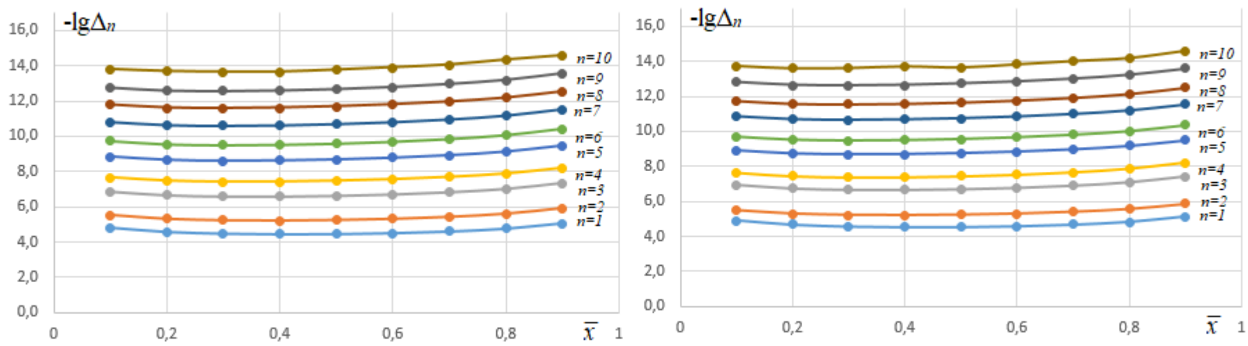
Application of logarithmical scale for visualization of error representation are described in [12–14].



a *b*

Figure 1: Computing results for the Lagrange interpolation polynomial:
a — $\Delta_n = |P_n(x) - P_{n+1}(x)|$; b — $\Delta_n = |P_n(x) - f(x)|$

It is convenient to represent the results of interpolation and the estimation of error on the graph in the form of the dependence of $-\lg\Delta_n$ on $\bar{x} = (x - x_j)/(x_{j+1} - x_j)$, $x \in (x_j, x_{j+1})$. In Figure 1 and Figure 2 the different curves correspond to different n ($j = 2$). The pairwise approach at location of the curves is explained by the fact that the function cos (x) is even, and only even terms present in its expansion for powers of x. In Figure 1, b and 2, b it is shown that curves are similar to the curves in Figure 1, *a* and Figure 2 accordingly, but the exact values of the cos (x) function are used to estimate the error. It is also seen that the curves in Figure 2 correspond to the curves in Figure 1 for $n + 1$.



a b

Figure 2: Computing results for the Newton interpolation polynomial with two sets of points:
a — $\Delta_n = |P_n(x) - P_{n+1}(x)|$; b — $\Delta_n = |P_n(x) - f(x)|$

CUBIC SPLINE

Description of the method and formulation

Let the segment $[a, b]$ be divided into n partial segments $[x_i, x_{i+1}]$, where $x_i < x_{i+1}$, $i = 0, 1, \ldots, n-1$, $x_0 = a$, $x_n = b$. Denote $h_i = x_i - x_{i-1}$. In the case of a uniform partition $h = (b-a)/n$, $x_i = a + ih$.

The function $f(x)$ is given by its values at the nodal points x.

A spline is a function that, together with several derivatives, is continuous on the entire given segment $[a, b]$, and on each partial segment $[x_i, x_{i+1}]$ separately is some algebraic polynomial.

$$y(x) = \begin{cases} f_1(x), x \in [x_0, x_1], \\ f_2(x), x \in [x_1, x_2], \\ \quad \ldots \qquad\quad \ldots \\ f_n(x), x \in [x_{n-1}, x_n]. \end{cases}$$

For this study a spline of the third degree was chosen, which has continuous the first derivative on the segment $[a, b]$. Let's denote it by $S_3(x)$. On each segment, the cubic spline has the following form:

$$S_3(x) = a_{i0} + a_{i1}(x - x_i) + a_{i2}(x - x_i)^2 + a_{i3}(x - x_i)^3, x \in [x_i, x_{i+1}]$$

and satisfies the following conditions:

$$S_3(x_i) = f(x_i),\ i = 0, \dots, n. \tag{6}$$

Based on the fact that the spline on each segment is determined by four coefficients to construct it on the entire segment, it is required to determine $4n$ coefficients. For their unambiguous definition, we need to set $4n$ equations. In addition, the condition (6) gives $2n$ equations, since this polynomial must pass through two given points: the beginning and the end of the segment. Moreover, the function $S_3(x_i)$ satisfying these conditions is continuous at all internal nodes.

The condition of continuity of the derivatives of the spline $S_3'(x), S_3''(x)$ at all internal nodes x_i, $i = 1, \dots, n-1$ of the grid, gives $2(n-1)$ equalities. In total, we get $4n-2$ equations.

Two additional (boundary) conditions are usually set in the form of restrictions on the value of the derivatives of the spline at the ends of the interval $[a, b]$.

In order to construct an interpolation cubic spline we use the following algorithm.

Let each value of the argument x_i, $i = 0, \dots, n$ corresponds to the value of the function $f(x_i) = y_i$ and it is required to find a functional dependence in the form of a spline satisfying the following requirements:

1) The function $S_3(x)$ is continuous on the segment $[a, b]$ together with its derivatives up to the second order inclusive.
2) $S_3(x_i) = y_i$, $i = 0, 1, \dots, n$.
3) The function $S_3(x)$ satisfies one of the variants of the boundary conditions.

The formulated problem has a unique solution.

The second derivative $S''_3(x)$, which is expressed by a continuous linear function, can be represented as the Lagrange polynomial of the first degree:

$$S_3''(x) = \frac{x_i - x}{h_i} m_{i-1} + \frac{x - x_{i-1}}{h_i} m_i,$$

where $h_i = x_i - x_{i-1}$, $m_i = S''_3(x_i)$.

We integrate twice both parts of the expression and use the conditions of continuity of the function and the first derivative, so, the following system of equations is obtained:

$$\frac{h_i}{6} m_{i-1} + \frac{h_i + h_{i+1}}{3} m_i + \frac{h_{i+1}}{6} m_{i+1} = \frac{y_{i+1} + y_i}{h_{i+1}} - \frac{y_i - y_{i-1}}{h_i}, \qquad i - 1, \dots, n - 1.$$

Finally, solving the system of equations with respect to the parameters m_i, we get:

$$S_3(x) = \frac{(x_i - x)^3 - h_i^2(x_i - x)}{h_i} m_{i-1} + \frac{(x - x_{i-1})^3 - h_i^2(x - x_{i-1})}{h_i} m_i + \frac{x_i - x}{h_i} y_{i-1} + \frac{x - x_{i-1}}{h_i} y_i.$$

This formula is used to calculate the values of the function $S_3(x)$. It is important to note that the system of equations is solved by the sweep method.

Numerical experiment

Let $f(x) = \cos(x)$, we interpolate the function on the segment $[0, \pi]$ with a uniform partition with doubling the number of segments n. The boundary conditions are the equality of the second derivative to 1 at the left boundary of the segment and to – 1 at the right one.

The column Δ_{max} represents the maximum error $|S_3(x) - f(x)|$, calculated at the points located between the grid nodes, K_Δ is the ratio of the error of the previous line to following one (coefficient of error reduction for doubling n).

It can be seen in table 4 that K_Δ retains the value corresponding to the fourth order of accuracy ($K_\Delta \approx 2^4$) up to values of n = 1000 – 3000, above which the rounding error prevails in the total error of the result.

Table 4

Cubic Interpolation with Correct Boundary Conditions

n	Δ_{max}	$\Delta_{оц}$	K_Δ
5	1,02E-03	-	-
10	6,29E-05	6,39E-05	16,25636
20	3,91E-06	3,93E-06	16,0827
40	2,44E-07	2,45E-07	16,02083
80	1,53E-08	1,53E-08	16,00522
160	9,54E-10	9,54E-10	16,00131
320	5,96E-11	5,96E-11	16,00031
640	3,73E-12	3,73E-12	16,00015
1280	2,33E-13	2,33E-13	16,00286
2560	1,45E-14	1,46E-14	16,00763
5120	1,11E-15	9,09E-16	13,1
10240	2,22E-16	2,78E-16	5

It should be noted that for the practical application of the error estimate, it is necessary to know the upper estimate of the k^{th} derivative of the function $f(x)$. But this is not always possible. To estimate the error, you can apply a rule using the regularity of the dependence of the error on h or n. It is observed that with an increase in the number of nodes, the interpolation error at any particular point x may vary irregularly, since the position of this point relative to neighboring nodes (the ratio $(x-x_{j-1})/(x_j-x_{j-1})$) may vary for different n. Using the example of this numerical experiment, it is seen that the maximum error on the segment $[0, \pi]$ decreases by a factor of $K_\Delta \approx 2^k$ when n is doubled.

Using the property of conservation of the K_Δ value for the maximum error $\Delta_{max}(n)$, we can obtain an estimate in the form

$$\Delta_{оц}(2n) = \frac{\Delta_{max}(n)}{2^k}.$$

For this it is necessary to have a method for estimating the quantity $\Delta_{max}(n)$, even if the exact value of the interpolated function f (x) is unknown. You can use the following method, which consists in comparing the values of $S_3(x)$ calculated for different numbers of segments into which the segment $[a, b]$ is divided n and $2n$ times. When n is doubled with a uniform or non-uniform partition, n new nodal points $x_{j-1/2}$ appear, lying between the common nodes x_{j-1} and x_j (table 4). Then, as the estimate for $\Delta_{max}(n)$, we choose

$$\Delta_{max}(n) \approx \max_{1\le j\le n}\left|S_3^{2n}(x_{j-1/2}) - S_3^{n}(x_{j-1/2})\right|.$$

The correct boundary conditions are chosen for Table 4. However, for other boundary conditions, a relatively large interpolation error is observed. For clarity, the result is presented in the form of the Table 5.

Table 5

Cubic Interpolation with Incorrect Boundary Conditions

n	Δ_{max}	$\Delta_{оц}$	K_Δ
5	1,90E-02	-	-
10	4,58E-03	4,74E-03	4,14E+00
20	1,13E-03	1,14E-03	4,04E+00
40	2,82E-04	2,83E-04	4,01E+00
80	7,06E-05	7,06E-05	4,00E+00
160	1,76E-05	1,76E-05	4,00E+00
320	4,41E-06	4,41E-06	4,00E+00
640	1,10E-06	1,10E-06	4,00E+00
1280	2,76E-07	2,76E-07	4,00E+00
2560	6,89E-08	6,89E-08	4,00E+00
5120	1,72E-08	1,72E-08	4,00E+00
10240	4,31E-09	4,31E-09	4,00E+00

CONCLUSIONS

Thus, after carrying out the numerical experiment, we can say that the results of interpolation obtained for interpolation polynomials in the Lagrange and Newton forms do not have significant differences in accuracy. Two sets of n points forming from one data set and two polynomials of a degree n allows getting an interpolation polynomial of $n + 1$ degree, which is used to estimate the error of both polynomials of n degree. An insignificant difference between the graphs in Figure 1, *a*, *b* and Figure 2, *a*, *b* indicates a high accuracy of the estimation of the interpolation error. The degree of polynomials does not depend in any way on the number of grid nodes, and therefore it does not change with its increase. Unlike the Lagrange interpolation polynomials, the sequence of cubic interpolation splines for the uniform grid always converges to an interpolated continuous function. The disadvantage of a cubic spline is the choice of the boundary conditions. With the help of the boundary conditions, we can include the parameters in the construction of the spline and control the spline behavior. When choosing incorrect boundary conditions (sometimes it is proposed to take $S''_3(a) = S''_3(b) = 0$), the accuracy of interpolation of the function and its first derivative, as a rule, decreases.

REFERENCES / СПИСОК ЛИТЕРАТУРЫ

1. Amosov A. A., Dubinsky Y. A., Kopchenova N. V. Computational Methods. Moscow: MPEI Publishing House, 2008.
2. Bjoerck A., Dahlquist G. Numerical Mathematics and Scientific Computation: Vol. 1. 1999.
3. Richargson L. W., “The deferred approach to the limit”. Phil. Trans. Roy. Soc. London, vol. 226, 1927, p. 299-361.
4. Paluri N. S. V., Sondur S., “Experiments with range computations using extrapolation”, Reliable Computing, 2007, vol. 13, no. 1, pp. 1-23.
5. Smith D. A., Ford W. F., “Numerical comparisons of non-linear convergence accelerations”. Mathematics of Computation, 1982, vol. 38, 158, pp. 481-499.
6. Sheykhalina N. M. Mathematical Modeling of Technical Objects and Processes Based on Methods of Multicomponent Analysis of Computational Experiment Results: Dissertation. Dr. Tech. Sci. Ufa, 2012.
7. Житников В. П., Шерыхалина Н. М., Соколова А. А. Предельно-квазистационарное копирование круглого электрод-инструмента // Вестник УГАТУ. 2017. Т. 21, № 1. С. 173–179. [[Zhitnikov V. P., Sheykhalina N. M., Sokolova A. A., “Marginal-quasi-stationary copying of circular electrode-tool”. Vestnik UGATU, vol. 21, no. 1, pp. 173-179, 2017.]]
8. Zhitnikov V. P., Sheykhalina N. M., Sokolova A. A., “Problem of reliability justification of computation error estimates”. Mediterranean Journal of Social Sciences, 2015, vol. 6, no. 2, pp. 65-78.
9. Житников В. П., Шерыхалина Н. М., Поречный С. С. Об одном подходе к практической оценке погрешностей численных результатов // Научно-техн. ведомости СПбГПУ. 2009. № 3 (80). С. 105–110. [[Zhitnikov V. P., Sheykhalina N. M., Porechnyy S. S., “About one approach to practical assessment of errors in numerical results”. SPbSPU, 2009, no 3 (80), pp. 105-110.]]

10. Zhitnikov V. P., Sheykhalina N. M., "Accuracy increase of complex problems solutions by numerical data post-processor handling". Computational Technologies, vol. 13, no. 6, 2008, pp. 61-65.
11. Sherykhalina N. M., Zhitnikov V. P., "Application of extrapolation methods of numerical results for improvement of hydrodynamics problem solution". Computational Fluid Dynamics Journal, vol. 10, no. 3, 2001.
12. Житников В. П., Шерыхалина Н. М., Соколова А. А. Оценка погрешности и ее обоснование с помощью фильтрации численных результатов, полученных при разных числах узловых точек сетки. // Известия Самарского научного центра РАН. 2017. Т. 19, № 1 (2). С. 401–405. [[Zhitnikov V. P., Sheykhalina N. M., Sokolova A. A., "Estimation of error and its justification by filtering of numerical results obtained at different numbers of grid nodes". Izvestiya Samara Scientific Center RAS, 2017, T.19, no. 1 (2), pp. 401-405.]]
13. Zhitnikov V. P., Sheykhalina N. M., "Methods of verification of mathematical models under uncertainty". Vestnik UGATU, 2000, no. 2, pp. 53-60.
14. Житников В. П., Шерыхалина Н. М., Федорова Г. И., Соколова А. А. Методика качественного улучшения результатов вычислительного эксперимента // Системная инженерия и информационные технологии. 2021. Т. 3, №1 (5). С. 58-64. [[Zhitnikov V. P., Sherykhalina N. M., Fedorova G. I., Sokolova A. A., "Methodology for qualitative improvement of the results of a computational experiment". System Engineering and Information Technologies, 2021, vol. 3, no 1 (5), pp. 58-64.]]

Поступила в редакцию 22 апреля 2023 г.

МЕТАДАННЫЕ / METADATA

Название: Численное исследование различных методов интерполяции.

Аннотация: Статья посвящена приближенному решению некоторых задач вычислительной математики. Обсуждается практическая интерполяция для решения задач. Рассматриваются различные методы интерполяции, такие как многочлен Лагранжа, многочлен Ньютона и кубический сплайн, и проводится подробный сравнительный анализ этих методов. Описанные методы применяются для восстановления функциональных зависимостей на примере элементарной функции. Показан численный эксперимент по восстановлению функциональной зависимости описанными методами. Представлена математическая модель погрешности интерполяции. С помощью двух полиномов, построенных на двух разных сетках, получается многочлен более высокой степени. Показано, что по рекуррентному коэффициенту Эйткена полученный многочлен более высокой степени может быть использован для оценки погрешности результатов вычисленных значений. Результаты расчетов и оценки погрешностей получены с большой точностью. Сравниваются результаты различных методов интерполяции.

Ключевые слова: интерполяция; численный метод; многочлен Лагранжа; многочлен Ньютона; сплайн; модель погрешности

Язык статьи / Language: английский / English.

Поддержка/Support: нет / no.

Об авторах / About authors:

Шерыхалина Наталия Михайловна
ФГБОУ ВО «Уфимский университет науки и технологий», Россия.
Проф. каф. выч. математики и кибернетики. Дипл. инж.-с/техник (Уфимск. гос. авиац. техн. ун-т, 1993). Д-р техн. наук по мат. моделир., числ. методам и компл-м программ (УГАТУ, 2012). Иссл. в обл. мат. моделир. течений жидкости и э/хим. формообразования, числ. методов и оценок погр.
E-mail: n_sher@mail.ru
ORCID: 0000-0002-2808-1311
URL: https://elibrary.ru/author_profile.asp?authorid=10871

Sherykhalina Nataliya Mikhailovna
Ufa University of Science and Technologies, Russia.
Prof., Dept. Computational Mathematics and Cybernetics, Faculty of Computer science and robotics, UGATU. Dipl. System engineer (UGATU, 1993). Dr. Tech. Sci. (UGATU, 2012). Research in the field of mathematical modeling of fluid flows and electrochemical shaping, numerical methods and error estimates.
E-mail: n_sher@mail.ru
ORCID: 0000-0002-2808-1311
URL: https://elibrary.ru/author_profile.asp?authorid=10871

Соколова Александра Алексеевна
ФГБОУ ВО «Уфимский университет науки и технологий», Россия.
Асс. каф. выч. математики и кибернетики. Дипл. информатик-математик (Уфимск. гос. авиац. техн. ун-т, 2014). Готовит дис. по мат. моделир. течений жидкости и э/хим. формообразования методами числ. фильтрации.
E-mail: alexandrakrasich@gmail.com

Sokolova Aleksandra Alekseevna
Ufa University of Science and Technologies, Russia.
Postgrad. (PhD) Student, Dept. Computational Mathematics and Cybernetics, Faculty of Computer Science and Robotics. Dipl. computer scientist-mathematician (Ufa State Aviat. Techn. University, 2014).
E-mail: alexandrakrasich@gmail.com

Шаймарданова Екатерина Ринатовна
ФГБОУ ВО «Уфимский университет науки и технологий», Россия.
Студ. бак-та по мат. обесп. и админ. информ. систем.
E-mail: shaymardanova.ekaterina.04@gmail.com

Shaymardanova Ekaterina Rinatovna
Ufa University of Science and Technologies, Russia.
Student, Faculty of Comp. Sci. & Robotics.
E-mail: shaymardanova.ekaterina.04@gmail.com