Non-Sibsonian interpolation on arbitrary system of points in Euclidean space and adaptive isolines generation

Vitali V. Belikov a,*, Andrei Yu. Semenov b,1

a Computing Centre, Russian Academy of Sciences, 40 Vavilov Street, Moscow 117333, Russia
b General Physics Institute, Russian Academy of Sciences, 38 Vavilov Street, Moscow 117942, Russia

Abstract

A new method for function interpolation on a set of arbitrary points in a finite-dimensional Euclidean space $\mathbb{E}^n$ is presented. This method differs from the well-known Sibson method. The properties of the new method are described including specific “harmonic” property. Comparison with the Sibson interpolation and with the interpolation based on the Delaunay triangulation are reviewed. The effective and economical algorithm for isolines generation based on the non-Sibsonian and the Delaunay interpolations is presented. The isolines have no intersections nor any losses in the numerical information. A compact algorithm of the higher-order non-Sibsonian interpolation is also described. © 2000 IMACS. Published by Elsevier Science B.V. All rights reserved.

Keywords: Non-Sibsonian interpolation; Euclidean space; Adaptive isolines generation

1. Introduction

In this paper we describe the non-Sibsonian interpolation algorithm. It is a new method for interpolating the values of a scalar function $f$ on a set of arbitrary points in a finite-dimensional Euclidean space $\mathbb{E}^n$. We define some of the properties and review the results of its comparison with the Sibson interpolation [25,26] (1980) and with the interpolation based on the Delaunay triangulation [11–13]. Delaunay triangulation was first proposed and studied by B.N. Delaunay in 1928–1929 [11,12] the most detailed description of the results were presented in [13] (see the references in [14] too). We present the compact algorithm for construction of the higher-order interpolation and describe the results obtained by applying the effective algorithm of isolines generation based on the non-Sibsonian interpolation and the Delaunay triangulation. These isolines have no intersections and no losses of fine structure on function behaviour. Numerical results of isolines generation are presented for tests and for results obtaining in the field of fluid dynamics modeling in the framework of the shallow water equations with natural complex-shaped relief.

* Corresponding author.
1 E-mail: say@lpl.gpi.ru

0168-9274/00/$20.00 © 2000 IMACS. Published by Elsevier Science B.V. All rights reserved.

PII: S0168-9274(99)00058-6
The proposed algorithm calculates the value $f_0 = f(x_0)$ a scalar function $f = f(x)$ at a prescribed point $x_0$ in $\mathbb{R}^n$. This calculation is based on the known function values \{$f_i = f(x_i)$\} on a fixed system of points (nodes) \{$x_i$\} in $\mathbb{R}^n$. The point to which we interpolate the values of $f(x)$ is assumed to be inside the domain bounded by a convex hull constructed on the basis of points.

A set of applied methods and procedures can use the interpolation algorithms of this paper. In particular, algorithms of this type are used in computer graphics, computational hydrodynamics (for, e.g., flow calculations on arbitrary grids or by means of various particle methods [8,10,16]), data processing in cartography and geodesy, etc. Creation and investigation of such interpolation methods is significance and importance in surface data fitting, solution of partial differential equations, and for improvement of existing numerical algorithms.

2. Interpolation methods

We shall consider interpolation methods for calculating the value $f_0$ of a function $f$ at a point $x_0$ in $\mathbb{R}^n$ in terms of its values \{$f_i = f(x_i)$\} on a fixed set of points \{$x_i$\} in $\mathbb{R}^n$ that are reducible to the final formula:

$$f_0 = \sum_{m=1}^{M} \alpha_m f_m, \quad \sum_{m=1}^{M} \alpha_m = 1, \quad \alpha_m \geq 0, \quad m = 1, 2, \ldots, M. \quad (1)$$

$\alpha_m$ are interpolation coefficients, which depend on the location of the set of points and do not depend on $f$. $M$ is the number of points used to interpolate the function $f$. These points are called the neighbors of $x_0$. An additional requirement for formula and the coefficients $\alpha_m$ is that hold any linear function $f$. It is the first-order interpolation property. The condition $\sum_{m=1}^{M} \alpha_m = 1$ ensures that (1) holds exactly for the constant function $f$. The additional conditions $\alpha_m > 0$ guarantees the boundedness of the norm of the interpolated result: $\|f_0\| \leq \max_m \|f_m\|$.

Interpolation algorithms differ in terms of how the coefficients $\alpha_m$ are chosen and in terms of the neighbors of the given point $x_0$ in $\mathbb{R}^n$. Among numerous well-known methods only the Sibson interpolation [25,26] guarantees the uniqueness and continuity of the results of interpolation and their stability with respect to small geometric perturbations. Until now it was not known whether another method similar to the form (1) existed. This paper describes the alternative non-Sibsonian method [1,3] of interpolation and thus demonstrates the nonuniqueness of the Sibson method.

Analysis of uniqueness and continuity of interpolation is motivated by the fact that one approach to the interpolation of functions requires dividing the space by nonintersecting simplexes in $\mathbb{R}^n$ (triangles for $n = 2$, tetrahedrons for $n = 3$, etc.) with vertices at the nodes of the given set $\{x_i\}$. The interpolated function is approximated by a smooth (e.g., linear) function. This allows us to construct $f_0$ at any point $x_0$. A correct decomposition of a space into triangles (or simplexes in $\mathbb{R}^n$), which is called a triangulation, can be performed by various methods and is not unique. The accuracy of interpolation depends on the quality of triangulation. It has been shown that the error of linear interpolation is reduced by increasing the minimal angle [23]. There is a specific triangulation, among others, that has a number of minimax properties. This is the triangulation that maximizes the minimal angle over all triangles. It is called the Delaunay triangulation (see [11–13,17,22,24]) and is also nonunique. (It is sufficient to consider a square, which can be triangulated by either diagonal.) The nonuniqueness implies that the structure of the neighbors may drastically change with a small change in the coordinates of the nodes,
which leads to considerable changes in the results of interpolation. This property leads to interpolation errors, nonsmooth behaviour of the results and nonuniqueness.

Decomposition of the space into Dirichlet cells or convex polyhedrons of the Dirichlet domain is another important example. For a given node \( x_0 \) the Dirichlet cell is the union of points from which the distance to the point \( x_0 \) is less than to other node. Among the numerous properties of these cells, we should emphasize the uniqueness of decomposition and the continuous dependence of all geometric parameters of cells on the coordinates of nodes. One method for constructing these cells is too slow to be used in applications—the corresponding number of operations is \( O(N^2) \), where \( N \) is the number of points. The approach is: if we link the point \( x_0 \) with all points \( x_k \) by segments and erect a perpendicular (or a hyperplane in \( \mathbb{E}^n \)) at the center of each segment on both sides, then the resulting convex polygon (polyhedron) is the Dirichlet cell. The Dirichlet cells decompose the entire space into convex polygons (polyhedrons), each contain only one of the nodes. Algorithms for decomposition of \( \mathbb{E}^n \) into convex polyhedrons of the Dirichlet domains were first proposed and studied in 1908–1909 [29]; see paper The Voronoi Types of Lattices by B.N. Delaunay in [19]. The decompositions of this type are called the Dirichlet–Voronoi tessellations (see paper Tessellation in [20]). The decomposition into Dirichlet cells is uniquely constructed and is the geometric dual of the Delaunay triangulation.

It is interesting to note that there are a number of physical problems in which the concept of the Dirichlet cell is introduced in a natural manner, for example, in geometric crystallography, see comments by B.N. Delaunay in [29] and references in [14]. Many other interesting application (crystallography, geographical information systems, econometrics, etc) of the Dirichlet–Voronoi tessellation also appear.

There exist various practical methods for constructing Dirichlet cells by Delaunay triangulation [9, 18]. Some example of generating Dirichlet cells are shown in Fig. 1. We indicate nodes inside the

![Fig. 1. Two examples of the Dirichlet–Voronoi tessellations.](image)
cells by black points. The domain boundaries are shown by lines. In this communication we construct a decomposition by a method that is analogous to the Bowyer method [9], with minor modifications introduced for faster performance. First, to preclude construction of unbounded Dirichlet cells, all points are placed in enveloping simplex. Second, to accelerate the geometric search, we use the fast quadtree type algorithm. In this decomposition algorithm, the number of operations is $O(N \log N)$, where $N$ is complete number of nodes. The construction of Dirichlet cells naturally leads to the definition of neighbors for the point $x_0$. These are the nodes belonging to the Dirichlet cells that have common faces with the cell around $x_0$.

**The Sibson interpolation.** Defining the neighbors in this manner allows us to formulate the Sibsonian interpolation method [25,26] in a form similar to (1). In this case, the coefficients are set proportional to the areas (volumes) $V_m$ of the polygons (polyhedrons) that are cut from the Dirichlet cell for $x_0$ by the Dirichlet cells that are constructed for the corresponding neighbors of $x_0$ in the absence of $x_0$:

$$f_0 = \sum_{m=1}^{M} \alpha_m f_m, \quad \alpha_m = \frac{V_m}{\sum_{j=1}^{M} V_j}, \quad m = 1, 2, \ldots, M.$$ 

The rigorous definition of the Sibson interpolation is as follows. Let $\{x_i\}, \ i = 0, 1, 2, \ldots, N$, be the set of separate points in the Euclidean space $\mathbb{E}^n$. Suppose that

$$T_k = \{x \in \mathbb{E}^n: d(x, x_k) < d(x, x_m), \ m \neq k\},$$

$$T_{km} = \{x \in \mathbb{E}^n: d(x, x_k) < d(x, x_m) < d(x, x_l), \ l \neq m, k\} \quad (d \text{ is the Euclidean distance}),$$

and $T_k$ is the Dirichlet cell for $x_k$. Then $T_k = \bigcup_{m, m \neq k} T_{km}$ ($T_{km}$ may be an empty set). If $|T_{km}| < +\infty$, then

$$\sum_{m \neq k} |T_{km}| x_m = |T_k| x_k,$$

where $| \cdot |$ denotes the Lebesgue measure in $\mathbb{E}^n$. $|T_{km}|$ defined in this way are called the Sibson coefficients (coordinates). Their properties, formal definition and generalizations are described in [15,25,26]. In terms of notation used above, $\alpha_m$ are introduced as

$$\alpha_m = \frac{|T_{0m}|}{|T_0|},$$

which is $\alpha_m$ for the point $x_0$. Let us consider interpretation of the Sibson interpolation. Eq. (1) has the form

$$f_0 \left( \sum_{m=1}^{M} V_m \right) = \sum_{m=1}^{M} V_m f_m,$$

and, hence, in continuous approach such $f_0$ calculation conserves the value of volume integral $\iiint f(x) \, dV = \iiint f(x) \, dx_1 \ldots dx_n$ in $\mathbb{E}^n$. 
3. Non-Sibsonian interpolation

The method of interpolation proposed here is based on determining the neighbors through decomposition into the Dirichlet cells [1,3]. Let \( x_0 \) belong to the Dirichlet polygon (polyhedron) with \( M \) sides (faces). Denote the lengths of sides (or the areas of faces in the three-dimensional case) by \( s_m \), \( m = 1, 2, \ldots, M \). Denote further the lengths of the altitudes drawn from \( x_0 \) to \( m \)th sides (i.e., the distances from \( x_0 \) the \( m \)th face), by \( h_m \). Then, (1) becomes

\[
f_0 = \sum_{m=1}^{M} \alpha_m f_m, \quad \alpha_m = \frac{s_m}{h_m} \left/ \sum_{j=1}^{M} \frac{s_j}{h_j} \right., \quad m = 1, 2, \ldots, M. \tag{2}
\]

This method of calculating is simpler and more economical than that employed in the Sibson method since it does not require computing the intersection areas of polygons in the two-dimensional case or the intersection volumes of polyhedrons in the three-dimensional case.

We give the following rigorous definition of the new method. Let \( \{x_i\}, \; i = 0, 1, \ldots, N \), be the set of separate points in the Euclidean space \( \mathbb{E}^n \). Suppose \( T_k = \{x \in \mathbb{E}^n: d(x, x_k) < d(x, x_m), \; m \neq k\} \) and \( t_{km} = \{x \in T_k \cap T_m, \; m \neq k\} \) (\( d \) is the Euclidean distance), where \( T_k \) is the Dirichlet cell for \( x_k \), and \( t_{km} \) may be an empty set. If \( d(x_k, x_m) \neq 0 \), then

\[
\sum_{n \neq k} \frac{|t_{km}| x_m}{d(x_k, x_m)} = x_k \left[ \sum_{n \neq k} \frac{|t_{km}|}{d(x_k, x_m)} \right],
\]

where \( | \cdot | \) denotes the Lebesgue measure in \( \mathbb{E}^{n-1} \). In terms of the notation used above,

\[
\alpha_m = \frac{|t_{0m}|}{d(x_0, x_m)} \left/ \sum_{j=1}^{M} \frac{|t_{0j}|}{d(x_0, x_j)} \right.,
\]

which means \( \alpha_m \) for the point \( x_0 \).

The interpolation properties, such as uniqueness and continuous dependence of its coefficients \( \alpha_m \) on the coordinates of nodes, follow from its definition. The main property theorem was proved in [1]. The linear combination of the Sibson and the non-Sibson interpolation define an one-parameter class of interpolation. Is there any more interpolation family with such properties is unknown. We can state the hypothesis that all interpolations are the linear combination of the Sibson and this non-Sibsonian interpolations.

Let us first prove that (2) defines a linear interpolation in the planar (two-dimensional) case. We consider a linear function \( f(x, y) = ax + by + c \) on the plane \( (x, y) \), and show that (2) is an identity in this case. We first place \( x_0 \) (for definiteness) at the point \((0, 0)\). With a shift by a constant, we set \( f(0, 0) = 0 \) (we can do this because \( \sum_{m=1}^{M} \alpha_m = 1 \)). Then, changing from \((x, y)\) in \( \mathbb{E}^2 \) to \( z = x + iy \) in the complex plane \( \mathbb{C} \), the criterion that must be proved for linear interpolation is reduced to (3):

\[
\left[ \sum_{m=1}^{M} (x_m + iy_m) \frac{s_m}{h_m} \right] \left( \sum_{j=1}^{M} \frac{s_j}{h_j} \right)^{-1} = 0 + i0. \tag{3}
\]
To prove this, we note that the sides of the Dirichlet cell that contain \((0, 0)\) are the complex vector-numbers \(z_m\). Since the polygon has a closed contour, the following identity holds:

\[
\sum_{m=1}^{M} z_m = 0.
\]  

Introducing the trigonometric form \(z_m = s_m \exp(i\varphi_m)\), we rearrange (4) as follows:

\[
\sum_{m=1}^{M} z_m = \sum_{m=1}^{M} z_m^+ = \sum_{m=1}^{M} s_m \exp(-i\varphi_m) = -i \sum_{m=1}^{M} s_m \exp(i\pi/2 - i\varphi_m)
\]

\[
= -i \sum_{m=1}^{M} \left[h_m \exp(i\pi/2 - i\varphi_m)\right] \frac{s_m}{h_m} = -i \sum_{m=1}^{M} (x_m + iy_m) \frac{a_m}{2} = 0.
\]

This proves (3). In the proof we used \(|x_m + iy_m| = 2h_m\).

Now, we prove the same for the case of arbitrary \(E^n\). By analogy consider a linear function \(f(x)\) in \(E^n\) and show that (2) holds for it identically. We first place \(x_0\) (for definiteness) at the point \(x = 0\) and, using a shift by a constant, set \(f(0) = 0\) (we can do this because \(\sum_{m=1}^{M} a_m = 1\)). Then the criterion for linear interpolation that is to be proved has the following vector form:

\[
\left( \sum_{m=1}^{M} \frac{x_m s_m}{h_m} \right) \left/ \left( \sum_{j=1}^{M} \frac{s_j}{h_j} \right) \right. = 0.
\]  

To prove this, we consider the unit base vector \(I_p\) with number \(p\) in orthogonal coordinate system introduced in \(E^n\), where \(p = 1, 2, \ldots, n\). Consider now the scalar flux of a constant vector field equal to \(I_p\) across the closed surface \(S_0\) of the Dirichlet polyhedron that encloses \(x = 0\) and has the volume \(V_0\). By the divergence theorem,

\[
\oint_{S_0} (I_p, dS) = \int_{V_0} \int \text{div} I_p dV = 0.
\]

In Eq. (6) the formula \((a, b)\) denotes a scalar product of two vectors \(a\) and \(b\). Using (6), we obtain the following sequence of identities:

\[
\sum_{p=1}^{n} \oint_{S_0} (I_p, dS) = \sum_{p=1}^{n} I_p \cdot 0 = \sum_{p=1}^{n} I_p \sum_{m=1}^{M} \left( \frac{s_m x_m}{|x_m|} \cdot I_p \right)
\]

\[
= \sum_{p=1}^{n} I_p \sum_{m=1}^{M} \frac{s_m (x_m \cdot I_p)}{2h_m} = \sum_{m=1}^{M} \left[ \sum_{p=1}^{n} I_p (x_m \cdot I_p) \right] \frac{s_m}{2h_m} = \sum_{m=1}^{M} \frac{x_m s_m}{2h_m}.
\]

This proves (5). In the proof, we used \(|x_m| = 2h_m\).

Other properties of the interpolation, such as uniqueness and continuous dependence of its coefficients on the coordinates of nodes, follow from its definition and can be elementary proved see, for example, [21].
Harmonic property. Let us consider an interpretation of the non-Sibsonian interpolation. Eq. (2) has the form

$$\sum_{m=1}^{M} \frac{(f_m - f_0)}{h_m} s_m = 0.$$  

Hence, in discrete case it approximates and in continuous consideration it satisfies the equality

$$\oint_{S_0} \left( \frac{\partial f}{\partial \mathbf{n}} \right) dS = 0 \quad \text{or} \quad \Delta f = 0, \quad \text{where} \quad \Delta = \frac{\partial^2}{\partial x_1^2} + \cdots + \frac{\partial^2}{\partial x_n^2}.$$ 

dS is a boundary surface element, n is a normal. Therefore, (2) provides the local discrete solution of harmonic equation in $\mathbb{E}^n$. Then $\alpha_m$ can be called as “harmonic” coordinates [6].

Such “harmonic” property of the non-Sibsonian interpolation [6] permits us to create the compact procedure for generation of higher-order $f$-interpolation in point $x_0$. For $k$th order interpolation it is sufficient that value $f$ satisfies to equation $\Delta^k f = 0$. Then we can calculate subsequently for $x_0$-neighbors the values $g_1, g_2, \ldots, g_{m-1}$ in accordance with formulas $\Delta f = g_1, \Delta g_1 = g_2, \Delta g_2 = g_3, \ldots, \Delta g_{k-1} = g_k = 0$. Further we calculate $f_0$ going through the above sequence of equations in reverse order.

In particular, for $k = 2$ we obtain two equations: $\Delta f = G$, $\Delta G = 0$ and, hence the final formulas for calculations of the second-order interpolation have the following form:

$$j \neq 0: \quad G_j = \frac{1}{V_j} \sum_{m=1}^{M(j)} \frac{(f_m - f_j)}{h_{m,j}} s_{m,j}, \quad (7a)$$

$$j = 0: \quad G_0 = \sum_{m=1}^{M} \alpha_m G_m, \quad (7b)$$

$$f_0 = \sum_{m=1}^{M} \alpha_m f_m - G_0 V_0 / \left( \sum_{m=1}^{M} s_m / h_m \right). \quad (7c)$$

Here, $M = M(0)$ is number of the Dirichlet neighbors only for point $x_0$: $M(j)$, $j \neq 0$, is number of neighbors for point $x_j$: $V_j$ is the areas (volumes) of the corresponding Dirichlet cells.

For derivation of Eq. (7) we used the following equality in three different forms (continuous, integral and discrete). Continuous form is

$$\oint_{S} \left( \frac{\partial A}{\partial \mathbf{n}} \right) dS = \iint B dV.$$ 

Integral form is

$$\oint_{S} \left( \frac{\partial A}{\partial \mathbf{n}} \right) dS = \iiint B dV.$$ 

Discrete form is

$$\sum_{p=1}^{P} \frac{(A_p - A_q)}{h_p} s_p = B_q V_q.$$ 

Eq. (7) can be obtained from these formulas for concrete functions $A$, $B$ and numbers $P$, $q$. 
Remark. The above presented formulas for higher-order interpolation are not general and are derived in assumptions of sufficiently smooth variations of geometric parameters of the Voronoi–Dirichlet tessellation, for the case when $E^n$ has almost uniform tessellation. The more general case is in investigation.

4. Comparison of interpolation algorithms

We present the results of a comparative numerical analysis of the following three methods of interpolation: the Delaunay triangulation (the diagrams are marked by $D$), the Sibson interpolation (marked by $S$), and the new non-Sibsonian interpolation (marked by $nS$). For the triangulation applied, the decomposition was performed once in the beginning and was not changed further. During interpolation, we found the triangle where $x_0$ was located and then performed the corresponding linear interpolation. When performing interpolation based on Dirichlet cells, we always calculated the new Dirichlet–Voronoi tessellation as the point $x_0$ was added or moved. Some examples of networks from the Dirichlet cells are shown in Fig. 1.

We present below the results obtained for functions that have significantly different values at the nodes. In this case, the difference between methods is more visible.

Fig. 2 shows some of the typical graphs of an interpolated function prescribed at the nodes in Fig. 1(b) along the horizontal dashed line, see tests [27], in accordance with the three methods of interpolation. We used a step function as a basic function. Namely, we set the function equal to zero for $x < 10$ and to unity for $x > 10$ on the network shown in Fig. 2. Fig. 2 demonstrates that $nS$ and $S$ results in smoother reconstructions than that obtained with $D$. Moreover, these methods provide continuous dependence on the initial data. Note that we have obtained slightly smoother results with $S$ as compared to $nS$.

Fig. 3 shows a fragment of the function recovered along a vertical section indicated by the dashed line in Fig. 2. We see that $nS$ and $S$ provide better smoothness than that obtained with $D$, as well as continuous dependence on the initial data. Note that the results obtained with $D$ are characterized by

Fig. 2. Graphs of an interpolated step type function $f$ along horizontal dashed line in Fig. 1(b). (b) is upper fragment of (a).
Fig. 3. Graphs of an interpolated step type function $f$ along vertical dashed line in Fig. 1(b).

Fig. 4. Graphs of an interpolation error $E$ as a function of nodes numbers $N$. (a) graphs for smooth function, (b) for non-smooth step type function.

a considerable difference in interpolation over adjacent points for small displacement of nodes. Similar results were obtained in other cases of isotropic distribution of nodes [2].

The development of the new method $nS$ was directly motivated by the complexity of implementation of $S$, especially in the three-dimensional case. Indeed, implementation of $S$ in $\mathbb{R}^n$ requires calculation of complex $n$-dimensional volumes formed by the intersecting Dirichlet cells. This is the main difference between algorithms $nS$ and $S$, which are rather similar in terms of interpolation results. We can only note that the results obtained with $S$ are slightly smoother than those obtained with $nS$. This may be explained by the fact that $S$ deals with $n$-dimensional volumes, while $nS$ deals with $(n - 1)$-dimensional volumes divided by a linear dimension.

Fig. 4 illustrates an investigation of convergence rate of three interpolation methods. We calculate an interpolation error $E$ as integral difference between an exact and interpolated values of function defined on $N$ random points, $20 \leq N \leq 10240 = 20 \cdot 2^9$ in unit square. Fig. 4 shows graphs of $E$ as a function
of $N$ and, hence, as a function on average distance among points. Fig. 4(a) is the $E$ graph for smooth function $f = \sin(xy)$, Fig. 4(b) is $E$ graph for non-smooth step type function: $f = 0$ for $x \leq y$ and $f = 1$ for $x > y$. All interpolation methods show first order convergence.

Another illustrations of interpolation investigation are done in [2]. In particular, [2] contains the illustrative comparison of interpolant surfaces (base shapes or influence function) for all methods of interpolation.

The comparative analysis of the proposed interpolation method shows that, in terms of smoothness, it can be rated as intermediate between the linear interpolation based on the Delaunay triangulation and the Sibson interpolation, whereas its properties are closer to the latter. In contrast to the triangulation method, the proposed non-Sibsonian interpolation is unique; at the same time, it is simpler than the Sibson interpolation and is faster in practical realizations. Computational experiments confirmed its practical first-order convergence. On the basis of the proposed interpolation the higher-order interpolations can be constructed in accordance with the Sibsonian approach [15] or by using the specific non-Sibsonian “harmonic” properties (see [11, Eq. (7)]).

5. Adaptive isoline algorithm

As example of the application of the non-Sibsonian interpolation we consider the new algorithm for plane isolines construction. This algorithm is based on the special “adaptive” generation and has the separate interest.

First time we used for isolines generation some standard numerical codes. But in some cases for the shallow water modeling by method [4,5,7] with real complex-shaped bottom relief we observed the anomalous behaviour of water depth isolines. There are some isolines intersections and non-stability. This irregular behaviour have not any physical sense and caused by analysis of non-uniqueness in numerical codes. The creation of a new isolines generation is an attempt to remove such irregularities.

Our algorithm is based on the Delaunay triangles and the non-Sibsonian interpolation. The values of function are prescribed to all triangles nodes. The usage of triangles permits us to interpolate the values by monotone linear function and provides the uniqueness on isolines course. Moreover, the influence of non-uniqueness of the Delaunay triangulation is reduced by the special adaptive procedure based on the non-Sibsonian interpolation.

In essence, the isolines algorithm use the triangulation on system of nodes extended in accordance with the following rule. For the initial Delaunay triangulation and for each two points $x_1$ and $x_2$ connected by triangle edge we control the following inequality in the point $x_{12} = \frac{1}{2} (x_1 + x_2)$:

$$\left| f_D(x_{12}) - f_{nS}(x_{12}) \right| \leq \varepsilon \max f - \min f.$$  \hspace{1cm} (8)

where $f_D(x_{12}) = \frac{1}{2} \left[ f(x_1) + f(x_2) \right]$ is the value of function in point $x_{12}$ in accordance with the Delaunay interpolation; $f_{nS}(x_{12})$ is the function value in point $x_{12}$ in accordance with the non-Sibsonian interpolation; $\varepsilon$ is the accuracy parameter of tessellation difference, where $\varepsilon = 0.01–0.05$. If inequality (8) is not realized then we add the point $x_{12}$ to initial system of points. The value of function $f$ in point $x_{12}$ is equal to $f_{nS}(x_{12})$. After global checking of inequality (8) for all points we obtain the new extended points system. Further, we generate for these points the new Delaunay triangulation and repeat the checking of condition (8) anew. For final stage we obtain the Delaunay triangulation for extended system of points with the following properties. In spite of probable non-uniqueness of the function values field for this
Delaunay triangulation, these values are approximated with $\varepsilon$-accuracy by the unique non-Sibsonian interpolation. Thus, the field of function values which can be reconstructed by the Delaunay interpolation is practically unique for sufficiently small value of $\varepsilon$. Secondly, the Delaunay triangle tessellation permit us to generate isolines without any anomalous behaviour. At third, this tessellation has automatically adaptive character. Therefore number of points increases in the region of large gradients of function. It permits us to automatically raise accuracy of isolines and to conserve the numerical information on function structure and behaviour.

Fig. 5(a) shows the $10 \times 10$ points grid in which nodes are defined the values of some test function with sufficiently non-simple behaviour. Fig. 5(b) shows 10-isolines picture for this function. These isolines are generated by the Delaunay triangulation from Fig. 5(a). The isolines field is unique, but its quality is not satisfactory: the isolines do not demonstrate the adequate function behaviour.

Fig. 6 shows the extended points system in accordance with inequality (8) for $\varepsilon = 0.01$. The complete number of these grid points is equal to 1087.

Figs. 7(a), (b) are the 10-isolines and 20-isolines fields correspondingly, which completely image the function structure. To achieve such accuracy on uniform grid it required more than 10000 points. Hence, the proposed algorithm permits us to raise economically the accuracy of isolines representation without any lines intersections, permits us to obtain isolines practically unique and without losses of numerical behaviour. Such algorithm can add another known algorithms.

For correct comparison isolines fields for different grids we show the 10-isolines picture, Fig. 8(b), for $37 \times 37$ extended uniform points grid, Fig. 8(a) (initial grid see in Fig. 5(a)). The interpolation
Fig. 6. Extended system of points for the Delaunay triangulation.

Fig. 7. Isolines picture for function $f$ defining on extended system of points from Fig. 6. (a) 10-isolines picture, (b) 20-isolines picture.
is calculated in accordance with the non-Sibsonian interpolation. Both isolines fields (Fig. 8(b) and Fig. 7(a)) are unique, but isolines on adaptive system of points resolve more adequate all singularities of investigated function.

Fig. 9 shows isolines fields for water depth in numerical modeling of natural flows in the framework of the shallow water equations [4,5,7]. Upper figure corresponds to the upper grids from Fig. 10, the lower figure corresponds to adaptive points grid from lower grid on Fig. 10. The isolines based on adaptive system of points resolve more fine characteristical singularities of function under graphical presentation.

We used for adaptive isolines generation also the second order non-Sibson interpolation in accordance with compact formulas (7) and higher-order interpolations. For water fields under investigation the results have not sufficient difference. We used also instead the non-Sibsonian interpolation the Sibson interpolation. The algorithm based on $nS$-interpolation is more fast than one based on $S$-interpolation. Another differences in our tests were not practically essential.

6. Summary

From the preceding results, we can conclude the following:

- The non-Sibsonian interpolation—a new method of first-order interpolation of the function values on the set of arbitrary points in a finite-dimensional Euclidean space $\mathbb{E}^n$, is robust and economical.
- The interpolation proposed is unique; moreover, it is simple and computationally faster than the Sibson interpolation.
- The specific “harmonic” property of the non-Sibsonian interpolation permits to construct the compact algorithms of the higher-order function interpolation.
The proposed new method of interpolation can be used in many applied calculations and can be recommended for a number of practical applications. In particular, it can be used for higher-order function interpolation. It can be used instead the Sibson interpolation in the Natural Element Method [28], etc. As example of such application, the fast effective algorithm of isolines generation based on the non-Sibsonian and the Delaunay interpolations and without any isolines intersections and without losses fine structure of data field is presented.

**Acknowledgements**

The authors are grateful to Drs. V.D. Ivanov, V.K. Kantorovich and S.A. Korytnik for assistance in providing of interpolation numerical testing.
Fig. 10. System of points with the Delaunay triangulations for isolines generation. Lower grid is extended system of points adapted to water depth values.

References


