40. Formal signs determination of efficiency assessment indicators for the operation with the distributed parameters / Lutsenko I., Fomovskaya E., Oksanych I., Vikhrova E., Serdiuk O. // Eastern-European Journal of Enterprise Technologies. 2017. Vol. 1, Issue 4 (85). P. 24–30. doi: 10.15587/1729-4061.2017.91025
41. Development of a verification method of estimated indicators for their use as an optimization criterion / Lutsenko I., Fomovskaya E., Oksanych I., Koval S., Serdiuk O. // Eastern-European Journal of Enterprise Technologies. 2017. Vol. 2, Issue 4 (86). P. 17–23. doi: 10.15587/1729-4061.2017.95914
42. Development of test operations in different duration to increase verification quality of efficiency formula / Lutsenko I., Fomovskaya E., Vihrova E., Serdiuk O., Fomovsky F. // Eastern-European Journal of Enterprise Technologies. 2018. Vol. 1, Issue 4 (91). P. 42–49. doi: 10.15587/1729-4061.2018.121810
43. Development of the method of modeling operational processes for tasks relating to decision-making / Lutsenko I., Oksanych I., Shevchenko I., Karabut N. // Eastern-European Journal of Enterprise Technologies. 2018. Vol. 2, Issue 4 (92). P. 26–32. doi: 10.15587/1729-4061.2018.126446

UDC 519.6: 004.942
DOI: 10.15587/1729-4061.2018.130787

OPTIMIZED SMOOTHING OF DISCRETE MODELS OF THE IMPLICITLY DEFINED GEOMETRICAL OBJECTS' SURFACES

S. Choporov
PhD, Associate Professor*
E-mail: s.choporoff@znu.edu.ua

S. Homeniuk
Doctor of Technical Sciences, Professor*
E-mail: serega@znu.edu.ua

S. Grebenyuk
Doctor of Technical Sciences, Associate Professor, Head of Department**
E-mail: gsm1212@ukr.net

*Department of Software Engineering***
Department of Fundamental Mathematics*
***Zaporizhzhia National University
Zhukovskoho str., 66, Zaporizhzhia, Ukraine, 69600

При використанні багатьох сучасних методів автоматичної генерації дискретних моделей поверхонь неявно заданих геометричних об'єктів втрачається точність апроксимації в околах особливостей (отворів, зламів тощо). Для покращення дискретних моделей геометричних об'єктів використовують різні методи згладжування. Існуючі методи згладжування проблеми спрямовані на трикутні елементи, але менш дослідженою є оптимізації дискретних моделей поверхонь геометричних об'єктів на базі елементів іншої форми (наприклад, чотирикутників).

Запропоновано математичний апарат, заснований на використанні енергетичного функціоналу для кожного вузла моделі. Запропонований функціонал враховує відстань від поточного вузла до суміжних і дистанцію від геометричних центрів інцидентних елементів до поверхні.

Розроблено алгоритм мінімізації енергетичного функціоналу при згладжуванні дискретних моделей поверхонь неявно заданих геометричних об'єктів. Розроблений алгоритм є модифікацію метода Гауса на випадок пошуку мінімуму в локальних координатах багатокутника, утвореного сусідніми елементами. Алгоритм є локальним: мінімізація виконується послідовно для кожного вузла моделі, тому багатократне його застосування дозволяє отримати моделі з більш точною апроксимацією поверхні.

Розроблений алгоритм мінімізації функціоналу не потребує додавання нових вузлів. Як наслідок, можливо використовуючи єдину процедуру оптимізувати дискретні моделі поверхонь на базі трикутників, чотирикутників або мішаного типу (що містять трикутники і чотирикутники одночасно). У результаті підвищується точність апроксимації поверхонь в околах особливостей, що показано на прикладах згладжування моделей складних об'єктів

Ключові слова: геометричний об'єкт, дискретна модель, неявна функція, згладжування, поверхня, енергетичний функціонал

1. Introduction

Modeling of many natural and technical objects is associated with the need to describe their shape in the form of mathematical relations. A very common way of determining the set of points belonging to some object is to use implicit mathematical functions.

It is assumed that a set of points Ω is defined implicitly if a logical predicate S(*P*) such that Ω={*P*: S(*P*)=true} is defined for each point *P*=(*x*, *y*, *z*). The simplest form of such a predicate is the restriction to the sign of some real function in the form of the inequality F(*P*)≥0. For example, the function $\mathrm{sphere}(x, y, z)=r^2-x^2-y^2-z^2$ is greater than zero at the points of the domain limited by a sphere of radius *r* centered at the origin, equal to zero on the boundary of this domain and less than zero at exterior points.

Implicit functions can be constructed step-by-step using set-theoretic operations (negation, disjunction, conjunction). Such operations are implemented in the form of systems of R-functions [1, 2]. In most cases, a system of R-functions is used in practice to construct F [2]

$$\begin{cases} \neg f_1 \equiv -f_1, \\ f_1 \vee f_2 \equiv f_1 + f_2 + \sqrt{f_1^2 + f_2^2}, \\ f_1 \wedge f_2 \equiv f_1 + f_2 - \sqrt{f_1^2 + f_2^2}, \end{cases} \qquad (1)$$

where f_1 and f_2 are the values of implicit functions.

Thus, the implicit function corresponding to a body in the shape of a sphere of radius 1.0 centered at the origin with a circular cylindrical hole of radius 0.5 with generatrices parallel to Oz (Fig. 1) can be defined by the formula

$$w_1(x,y,z) = \left(1 - x^2 - y^2 - z^2\right) \wedge \neg\left(0{,}25 - x^2 - y^2\right). \qquad (2)$$

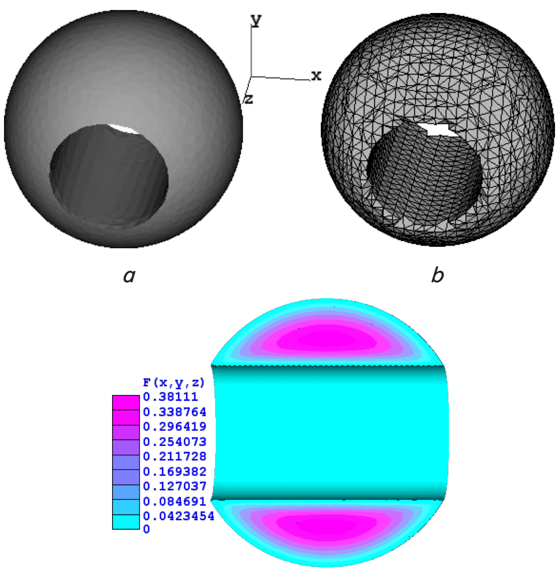


Fig. 1. Sphere with a circular cylindrical hole: *a* – general view; *b* – surface constructed by the marching cubes method; *c* – distribution of the values of the function (2) in the plane section x=0

At the same time, in the problems of computer graphics and numerical analysis, it becomes necessary to construct meshes of the boundaries of the objects under study. The use of existing methods [3–5] for generating meshes of the boundaries of implicit functions can lead to an inaccurate approximation of the boundary near features (holes, breaks, etc.). The increase in the number of nodes and elements in such areas allows achieving acceptable results in terms of visualization, but leads to a significant growth in the number of unknowns if the models are used in numerical analysis. As a result, it is important to develop methods for improving meshes of the boundaries of geometric objects that do not require an increase in the number of nodes.

2. Literature review and problem statement

The marching cubes method [3] is one of the first and at the same time very popular algorithms for triangulation of the boundaries of geometric objects defined by implicit functions. This method uses a regular grid, at the tops of cells of which the values of the implicit function define a set of triangles (one of 16 possible templates) approximating the section of the boundary. Its main disadvantages are topological ambiguity, the presence of triangles with very sharp angles and rough approximation of features (Fig. 1, *b*). To eliminate cases of topological ambiguity of the marching cubes method, it has been proposed in [4] to use bilinear interpolation and the number of templates has been increased to 33. In [5, 6], approaches to improving the results of the marching cubes method using dual meshes [5] or metrics [6] have been proposed. These approaches consider the deviation of the nodes from the surface and normals of triangles from the gradients of the implicit function.

In [7], in order to optimize the initial triangulation, it has been proposed to minimize the energy functional when searching for the positions of nodes and decreasing their number. In [8], it has been suggested to use implicit surface interpolation together with the energy functional.

A more accurate representation of singularities can be obtained using adaptive meshes. In [9], it has been proposed to use hierarchical structures (octrees) to increase the number of nodes in regions with the greatest surface curvature. The use of hierarchical structures for mesh improvement in the adaptive finite element analysis has been shown in [10]. In [11], features of implementation of hierarchical structures to improve models using a graphic processor have been considered.

An alternative to scanning a region with volume elements (hexahedra or tetrahedra) is scanning of the surface with flat elements. Thus, in the marching triangles method, the surface is filled with triangles, starting from an arbitrary point of it. In [12], schemes for scanning implicit surfaces by triangles, which allow generating adaptive meshes have been given.

In [13], it has been proposed to use the Delaunay criterion with constraints in the generation or improvement of the coordinates of the vertices of triangles on the surface. In [14], the scheme with the Delaunay criterion has been applied for adaptive triangulation of surfaces with allowance for materials (composite objects).

In [15], an approach to the search for surface singularities based on segmentation and clustering, when optimizing the meshes of triangular elements has been proposed. To solve a similar problem, it has been proposed in [16] to use semantic optimization.

The analysis of the results of the above studies allows concluding that the existing methods solve the problem of mesh improvement mainly in relation to triangles. The approaches and methods proposed in [6–10, 13, 14] require the insertion of new nodes into a mesh, which makes it practically impossible to use them for quadrilaterals. The use of hierarchical structures according to the scheme proposed in [11] allows achieving an acceptable quality. However, this will significantly increase the number of nodes and elements, which will lead to an increase in the dimensionality of the problem (for example, in applications to finite element analysis). Segmentation and clustering [14, 15] allow identifying areas with geometric singularities, but also require the insertion of additional nodes. Thus, effective approaches to the improvement of meshes based on quadrangles have not been identified in the works published at the moment.

3. The aim and objectives of the study

The aim of the paper is to develop an approach to improving meshes of implicitly defined geometric objects. This will allow improving the quality of approximation of singularities of geometric objects. In the final result, the accuracy of visualization and numerical solutions based on meshes will increase.

To achieve the aim, it is necessary to accomplish the following objectives:

– to develop a functional criterion for selecting the most suitable coordinates of nodes on the surface;

– to develop an algorithm of improving surface meshes of geometric objects defined by implicit functions.

4. Approach to improving surface meshes of implicitly defined geometric objects

Let some three-dimensional geometric object Ω be defined by the implicit function F(x, y, z), which is greater than zero at interior points of Ω, equal to zero on the boundary of Ω and less than zero at exterior points of Ω. Suppose that the mesh **M** is constructed for the boundary of Ω. Intuitively, the mesh of the boundary of a three-dimensional object is a piecewise linear surface consisting of polygonal faces connected along their edges. Formally, the surface mesh is a pair of sets of the form

$$\mathbf{M}=(\mathbf{V},\mathbf{E}), \quad (3)$$

where

$$\mathbf{V}=\{v_1=(x_1, y_1, z_1), v_2=(x_2, y_2, z_2),..., v_m=(x_m, y_m, z_m)\}$$

is the set of node coordinates;

$$\mathbf{E}=\{e_1=(k_{1,1},..., k_{1,n}), e_2=(k_{2,1},..., k_{2,n}),..., e_q=(k_{q,1},..., k_{q,n})\}$$

is the set n-gonal ($n\geq3$) elements – tuples of node numbers from the set **V** determining the position of the vertices of the element; m is the number of nodes; q is the number of elements.

Further, the following notations are used: **M**[k] – the radius vector to the node with the number k from the set V of the mesh **M**; **M**.adjacent(k) – the set of node numbers from **V** that have a common edge with $v_k\in\mathbf{V}$; **M**.incident(k) – the set of numbers of elements from **E** incident at the node v_k; **M**(i) – the tuple with the number i (element $e_i\in\mathbf{E}$); **M**(i, j) – the node number in the j-th vertex of the element e_i.

4. 1. Distance-length functional

One of the most widely used and at the same time simple methods of mesh improvement is local Laplacian smoothing [10]. It moves each node to the geometric center of the polygon formed by adjacent nodes (Fig. 2). This smoothing is traditionally applied several times.

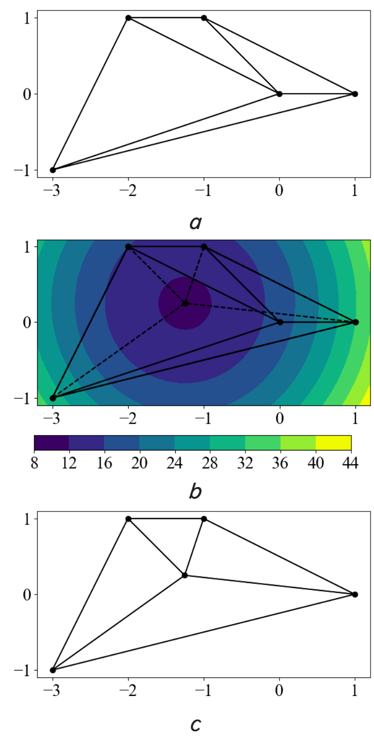


Fig. 2. Laplacian smoothing for a fragment of a plane mesh of triangles: a –original mesh; b – distribution of the values of the function (5); c – the result of smoothing

In Laplacian smoothing, each edge connected to the central node can be considered as a spring with initial zero length [7]. The sum of the forces acting at the i-th node of the mesh **M** is defined by the formula

$$\mathbf{F}_i=\kappa\sum_{k\in\mathbf{A}}\left(\mathbf{M}[k]-\mathbf{M}[i]\right), \quad (4)$$

where κ is the spring stiffness; **A**=**M**.adjacent(i).

The potential energy of the spring system

$$\mathbf{E}_i=\frac{\kappa}{2}\sum_{k\in\mathbf{A}}\left\|\mathbf{M}[k]-\mathbf{M}[i]\right\|^2. \quad (5)$$

The equilibrium point of the spring system corresponds to a minimum of the potential energy.

The formula (5) in the three-dimensional case can be written in the form

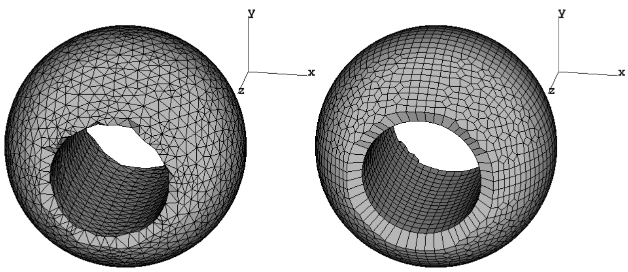
$$\mathbf{E}_i=\frac{\kappa}{2}\sum_{k\in\mathbf{A}}\left((x_k-x_i)^2+(y_k-y_i)^2+(z_k-z_i)^2\right). \quad (6)$$

By differentiating and equating to zero, we obtain

$$x_i=\frac{1}{|\mathbf{A}|}\sum_{k\in\mathbf{A}}x_k,\quad y_i=\frac{1}{|\mathbf{A}|}\sum_{k\in\mathbf{A}}y_k,\quad z_i=\frac{1}{|\mathbf{A}|}\sum_{k\in\mathbf{A}}z_k, \quad (7)$$

where |**A**| is the number of adjacent vertices (elements in the set **A**).

The result is a simple and very efficient procedure from a computing point of view. Its application for surface meshes requires a search of the projection of a new position of a point onto the boundary of the domain. Multiple iterative application of such smoothing tends to provide equal length to the sides of the elements. However, the lack of information about the accuracy of approximation leads to destroyed surface features (Fig. 3).



a *b*

Fig. 3. Result of Laplacian smoothing of surface meshes defined by the formula (2): *a* — triangle; *b* — quadrangles

We consider the energy function of the form

$$\mathbf{E}_i = \kappa\sum_{k\in\mathbf{A}}\left\|\mathbf{M}[k]-\mathbf{M}[i]\right\|^2 + $$
$$+\sum_{q\in\mathbf{I}}\left\|C_q - \text{projection}\left(C_q, \text{F}(x,y,z)\right)\right\|^2, \qquad (8)$$

where $\mathbf{I}$=$\mathbf{M}$.incident(i) is the set of numbers of elements incident at the node with the number i;

$$C_q = \sum_{j=1}^{n}\mathbf{M}\left[\mathbf{M}(q,j)\right]$$

is the center of mass of the element with the number q; projection(C_q, F(x, y, z)) is the projection of the point C_q onto the surface defined by the function F(x, y, z).

The first term in (8) guarantees the existence of a minimum [7]. The choice of the second term is made from the following assumption. Each flat element of a mesh can be considered as an approximation of a part of the surface (Fig. 4), and the distance from the center of the element to the surface is an indicator of the accuracy of such approximation.

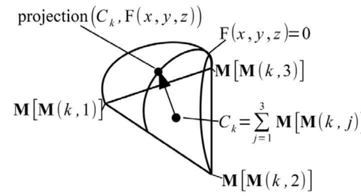


Fig. 4. Approximation of a part of the surface by a triangle

When calculating the value of the formula (8), the search for the projection of a point onto the surface can be carried out in the direction of a gradient (for exterior points) or anti-gradient (for interior points). The use of the side lengths of elements and distances from the centers of elements to the surface determines the name "distance-length functional".

4. 2. Algorithm of search for a minimum of the distance-length functional

It is impossible to find an analytic expression for the minimum point of the functional (8), as it is done in the formulas (6), (7), due to the implicitness and, as a rule, the nonlinearity of the function F(x, y, z). For a numerical search for a minimum, one can use a modification of the Gaussian method formalized in the form of the following algorithm.

algorithm local_minimum
  input:
    F(x, y, z) – the function representing a geometric object;
    $\mathbf{M}$ – the surface mesh, defined by the function F(x, y, z);
    i – the number of the node for optimization;
    κ – spring stiffness;
    ε – the accuracy of the method;
  output:
    $\mathbf{M}$ – surface mesh
  begin
    $\mathbf{A} \leftarrow \mathbf{M}.\text{adjacent}(i)$;
    $\mathbf{I} = \mathbf{M}.\text{incident}(i)$
    for each *k* in A do
      $P_0 \leftarrow \mathbf{M}[i]$;
      $l \leftarrow 0{,}1$;
      $f_0 \leftarrow \kappa\sum_{k\in\mathbf{A}}\left\|\mathbf{M}[k]-\mathbf{M}[i]\right\|^2 + \sum_{q\in\mathbf{I}}\left\|C_q - \text{projection}\left(C_q,\text{F}(x,y,z)\right)\right\|^2$,
      do
        $\mathbf{M}[i] \leftarrow \text{projection}\left((1-l)\mathbf{M}[i] + l\mathbf{M}[k], \text{F}(x,y,z)\right)$;
        $f_1 \leftarrow \kappa\sum_{k\in\mathbf{A}}\left\|\mathbf{M}[k]-\mathbf{M}[i]\right\|^2 + \sum_{q\in\mathbf{I}}\left\|C_q - \text{projection}\left(C_q,\text{F}(x,y,z)\right)\right\|^2$;
        if $f_0 \le f_1$ **then**
          $\mathbf{M}[i] \leftarrow P_0$;
          $h \leftarrow \frac{h}{2}$;
        else
          $P_0 \leftarrow \mathbf{M}[i]$;
        endif
      while $h>\varepsilon$
    endfor
  end

The search is carried out in the l-coordinates of the polygon formed by the nodes adjacent to the i-th nodes. Descent is performed on each of the l-coordinates, while the objective function decreases. An adaptive step is used: in each direction, starting with a step whose value is empirically assumed to be 0.1, if the objective function ceases to decrease, then the step is halved.

The application of the local algorithm to each node of a mesh can be regarded as an iterative approximation to the minimum of the functional (8) in the global statement. As a result, we obtain the following algorithm of mesh improvement.

algorithm boundary_refinement
 input:
  $F(x, y, z)$ – the function representing a geometric object;
  M – the surface mesh, defined by the function $F(x, y, z)$;
  κ – spring stiffness;
  ε – the accuracy of the method;
 output:
  M – surface mesh
 begin
  for each $i=\overline{1,|\mathbf{V}|}$ **do**
   local_minimum($F(x, y, z)$, **M**, i, κ, ε);
  endfor
 end

By analogy with Laplacian smoothing, the boundary_refinement algorithm should be applied several times. Moreover, methods of improving meshes based on topological changes (for example, flip [5, 14] for triangles) can additionally be used in each iteration.

5. Results of local minimization of the distance-length functional for surface mesh improvement

Consider the results of applying the local algorithm for improving the positions of nodes, based on minimizing the distance-length functional, in order to improve the surface mesh defined by the function (2). The original model is shown in Fig. 1, *b*. Fig. 5 shows the results of applying the proposed algorithm for $\kappa=10^{-3}$ and $\varepsilon=10^{-6}$. The color of the element shows the value of the smallest angle in degrees (it is assumed that the larger the minimum angle, the better the model). It can be noted that, starting with four iterations, the surface break near the hole is approximated by the nodes and edges of the elements, but at the same time sharp (less than 10 degrees) angles remain.

The proposed algorithm can also be used to improve the meshes of quadrilateral elements. Fig. 6, *a* shows the original mesh, as well as the results of applying the proposed algorithm for $\kappa=10^{-3}$ and $\varepsilon=10^{-6}$ (Fig. 6, *b*–*f*). The color of the element shows the value of the largest angle in degrees (in the case of quadrilaterals, the larger the maximum angle, the closer it is to the degenerate element). As in the case of triangles, starting with four iterations, the surface break near the hole is approximated by the nodes and edges of the elements, but the elements close in shape to triangles remain.

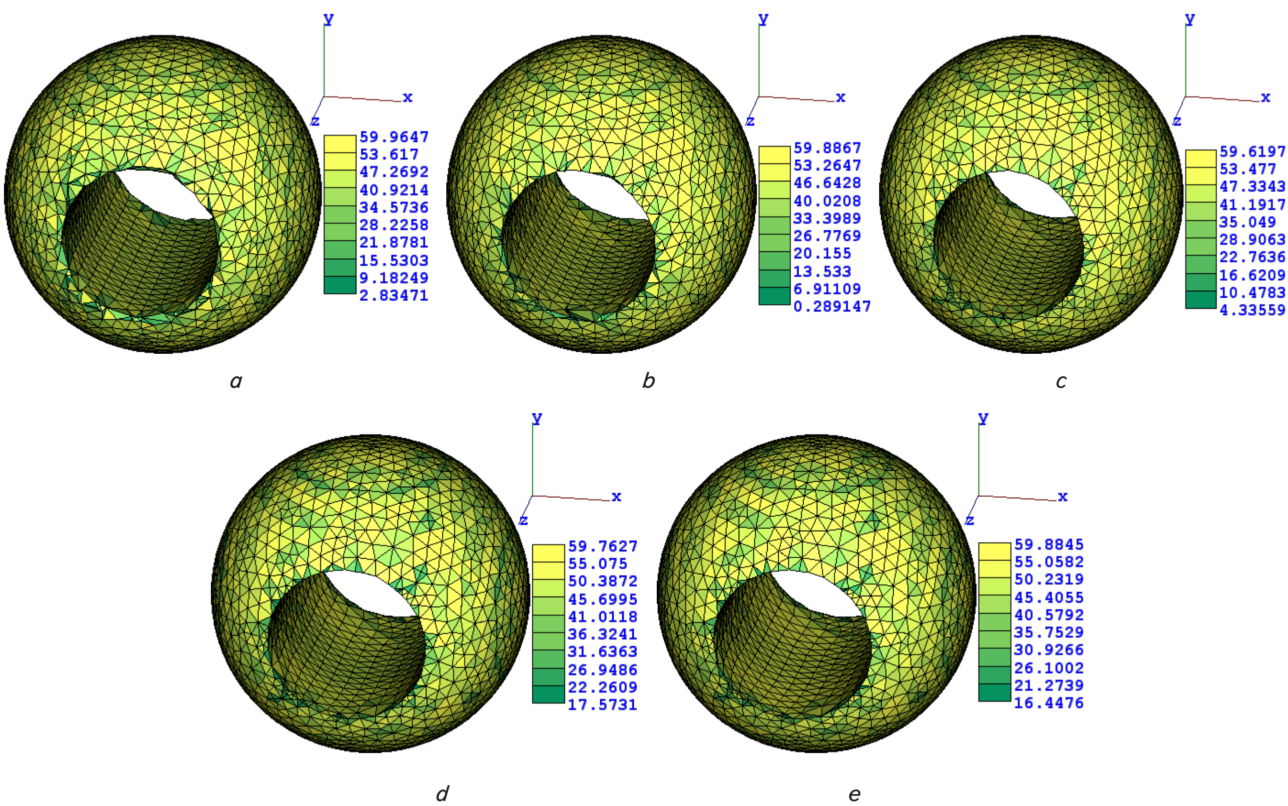


Fig. 5. Application of local minimization of the distance-length functional to improve the model of a sphere with a circular hole (triangles): *a* – 1 iteration; *b* – 2 iterations; *c* – 4 iterations; *d* – 8 iterations; *e* – 16 iterations

When investigating the effect of the parameter κ, the second term in the formula (8) can be regarded as the force of attraction of the centers of the elements to the surface. Then the stiffness κ determines the possibility of "resistance" of the system of springs between the nodes to the movement of the nodes on the surface. Accordingly, the larger κ, the closer the result of minimization to Laplacian smoothing, and with a decrease in κ, breaks will be approximated more accurately, but it is possible to obtain closely spaced nodes. Fig. 7 shows the results of varying κ for 8 iterations of mesh improvement.

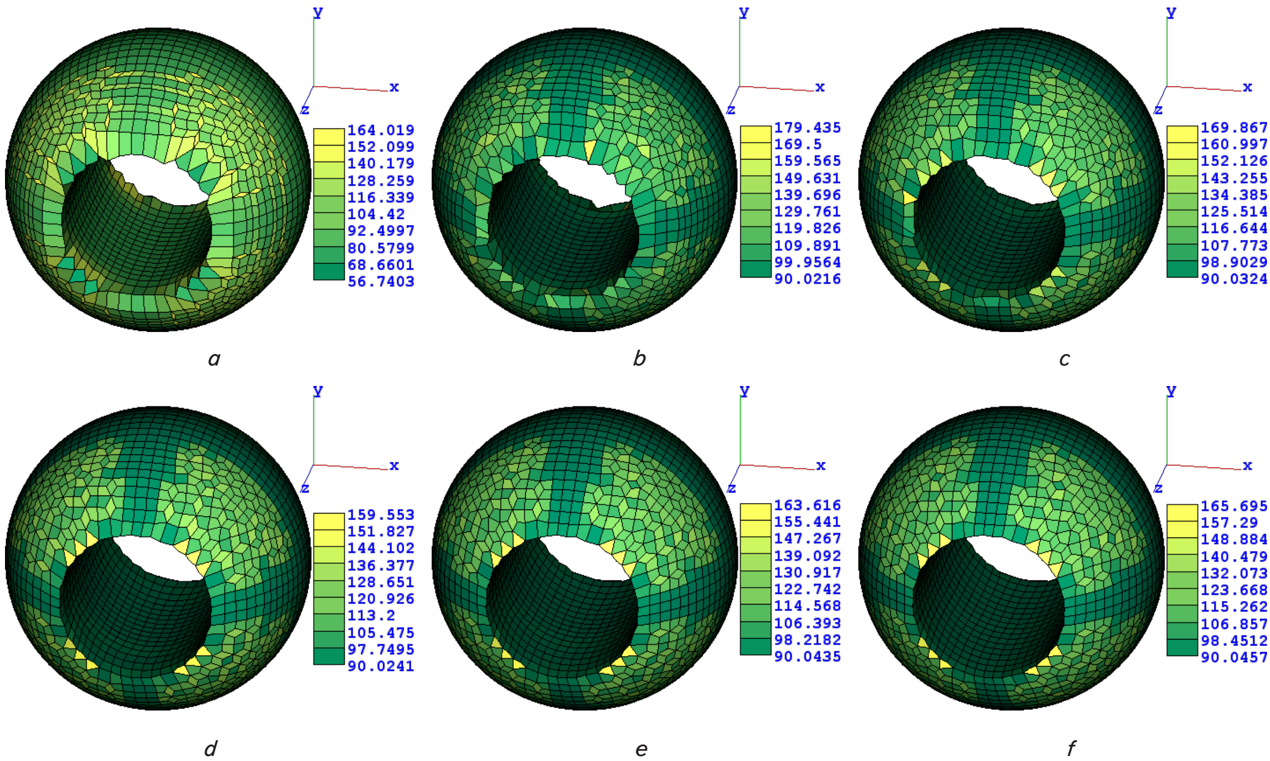


Fig. 6. Application of local minimization of the distance-length functional to improve the model of a ball with a circular hole (quadrilaterals): *a* – original mesh; *b* – 1 iteration; *c* – 2 iterations; *d* – 4 iterations; *e* – 8 iterations; *f* – 16 iterations

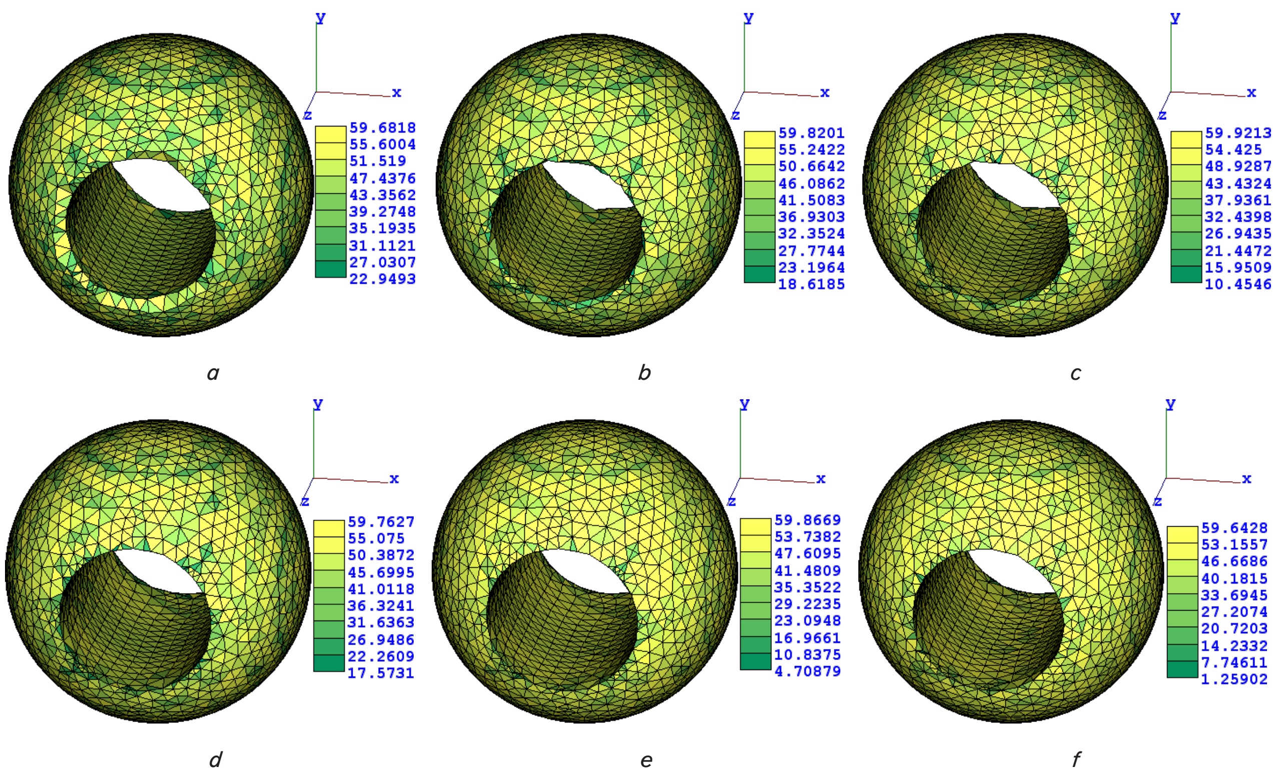


Fig. 7. Effect of the parameter κ: *a* – κ=0.1; *b* – κ=0.01; *c* – κ=0.005; *d* – κ=0.001; *e* – κ=0.0005; *f* – κ=0.0001

The function

$$\text{cube}(x,y,z)=\left(\left[\left(1-x^2\right)\wedge\left(1-y^2\right)\wedge\left(1-z^2\right)\right]\vee\left(0{,}25-x^2-z^2\right)\right)\wedge$$
$$\wedge\neg\left(0{,}25-x^2-y^2\right)\wedge\neg\left(0{,}25-y^2-z^2\right), \qquad (9)$$

corresponds to the model of a cube (Fig. 8) with two circular holes (generatrices parallel to *Ox* and *Oz*) and cylindrical handles with generatrices parallel to *Oy*. The side of the cube is 2.0; the radiuses of the holes and handles – 0.5; the length of the handles is 1.0.

Fig. 8, *a* shows the original mesh of triangular elements, as well as the results of applying the local minimization algorithm of the distance-length functional for $\kappa=10^{-3}$ and $\varepsilon=10^{-6}$ (Fig. 8, *b–d*). Similarly, Fig. 9 shows the results for improving the mesh of quadrilateral elements (Fig. 9, *a–d*).

The proposed algorithm can be applied to the improvement of models of smooth surfaces. For example, for the surface "genus3" [12], which can be defined by the formula

$$\text{genus3}(x,y,z)=-r_z^4 z^2+\left(1-\frac{x^2}{r_x^2}-\frac{y^2}{r_y^2}\right)\times$$
$$\times\left(\left(x-x_1\right)^2+y^2-r_1^2\right)\left(\left(x+x_1\right)^2+y^2-r_1^2\right)\left(x^2+y^2-r_1^2\right), \qquad (10)$$

with the parameter values

$$r_x=6{,}0,\;\; r_y=3{,}5,\;\; r_z=4,\;\; r_1=1{,}2 \;\text{ and }\; x_1=3{,}9,$$

the results of mesh improvement ($\kappa=10^{-3}$ and $\varepsilon=10^{-6}$) are shown in Fig. 10.

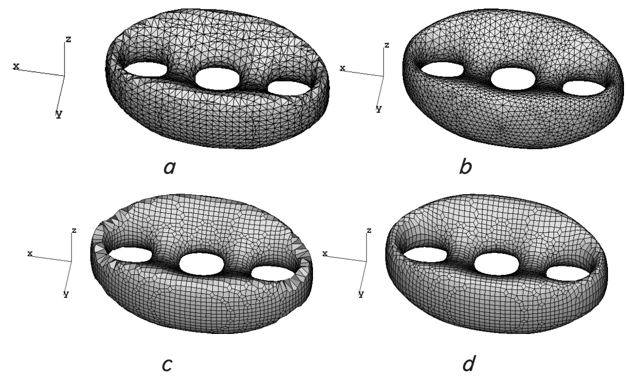


Fig. 10. Application of local minimization of the distance-length functional to improve the "genus3" model: *a* – original triangular mesh; *b* – 8 iterations of improving the triangular mesh; *c* – original quadrilateral mesh; *d* – 8 iterations of improving the quadrilateral mesh

Thus, the application of the proposed algorithm to smooth surface meshes will lead to an increase in the number of nodes in regions with the greatest curvature.

6. Discussion of the results of application of local minimization of the distance-length functional for surface mesh smoothing

The obtained results of modeling surfaces of geometric objects (Fig. 5–10), defined by implicit functions, show that the proposed approach to local minimization of the distance-length functional allows improving the accuracy of meshes near features (surface breaks).

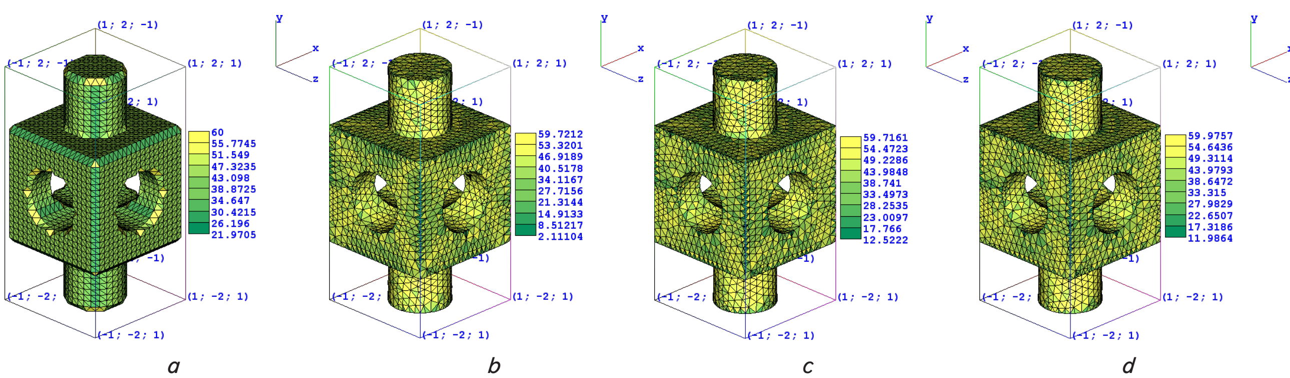


Fig. 8. Application of local minimization of the distance-length functional to improve the "cube" (triangles) model: *a* – original mesh; *b* – 4 iterations; *c* – 8 iterations; *d* – 16 iterations

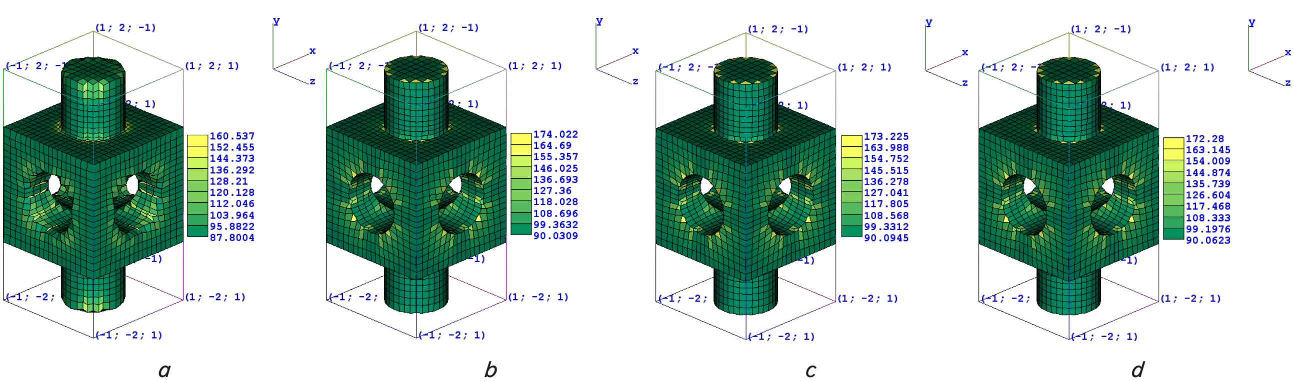


Fig. 9. Application of local minimization of the distance-length functional to improve the "cube" (quadrilateral) model: *a* – original mesh; *b* – 4 iterations; *c* – 8 iterations; *d* – 16 iterations

The accuracy of the approximation of features depends on the stiffness parameter of the edges of the elements of the original mesh, as shown in Fig. 7.

The advantage of the proposed approach is that (unlike the results of [6–10]) it can be applied to both triangles and quadrangles, and also does not require the insertion of additional nodes and elements.

It should be noted that this approach to mesh improvement does not consider quality metrics (minimum or maximum angles, aspect ratios). As a result, the possibility of obtaining models with very sharp or obtuse angles can be indicated as a drawback. Also, as can be seen from Fig. 6, 9, quadrangles similar in shape to triangles (one of the angles is greater than 160 degrees) can be obtained near surface breaks, which cannot be eliminated without additional topological modification.

The results of the work can be applied when implementing the module for mesh improvement in computer-aided engineering based on numerical methods. For example, in the finite and boundary element methods, a more accurate representation of the geometric singularities of objects allows improving the quality of the models.

The proposed approach can be used in common with the scheme of local smoothing proposed by the authors in [17] for internal nodes of meshes of objects. As a result, a single procedure for all nodes of the model of a three-dimensional object based on spatial elements (for example, tetrahedra or hexahedrons) can be formed. The prospect of further improvement is the consideration of metrics of the quality of elements while minimizing the functional. For example, the addition of algorithms for local topological modification of elements will allow controlling the values of angles and aspect ratios in elements.

7. Conclusions

1. The proposed approach to smoothing meshes of implicitly defined surfaces, based on local minimization of the distance-length functional, allows improving the accuracy of the approximation of singularities. At the same time, for the numerical implementation of the search for a minimum, a modification of the Gaussian method for the case of search in *l*-coordinates is proposed. As a result of applying the approach, minimum angles for triangles increase from 2 to 12–16 degrees, while the boundary singularities are more qualitatively represented.

2. For best results, the proposed approach should be applied consistently several times, due to the local nature of the minimum search procedure. The studies have shown that with a small number of iterations (less than 4), poor quality meshes are obtained. Starting with the fourth iteration, geometric singularities are already approximated by the nodes. The subsequent iterations improve the quality of the mesh by moving the nodes along singularities. When performing more than eight iterations, the results do not change significantly.

Acknowledgments

The materials of the paper were prepared within the framework of the state budget subject “The Development of a Software Resource for the Cloud-based Engineering Analysis of Aerospace Objects” (0117U007204) of the Ministry of Education and Science of Ukraine.

References

1. Maksimenko-Sheiko K. V. R-function in the mathematical modeling of the geometry and physical fields: monograph. Kharkiv: Institute of Problems of Mechanical Engineering of the NAS of Ukraine, 2009. 306 p.
2. R-functions and chevron surfaces in machine building / Sheiko T. I., Maksymenko-Sheiko K. V., Litvinova Yu. S., Lisin D. A. // Journal of mechanical engineering. 2017. Vol. 20, Issue 2. P. 54–60. URL: http://journals.uran.ua/jme/article/view/105801/101083
3. Lorensen W. E., Cline H. E. Marching Cubes: A High Resolution 3D Surface Construction Algorithm // Proceedings of the 14th annual conference on Computer graphics and interactive techniques – SIGGRAPH ‘87. 1987. Vol. 21, Issue 4. P. 163–169. doi: 10.1145/37401.37422
4. Chernyaev E. V. Marching Cubes 33: Construction of Topologically Correct Isosurfaces // Technical Report CERN CN 95-17. CERN. 1995. 8 p.
5. Ohtake Y., Belyaev A. Dual/Primal Mesh Optimization for Polygonized Implicit Surfaces // Proceedings of the seventh ACM symposium on Solid modeling and applications – SMA ‘02. 2002. P. 171–178. doi: 10.1145/566282.566308
6. Ohtake Y., Belyaev A., Pasko A. Dynamic Mesh Optimization for Polygonized Implicit Surfaces with Sharp Features // The Visual Computer. 2003. Vol. 19, Issue 2-3. P. 115–126.
7. Mesh Optimization / Hoppe H., DeRose T., Duchamp T., McDonald J., Stuetzle W. // Proceedings of the 20th annual conference on Computer graphics and interactive techniques – SIGGRAPH ‘93. 1993. P. 19–26. doi: 10.1145/166117.166119
8. Igarashi T., Hughes J. F. Smooth meshes for sketch-based freeform modeling // ACM SIGGRAPH 2006 Courses on – SIGGRAPH ‘06. 2006. doi: 10.1145/1185657.1185774
9. Miranda A. C. O., Martha L. F. Mesh Generation on High-Curvature Surfaces Based on a Background Quadtree Structure // Proceedings of 11th International Meshing Roundtable. Sandia National Laboratories, 2002. P. 333–342.
10. A Three-Dimensional Adaptive Mesh Generation Approach Using Geometric Modeling With Multi-Regions and Parametric Surfaces / Miranda A. C. O., Lira W. W. M., Cavalcante-Neto J. B., Sousa R. A., Martha L. F. // Journal of Computing and Information Science in Engineering. 2013. Vol. 13, Issue 2. P. 021002. doi: 10.1115/1.4024106
11. Boubekeur T., Schlick C. A Flexible Kernel for Adaptive Mesh Refinement on GPU // Computer Graphics Forum. 2008. Vol. 27, Issue 1. P. 102–113. doi: 10.1111/j.1467-8659.2007.01040.x
12. Cermak M., Skala V. Polygonisation of disjoint implicit surfaces by the adaptive edge spinning algorithm of implicit objects // International Journal of Computational Science and Engineering. 2007. Vol. 3, Issue 1. P. 45. doi: 10.1504/ijcse.2007.014464
13. Dey T. K., Levine J. A. Delaunay meshing of isosurfaces // The Visual Computer. 2008. Vol. 24, Issue 6. P. 411–422. doi: 10.1007/s00371-008-0224-1

14. Fayolle P.-A., Pasko A. Optimized surface discretization of functionally defined multi-material objects // Advances in Engineering Software. 2012. Vol. 45, Issue 1. P. 301–312. doi: 10.1016/j.advengsoft.2011.10.007
15. Huang H., Ascher U. Surface Mesh Smoothing, Regularization, and Feature Detection // SIAM Journal on Scientific Computing. 2008. Vol. 31, Issue 1. P. 74–93. doi: 10.1137/060676684
16. Multi-view Stereo with Single-View Semantic Mesh Refinement / Romanoni A., Ciccone M., Visin F., Matteucci M. // 2017 IEEE International Conference on Computer Vision Workshops (ICCVW). 2017. doi: 10.1109/iccvw.2017.89
17. Choporov S. V. Smoothing of Quadrilateral Meshes by Local Functional Minimization // Visnyk of Kherson National Technical University. 2017. Vol. 2, Issue 3 (62). P. 234–239.

Розглянуто задачу структуризації групових експертних оцінок, сформованих в умовах невизначеності різної природи і наявності конфліктуючих експертних свідоцтв. Запропоновано методику агрегування групових експертних оцінок, що формуються в умовах різних видів невизначеності, яка дозволяє синтезувати групове рішення з урахуванням різних форм представлення експертних переваг (інтервальні, нечіткі, точкові експертні оцінки). Запропонована процедура дозволяє синтезувати групове рішення у разі, якщо в групі експертів є група або декілька груп експертів, які висловлюють свої переваги з використанням різних форм подання експертної інформації.

Такий підхід дозволяє максимально точно відображати експертні переваги щодо аналізованого об'єкта, не обмежуючи експертів жорсткою формою подання оцінок.

Для аналізу отриманої експертної інформації, та отримання індивідуальних експертних ранжувань аналізованих об'єктів, в роботі використаний метод парних порівнянь і його модифікації.

Встановлено, що для агрегування точкових експертних оцінок, більш точні результати комбінування можуть бути отримані на основі застосування правил перерозподілу конфліктів теорії правдоподібних і парадоксальних міркувань. Для агрегування інтервальних експертних оцінок рекомендується застосовувати одне з правил комбінування теорії свідоцтв. Встановлено, що для підвищення якості результатів комбінування доцільно визначати порядок комбінування експертних свідоцтв, наприклад, враховуючи міру відмінності і структуру експертних свідоцтв.

Одержані результати покликані сприяти підвищенню якості та ефективності процесів підготовки і прийняття рішень щодо аналізу та структуризації групових експертних оцінок

Ключові слова: експертні оцінки, агрегування експертних оцінок, метод парних порівнянь, правила комбінування

UDC 004.82:519.234
DOI: 10.15587/1729-4061.2018.133299

DEVELOPMENT OF A TECHNOLOGY OF STRUCTURING GROUP EXPERT JUDGMENTS UNDER VARIOUS TYPES OF UNCERTAINTY

I. Kovalenko
Doctor of Technical Sciences, Professor*
E-mail: ihor.kovalenko@chmnu.edu.ua
A. Shved
PhD*
E-mail: avshved@chmnu.edu.ua
*Department of software engineering
Petro Mohyla Black Sea National University
68 Desantnykiv str., 10, Mykolaiv, Ukraine, 54003

1. Introduction

In general terms, the technology of structuring group expert judgments can be presented in the form of successive steps, Fig. 1.

A panel of experts $\mathrm{E}=\{E_j \mid j=\overline{1,t}\}$ is given one and the same set of options (objects of expertise, alternatives) $\mathrm{A}=\{A_i \mid i=\overline{1,m}\}$ and the same instruction containing information as to what type of scorecard in which the experts will express their preferences will be used. It depends on the type of information received from the experts (words, conditional gradations, numbers, rankings, breakdowns or other types of objects of non-numeric nature).

As a result, a plurality of individual expert judgments $\mathrm{O}=\{O_i \mid i=\overline{1,t}\}$ is formed. The established set of expert judgments (EJs) enters the block of structuring procedures to perform operations of ranking, clusterization, and others. The obtained data fall into the block of evaluating the results of structuring, which contains a set of conditions that determines the correctness of the structurization block.

When solving problems of analysing group expert judgments and choosing appropriate methods (Fig. 2), two important circumstances should be taken into account:

– availability of diverse scales of expert measurements and a large number of different forms of representing expert