Curvature Based Mesh Improvement.

IRINA SEMENOVA.
Department of Mechanical Sciences and Engineering
Tokyo Institute of Technology, Graduate School
Tokyo, 152-8552,
JAPAN

VLADIMIR SAVCHENKO.
Faculty of Computer and Information Sciences
Hosei University,
3-7-2 Kajino-cho Koganei-shi
Tokyo, 184-8584,
JAPAN

ICHIRO HAGIWARA
Department of Mechanical Sciences and Engineering
Tokyo Institute of Technology, Graduate School
Tokyo, 152-8552,
JAPAN
hagiwara@mech.titech.ac.jp

Abstract: It is very important to improve the quality of surface meshes for numerical simulations, solid mesh generation, and computer graphics applications. Optimizing the form of the mesh elements it is necessary to preserve new nodes of the mesh as close as possible to a surface approximated by the initial mesh. This paper proposes a novel technique in which both of the requirements to mesh improvement are implemented. In the method presented here the new location of each node is found using values of principal curvatures in this node. Such procedure allows preserving new mesh very close to the initial surface while improving element quality. The method has been successfully tested on triangular meshes both for analytical surfaces (sphere, ellipsoid, paraboloid) and for arbitrary surfaces with great number of points. Comparison of the deviation of the mesh optimized by our method and by Laplacian smoothing from the original analytical surfaces shows advantage of the proposed method.

Key-Words: surface meshes, quality improvement, principal curvatures.

1. Introduction

It is well known that mesh quality is critical for accuracy and efficiency in the numerical solutions to PDE-based applications [1]. For generation of good quality solid meshes it is also necessary to have good quality mesh. There are two main ways for mesh optimization: modification of the mesh topology by inserting/deleting mesh nodes or edge flipping [2, 3] and node movement methods, commonly called mesh smoothing. A number of smoothing techniques have been developed ranging from simple Laplacian smoothing [4] to more sophisticated algorithms. Among them there are physically-based methods [5, 6] where nodes are moved under the influence of some forces so that the shape of incident elements is improved. Instead of local mesh optimization by moving each node on the basis of some geometric characteristics (as is done in Laplacian smoothing and in the physically-based methods) the optimization-based techniques allow improving all initial mesh. In these techniques so called cost function of angle [7] is optimized, an aspect ratio [8] or distortion metrics [9, 10] can be used.
It is necessary to note that the procedures for the denoising of the initial mesh are often called smoothing too. Quite a lot of work has been done around the subject, see [11] by Belyaev and Ohtake and references therein. In this paper we do not focus our attention on the problem of mesh denoising. Further we suppose that there are initial meshes without any noise.

Good shape of mesh elements is not only the criteria of mesh quality. It is also important to preserve the new nodes as close as possible to the smooth surface approximated by the initial mesh. Preservation of the nodes closed to original surface allows keeping important discrete surface characteristics such as normals and curvatures. Conservation these surface characteristics prevents drastic changes in forces such as surface tension. For large deformation of free boundaries in metal forming and interfaces in multi-material gas dynamics it is critical to keep deviation these characteristics for the optimized mesh from characteristics for the initial mesh as small as possible.

The problem of mesh optimization with preservation of the discrete surface characteristics has been investigated Garimella et. al [12]. In their method the nodes are repositioned in a series of local parametric spaces derived from individual mesh elements. When the repositioned nodes are mapped back to the real space, each node lays on the corresponding mesh element.

Note, however, that new nodes lie on the initial mesh but not on the surface approximated by this mesh. As a simple example consider a mesh on the sphere. The nodes of this mesh lie on the sphere. Applying algorithm described above we will get new nodes situating on the initial mesh, but they will not belong to the original sphere. Therefore obtained mesh will not be discrete approximation of the concerned sphere.

Thus, we can formulate the problem of mesh improvement in the following way. A triangular mesh which serves as a representation (discrete approximation) of a smooth surface is given. It is necessary to improve mesh element quality in such a way that new nodes keep situated on the approximated smooth surface.

In this paper we propose a novel technique called Curvature Based Mesh Improvement (CBMI) which effectively solves the formulated problem. The new position of each node is found using principal curvatures calculated for each node. Such procedure allows keeping vertices of the new mesh very close to approximated smooth surface. For analytical surfaces quantitative measures are presented to prove that the deviation of the new nodes from the approximated smooth surface is small.

The rest of the paper is organized as follows. In section 3 curvature estimation is described. CBMI algorithm is presented in section 4. Section 5 describes the results of a testing CBMI algorithm for meshes on the different surfaces. The paper concludes in section 6 with an overview of the algorithm and discussions of the future work.

2. Curvature estimation

This section describes how to calculate the principal curvatures in each node of the initial mesh. Since surface curvature calculation is based on the second order derivatives, it is common to use quadratic polynomials to approximate the surface locally. The presented work uses functions of the form $z = f(x, y)$ (Monge form). The quadric of this form is fitted to the nodes in a local coordinate system (x, y, z) whose z axis is along a normal at the concerned vertex and whose origin is at that vertex. In the simplest cases it is performed a least squares fit of the quadric $z = ax^2 + bxy + cy^2$. Addition linear terms in the quadric allows increasing accuracy of curvature estimation. It is also possible to use a full quadric in which the addition of the constant term allows the surface do not pass through the concerned vertex. The quadric is therefore given by: $z = f(x, y) = ax^2 + bxy + cy^2 + dx + ey + f$.

There are a number other techniques for curvature estimation, for more references and details see [13].

2.1 Construction of the quadric

Here, we shall give a short account of the quadric construction technique used in the application considered in this paper.

It is necessary to determine normals at the nodes of the initial mesh. For that we calculate normals for each triangle and then average them at shared points.

For quadric interpolation we use least squares method, whose detailed description may be found in [14]. Here we only describe main steps. For each node p_j of the initial mesh:

- Get the k nearest neighboring nodes.
- Compute the tangent plane P at concerned node (i.e. the plane perpendicular to the normal n_j at p_j).
- Define an orthonormal coordinate system (x, y) in P with a node p_j as an origin and take the normal n_j as a z axis.
- Find the coordinates (x_i, y_i, z_i), $i = 1,...,k$ of the k nearest neighboring nodes in the new coordinate system.

- Form the system: $AX = B$, where $A = \begin{pmatrix} x_1^2 & x_1 y_1 & \dots & 1 \\ x_2^2 & x_2 y_2 & \dots & 1 \\ \dots & \dots & \dots & \dots \\ x_k^2 & x_k y_k & \dots & 1 \end{pmatrix}$, $B = \begin{pmatrix} z_1 \\ z_2 \\ \dots \\ z_k \end{pmatrix}$
- $X = \begin{pmatrix} a \\ b \\ \dots \\ f \end{pmatrix}$ - required coefficients of the quadric.
- Solve the system $A^t A\overline{X} = A^t B$, where $\overline{X}$ is the least square solution, with classical Gauss method.

2.2 Calculation of the principal curvatures

After the quadric $z = f(x, y) = ax^2 + bxy + cy^2 + dx + ey + f$ has been found for each node of the mesh the principal curvatures are calculated using notions of classical differential geometry [15]. The matrix of the first fundamental form for our quadric is written in the form: $G = \begin{pmatrix} 1 + f_x^2 & f_x f_y \\ f_x f_y & 1 + f_y^2 \end{pmatrix}$.

The matrix of the second fundamental form is

$$Q = \begin{pmatrix} \frac{f_{xx}}{\sqrt{1 + f_x^2 + f_y^2}} & \frac{f_{xy}}{\sqrt{1 + f_x^2 + f_y^2}} \\ \frac{f_{xy}}{\sqrt{1 + f_x^2 + f_y^2}} & \frac{f_{yy}}{\sqrt{1 + f_x^2 + f_y^2}} \end{pmatrix}.$$

The eigenvalues of these pair of forms, i.e. of the equation $\det(Q - \lambda G) = 0$, are the principal curvatures: maximal $\lambda_{\max}$ and minimal $\lambda_{\min}$.

3. Method

3.1 Calculation of the "improved" normals

For realization of our method we need at each node of the mesh not only normals defined in section 3.1 but also "improved" normals which will be as close as possible to the smooth surface approximated by initial mesh. To calculate such normals we propose to use area-weighted coefficients. Consider an oriented triangle mesh. Let us define the "improved" normal in some node p of the mesh. For each triangle i incident to the concerned node

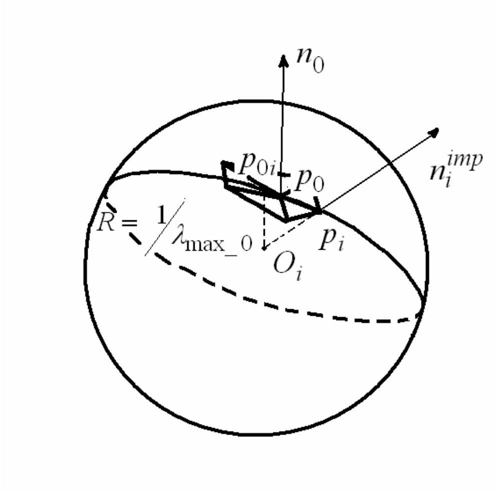


Fig.1: Node movement for non-regular mesh.

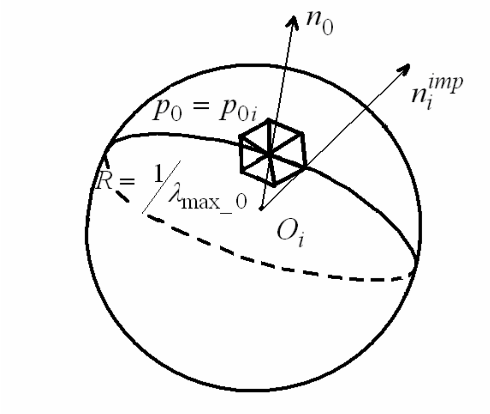


Fig.2: Node movement for regular mesh.

we calculate the weight coefficient $k_i = \frac{S_i}{\sum_i S_i}$. Here S_i is the area of the triangle i that incident to p. We calculate normals for each triangle incident to the concerned node and averaging them at that node with coefficients $1/k_i$. Then we normalize these averaged normals.

3.2 CBMI algorithm

Let us consider some node p_0 of the mesh and all nodes $p_1, p_2, \dots, p_k$ associated with the node p_0. Let n_i and n_i^{imp}, $i = 0,1,\dots,k$ are respectively "ordinary" and "improved" normals defined at the node p_i. For each node p_i, $i = 1,\dots,k$ new position p_{0i} of the node p_0 is found by the following procedure. If $|\lambda_{\max\_0}| > \varepsilon$ (in our examples we used $\varepsilon = 0.3$), where $\lambda_{\max\_0}$ is the maximal principal curvature defined in the node p_0, through the node p_i we

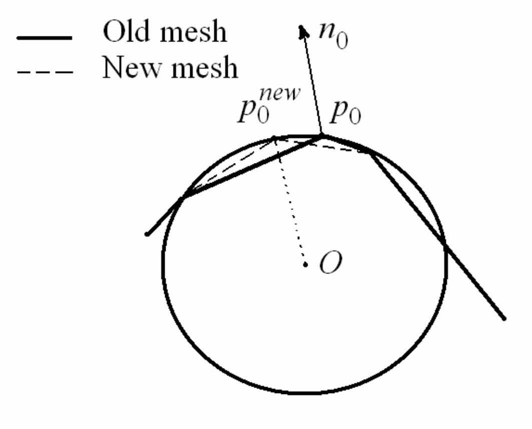


Fig.3: Node movement in 2D case.

draw a sphere with a center on the line defined by the normal n_i^{imp} and with a radius equal to $\left| \frac{1}{\lambda_{\max\_0}} \right|$. Then in the direction of the normal n_0 we draw the radius of that sphere. The point, in which this radius picks the sphere, is sought point p_{0i}. After the all points p_{0i}, $i = 1,...,k$ are found the new position of the node p_0 is defined by averaging the coordinates of p_{0i}. Described procedure is shown in Fig.1. It is clear that in case of the regular mesh the found new position of the node p_0 coincides with position of this node as shown in Fig.2.

For more lucidity we can consider how described algorithm works in 2D case. In this case the problem is formulated in the following way. It is necessary to reduce the difference between the lengths of the segments of the given polygonal line keeping the new nodes as close as possible to the curve approximated this polygonal path.

The CBMI algorithm is transformed into the following procedure. For each node of the polygonal line we define the “ordinary” normals similarly to 3D case. Then if the node is not the end vertex we draw the circumference passing through concerned node (p_0 for definiteness) and two nodes incident to it. From the center of the circumference we draw the radius in the direction opposite to the normal defined in p_0. Obtained point on the circumference is the new position for p_0 as shown in Fig.3.

In case of $|\lambda_{\max\_i}| < \varepsilon$ the nodes p_i, $i = 0,1,...,k$ are situated in the plane and the new position of p_0 is found by the simple Laplacian smoothing. It is possible to apply constrained Laplacian smoothing or another suitable method for improving of plane mesh.

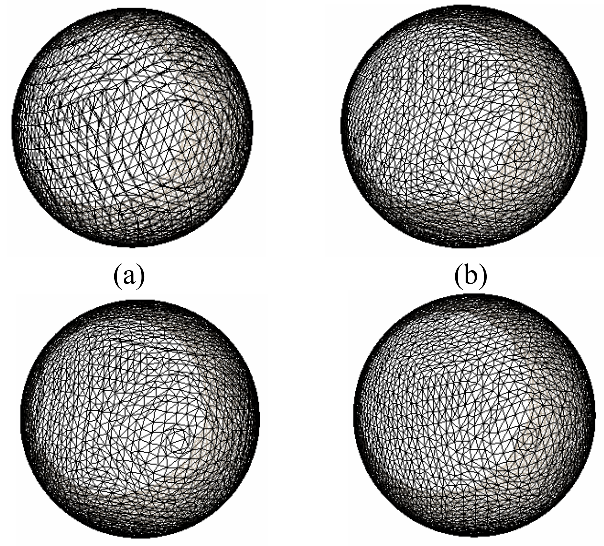


Fig.4: (a) Mesh of the sphere; (b) Mesh optimized with Laplacian smoothing; (c) Mesh optimized with CBMI algorithm using “ordinary” normals; (d) Mesh optimized with CBMI algorithm using “ideal” normals.

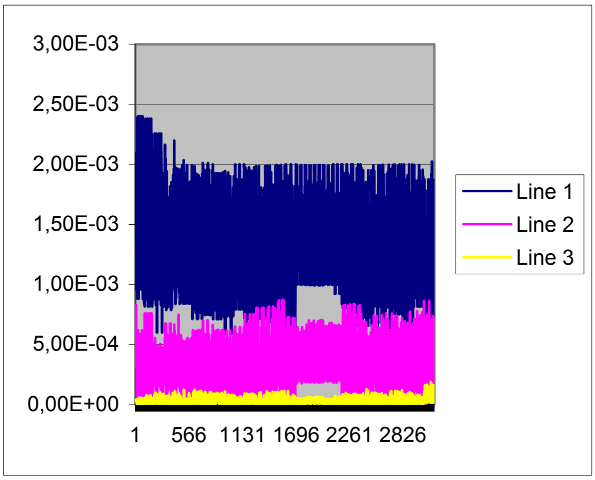


Fig.5: The deviation meshes obtained by Laplacian smoothing and CBMI algorithm from the initial sphere (Line 1 – Laplacian smoothing; Line 2 – CBMI with “ordinary” normals; Line 3 – CBMI with “ideal” normals)

Note that the “ordinary” normal serves as an indicator to the value of the node shift. The closer this normal is to the “ideal” normal (i.e. to the normal to the smooth surface approximated by the initial mesh), the smaller the value of the node movement is. On the other hand using the maximal curvatures guarantees that the new nodes are situated very close to the approximated smooth surface.

κ (aspect ratio)	Initial mesh	Laplacian smoothing	CBMI with "ordinary" normals	CBMI with "ideal" normals
1.0 -1.5	1114	2848	2456	2583
1.5 - 2.0	2676	2818	2784	2816
2.0 - 3.0	1075	531	912	797
3.0 - 4.0	446	25	65	22
4.0 - 5.0	141	6	9	12
5.0 - 7.5	279	2	4	0
7.5 -10.0	85	0	0	0
10.0-15.0	163	0	0	0
15.0-	251	0	0	0

Table 1: Hystograms of aspect ratio in initial and optimized with Laplacian and CBMI algorithm meshes for sphere (Fig.4).

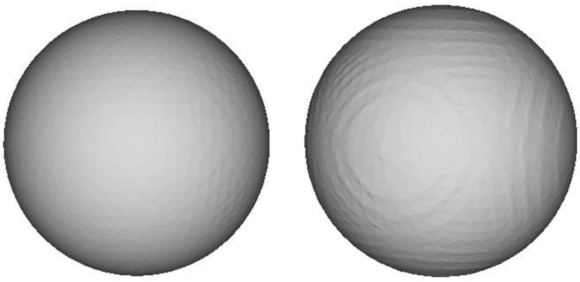


Fig.6: Comparison between images by CBMI algorithm and Laplacian smoothing. (a) The sphere optimized with CBMI algorithm using "ideal" normals (the difference of the volume from the original model is 0.93%); (b) The sphere processed with Laplacian smoothing (the difference of the volume from the original model is 1.83%).

4. Results

We have tested CBMI algorithm on various meshes. Firstly, we used triangulated meshes representing analytical surfaces to calculate the deviation the new mesh from these surfaces easily. Fig.4 shows triangulated mesh on the sphere and the results of Laplacian smoothing and CBMI algorithm. To demonstrate the improvement of the initial mesh we use aspect ratio $\kappa = l_{\max}/l_{\min}$, where $l_{\max}$ and $l_{\min}$ are respectively lengths of the maximal and minimal sides of the triangle. It can be seen from the Table 1 that Laplacian smoothing can improve the shapes of triangles a little better

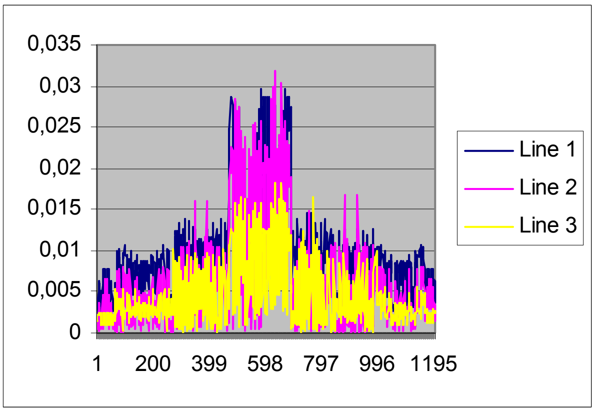


Fig.7: The deviation meshes obtained by Laplacian smoothing and CBMI algorithm from the initial ellipsoid (Line 1 – Laplacian smoothing; Line 2 – CBMI with "ordinary" normals; Line 3 – CBMI with "improved" normals).

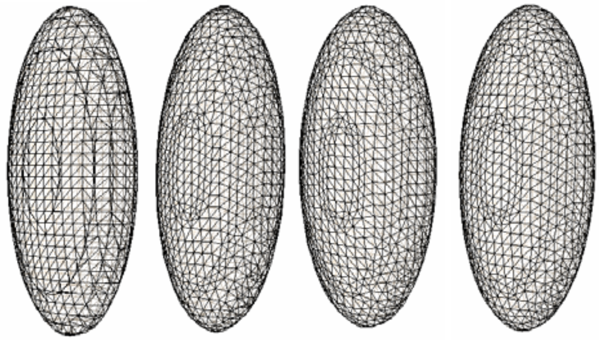


Fig.8: (a) Mesh of the initial ellipsoid; (b) Mesh optimized with the Laplacian smoothing; (c) Mesh optimized with CBMI algorithm using "ordinary" normals; (d) Mesh optimized with CBMI algorithm using "improved" normals.

than CBMI algorithm. On the other hand the Fig.5 demonstrates that the deviation of the new mesh from the initial sphere is much bigger for Laplacian smoothing.

We have tested our algorithm using "ordinary" normals (Fig.4c) and "ideal" normals calculated analytically (Fig.4d) to show how the accuracy of the method depends from the accuracy of normal estimation. Fig.7, 8, 9, and 11, Tables 2 and 3 demonstrate similar results for meshes on the ellipsoid and paraboloid. In Fig.8d and 11d we used CBMI algorithm with "improved" normals described in section 4.1. As it can be seen in Fig.6 and 10 the meshes improved with CBMI algorithm represent much more smooth surfaces then the meshes processed with Laplacian smoothing.

Finally complex meshes of a bone and a horse are presented in Fig.12 and 13 to illustrate the effectiveness CBMI algorithm on large surface meshes. The Tables 4 and 5 show the improvement of the initial meshes.

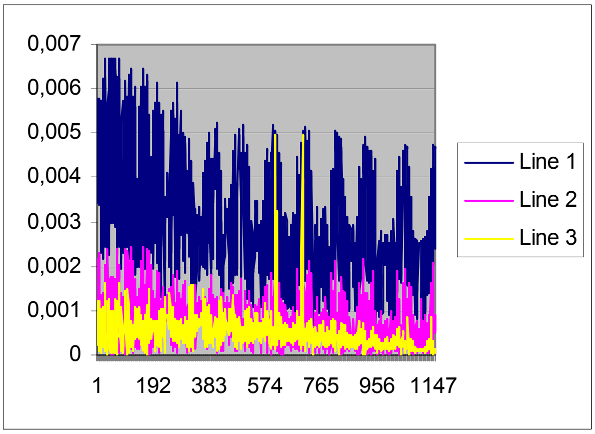


Fig.9: The deviation meshes obtained by Laplacian smoothing and CBMI algorithm from the initial paraboloid (Line 1 – Laplacian smoothing; Line 2 – CBMI with "ordinary" normals; Line 3 – CBMI with "improved" normals).

5. Conclusions

In this paper we presented a novel technique (CBMI algorithm) to improve the quality of the triangulated mesh without deviation of the new mesh from the initial surface too much.
At each node of the mesh the principal curvatures are calculated. The new position of the node is found using the value of the maximal curvature at that node.
CBMI algorithm has been successfully tested both for the meshes on the analytical surfaces and for the meshes on the complex surfaces.
Because for PDE applications the surfaces consisted of the simple geometric primitives, such as sphere, ellipsoid, cylinder, prevail we concentrate our attention on different analytical surfaces and show that CBMI algorithm is very suitable approach to improve mesh on such surfaces. The deviation a new mesh from the initial analytical surfaces (sphere, ellipsoid and paraboloid) has been calculated for the Laplacian smoothing and CBMI algorithm. The aspect ratio was presented and calculated to show the improvement of the meshes. The practical applications have good results in the sense of volume preserving and visual appearance (see Fig.6 and 10).
As it has been mentioned in Section 2 there are a number of remarkable techniques for surface denoising with simultaneous mesh improvement. Let us note that as it can be seen from color images in Fig.14, 15, and 16 (software tool available on WWW at http://www.mpi-sb.mpg.de/%7Ebelyaev/soft/soft.html of Dr. Ohtake from Max-Planck-Institut für Informatik

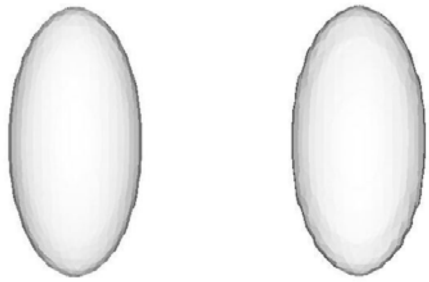


Fig.10: Comparison between images by CBMI algorithm and Laplacian smoothing. (a) The ellipsoid optimized with CBMI algorithm using "ideal" normals (the difference of the volume from the original model is 0.41%); (b) The ellipsoid processed with Laplacian smoothing (the difference of the volume from the original model is 2.86%).

κ (aspect ratio)	Initial mesh	Laplacian smoothing	CBMI with "ordinary" normals	CBMI with "improved" normals
1.0 -1.5	1118	1714	1615	1842
1.5 - 2.0	4786	612	668	493
2.0 - 3.0	232	64	107	61
3.0 - 4.0	164	6	6	0
4.0 - 5.0	60	0	0	0
5.0 - 7.5	106	0	0	0
7.5 -10.0	58	0	0	0
10.0-15.0	32	0	0	0
15.0-	148	0	0	0

Table 2: Hystograms of aspect ratio in initial and optimized with Laplacian smoothing and CBMI algorithm meshes for ellipsoid.

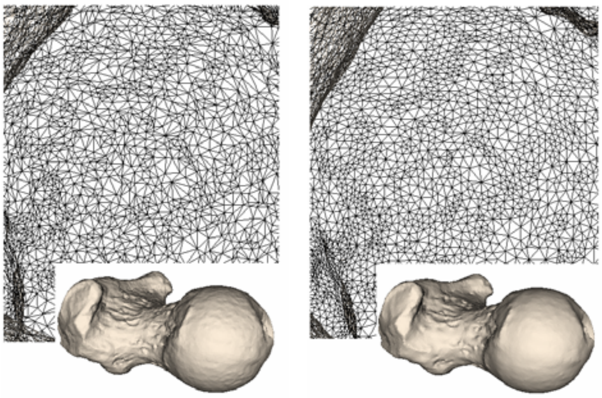
was used) applying CBMI algorithm to the surfaces without noise gives us more smooth results in comparison with other techniques.
In future works we intend to extend described algorithm to the general polygonal meshes. As it has been shown the accuracy of the method heavily depends on the accuracy of the normal estimation. Therefore we will attempt to improve the algorithm for normal estimation.

6. Acknowledgments

We wish to express our special thanks to Professor A. T. Fomenko (Moscow State University, Russia) for his valuable advices. The authors are grateful to D. Efremov (Moscow Pedagogical State University, Russia) and S. Rogojkin (Moscow State University, Russia) for their help and interesting discussions. We would like to thank also N. Kojekine (Tokyo Institute of Technology, Japan).

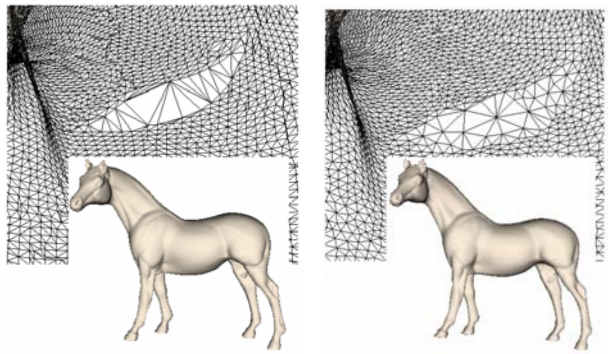
References:

1. P. J. Frey, About Surface Remeshing, *Proceedings of the 9th International Meshing Roundtable*, Sandia National Laboratories, 2000, pp. 123-136.
2. H. Hoppe, T. Duchamp, J. McDonald, W. Stuetzle., Mesh Optimization. In *Computer Graphics (SIGGRAPH 93),* 1993, pp. 19-26.
3. H. L. de Cougny, Refinement and Coarsening of Surface Meshes. *Engineering with Computers,* 1998, 14(3):214.
4. D. A. Field, Laplacian Smoothing and Delaney Triangulations, *J. Communications in Applied Numerical Methods,* Vol.4, 1998, pp. 709-712.
5. I. Babushka, O. C. Zienkiewicz, J. Gago, and E. R. de A. Oliviera (eds.), *Accuracy Estimates and Adaptive Refinement in Finite Eelement Computations,* John Wiley & Sons, Chichester, 1986, pp. 281-297.
6. K. Shimada and D. C. Gossard, Bubbled Mesh: Automated Triangular Meshing of Non-manifold Geometry by Sphere Packing, *Proceedings of the ACM Third Symposium on Solid Modeling and Applications*, 1995, pp. 409-419.
7. L. A. Freitag, On Combinian Laplacian and Optimization-Based Mesh Smoothing Techniques, *Trends in Unstructured Mesh Generation*, AMD – Vol.220, 1997, pp. 37-43.
8. V. Parthasarathy and S. Kodiyalam, A Constrained Optimization Approach to Finite Element Mesh Smoothing, *J. Finite Elements in Analysis and Design,* Vol.9, 1991, pp. 309-320.
9. S. A. Canann, J. R. Tristano, and M. L. Staten, An Approach to Combined Laplacian and Optimization-Based Smoothing for Triangular, Quadrilateral, and Quad-Dominant Meshes, *Proceedings of the 7th International Meshing Roundtable,* 1998, pp. 479-494.
10. O. P. Jacquotte and G. Goussement, Structured Mesh Adaptation: Space Accuracy and Interpolation Methods, *Computer Methods in Applied Mechanics and Engineering*, Vol.101, 1992, pp. 397-432.
11. A. Belyaev, Y. Ohtake, A Comparison of Mesh Smoothing Methods, *Accepted to Israel-Korea Bi-National Conference on Geometric Modeling and Computer Graphic*, Tel-Aviv, Israel, 2003, pp. 83-87.
12. R. V. Garimella, M. J. Shashkov, P. M. Knupp, Optimization of Surface Mesh Quality Using Local Parameterization, *Proceedings of the 11th International Meshing Roundtable,* Sandia National Laboratories, September 15-18, 2002, pp. 41-52.
13. T. Surazhsky, E. Magid, O. Soldea, G. Elber and E. Rivlin, A Comparison of Gaussian amd Mean Curvatures Estimation Methods on Triangular Meshes, *To appear in 2003 IEEE International Conference on Robotics & Automation (ICRA 2003)*.
14. G. Farin, *Curves and Surfaces for Computer Aided Design*, Fourth Edition, Academic Press, Boston, 1997.
15. B. A. Dubrovin, S. P. Novikov, A. T. Fomenko, *Modern Geometry. Methods and applications*, 2nd revised edition, Nauka, Moscow, 1986 (in Russian).



(a) (b)

Fig.12: Zoom in the mesh of the bone. Number of polygons is 68530. (a) Initial mesh and surface; (b) Mesh and surface optimized with CBMI algorithm.



(a) (b)

Fig.13: Zoom in the mesh of the horse. Number of polygons is 96966. (a) Initial mesh and surface; (b) Mesh and surface optimized with CBMI algorithm.

κ (aspect ratio)	Initial mesh	Mesh optimized with CBMI algorithm
1.0 -1.5	19240	41452
1.5 - 2.0	29912	21233
2.0 - 3.0	16492	5269
3.0 - 4.0	2609	429
4.0 - 5.0	170	85
5.0 - 7.5	93	50
7.5 -10.0	14	9
10.0-15.0	0	3
15.0-	0	0

Table 4: Hystograms of aspect ratio in initial and optimized with CBMI algorithm meshes for the bone.

κ (aspect ratio)	Initial mesh	Mesh optimized with CBMI algorithm
1.0 -1.5	22632	56877
1.5 - 2.0	66439	38547
2.0 - 3.0	5989	1444
3.0 - 4.0	782	79
4.0 - 5.0	370	15
5.0 - 7.5	376	4
7.5 -10.0	170	0
10.0-15.0	140	0
15.0-	68	0

Table 5: Hystograms of aspect ratio in initial and optimized with CBMI algorithm meshes for the horse.

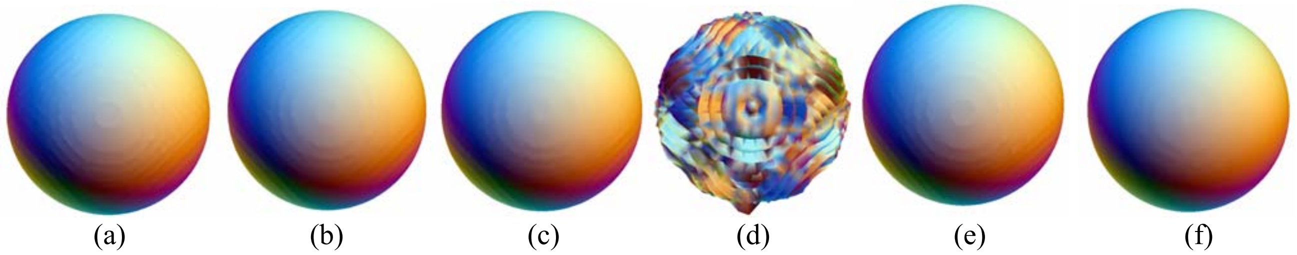


Fig.14: (a) initial sphere; (b) sphere optimized with Laplacian smoothing; (c) sphere optimized with Bi-Laplacian smoothing; (d) sphere optimized with Mean Curvature Flow (Desbrun); (e) sphere optimized with Taubin smoothing; (f) sphere optimized with CBMI algorithm.

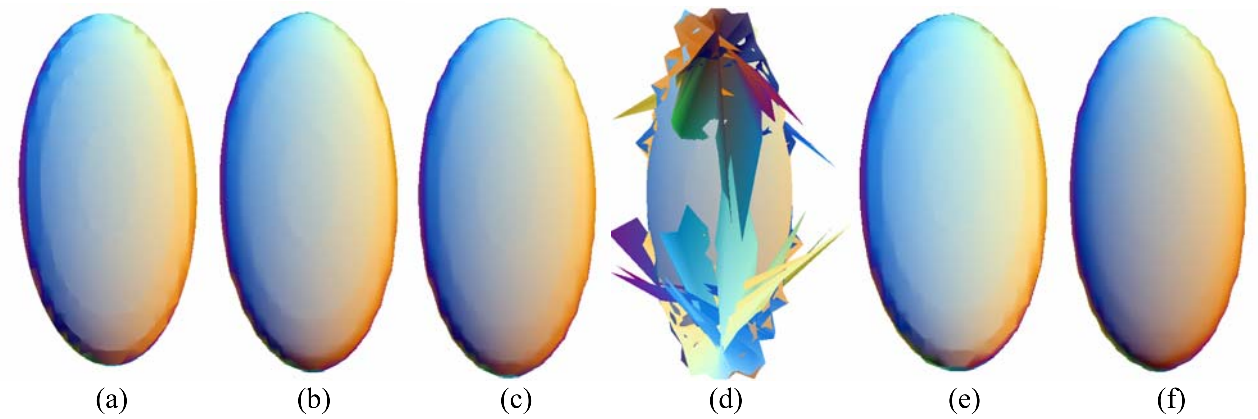


Fig.15: (a) initial ellipsoid; (b) ellipsoid optimized with Laplacian smoothing; (c) ellipsoid optimized with Bi-Laplacian smoothing; (d) ellipsoid optimized with Mean Curvature Flow (Desbrun); (e) ellipsoid optimized with Taubin smoothing; (f) ellipsoid optimized with CBMI algorithm.

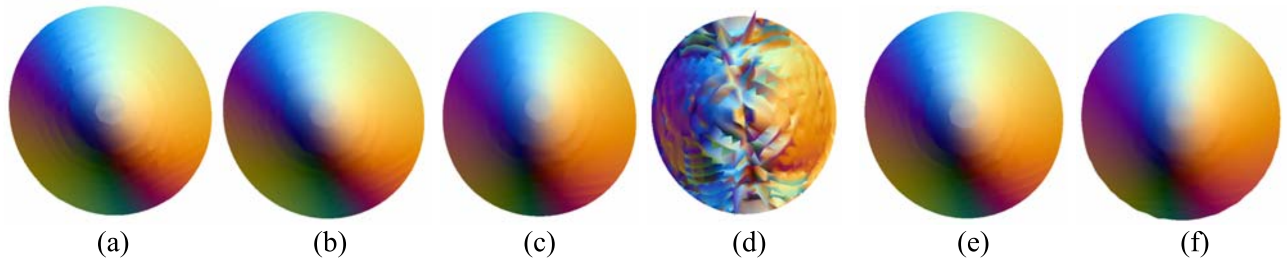


Fig.16: (a) initial paraboloid; (b) paraboloid optimized with Laplacian smoothing; (c) paraboloid optimized with Bi-Laplacian smoothing; (d) paraboloid optimized with Mean Curvature Flow (Desbrun); (e) paraboloid optimized with Taubin smoothing; (f) paraboloid optimized with CBMI algorithm.