

26th International Meshing Roundtable, IMR26, 18-21 September 2017, Barcelona, Spain

Hyperelastic springback technique for construction of prismatic mesh layers

V.A. Garanzha^{a,*}, L.N. Kudryavtseva^a

^a*Dorodnicyn Computing Center FRC CSC RAS, Vavilova str. 40, 119333 Moscow, Russia*

Abstract

We consider an algorithm for construction of thick prismatic mesh layers which works as follows. A triangular surface mesh is specified as input. Then, thin initial layer of highly compressed hyperelastic material glued to the surface is constructed using robust algorithm for computation of discrete normals. This pre-stressed material expands, possibly with self-penetration and extrusion to exterior of computational domain. Special preconditioned relaxation procedure is proposed based on the solution of stationary springback problem. It is shown that preconditioner can handle very stiff problems related to construction of very thick one-cell-wide layers for rather fine surface meshes. Once an offset prismatic mesh is constructed self-intersections are then eliminated using iterative prism cutting procedure. Next, variational advancing front procedure is applied for refinement and precise orthogonalization of prismatic layer near boundaries. It is guaranteed that the resulting mesh is free from inverted prisms.

© 2017 The Authors. Published by Elsevier Ltd.

Peer-review under responsibility of the scientific committee of the 26th International Meshing Roundtable.

Keywords: prismatic mesh layer; variational method; hyperelastic deformation; mesh untangling and optimization; preconditioned gradient search

1 Introduction

High quality simulation of viscous flows imposes rather strict requirements on computational meshes near solid boundaries. It is very important to construct meshes which provide orthogonality near boundary and precise control over mesh element size in the direction orthogonal to boundary irrespectively of the size and shape of surface mesh elements. Variational methods make this precise control possible [1]. Prismatic mesh layers consisting of triangular prisms, hexahedra or general polygonal prisms are flexible enough to be incorporated into automatic mesh generators while providing high quality mesh near boundaries. We consider semi-structured layers with the same mesh connectivity on each sublayer. In literature, sometimes more general case is considered where topology changes are admitted for mesh quality improvement [2]. However, we do not consider this case. Prismatic mesh layer is considered to be “thick” when its transverse size is comparable to the characteristic size of the geometric model. One can also call prismatic layer thick when its height is considerably larger compared to mesh element size on the surface.

* Corresponding author. Tel.: +7 499 135 73 78 ; fax: +7 499 783 33 26.

E-mail address: garan@ccas.ru

1. Variational principle for construction of prismatic layers

Let ξ_1, ξ_2, ξ_3 denote the Lagrangian coordinates associated with elastic material, and x_1, x_2, x_3 denote the Eulerian coordinates of a material point. Spatial mapping $x(\xi) : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ defines a stationary elastic deformation. The Jacobian matrix of the mapping $x(\xi)$ is denoted by C , where $c_{ij} = \partial x_i / \partial \xi_j$.

We look for the elastic deformation $x(\xi)$ that minimizes the following weighted stored energy functional [4]

$$F(x) = \int_{\Omega_\xi} w(\xi) W(C) d\xi, \quad (1)$$

where $W(C)$ is polyconvex elastic potential (internal energy) which is a weighted sum of shape distortion measure and volume distortion measure [8]:

$$W(C) = (1 - \theta) \frac{\left(\frac{1}{3} \text{tr}(C^T C)\right)^{3/2}}{\det C} + \frac{1}{2} \theta \left(\frac{1}{\det C} + \det C\right) \quad (2)$$

In most cases we set $\theta = 4/5$.

Since distortion measure (2) is minimized on the average, locally it can be quite large. In theory it can be infinite on the set of zero measure. In practice it means that with mesh refinement quality of mesh cell can locally deteriorate.

In practice, one can control the spatial distribution of distortion measure without actual contraction of the set of feasible mappings. Experience suggests that large values of distortion appear near boundaries and surfaces of material discontinuity. Hence it is possible to introduce a weight function $w(\cdot)$ in the Lagrangian or Eulerian coordinates which takes large values in critical regions and is close to unity elsewhere.

In the process of minimization, elements with a larger weight tend to have a smaller value of distortion function $W(C)$. Hence, their shapes and sizes are very close to the target ones. This simple approach proved to be very efficient for mesh orthogonalization near the boundary [8]. A proper choice of the weight allows us to satisfy the no-slip boundary conditions and to approximate boundary orthogonality conditions and prescribed mesh element size in the normal direction very accurately.

Theoretical arguments suggest that in order to eliminate the local singularities of the distortion function the weight distribution should be singular. However, this singularity is only reached in the limit of mesh refinement and for any given finite mesh weight distribution is bounded. One cannot prove that resulting deformation is quasi-isometric as in [8], [4] but numerical evidences suggest independence of the global mesh distortion bounds from the mesh size.

Suppose that domain Ω_ξ can be partitioned into convex polyhedra U_k . Then stored energy functional (1) can be approximated by the following semi-discrete functional:

$$F(x_h(\xi)) = \sum_k \int_{U_k} w(\xi) W(\nabla x_h(\xi)) d\xi \quad (3)$$

where $x_h(\xi)$ is continuous piecewise-smooth deformation.

In order to approximate integral over a convex cell U_k one should use certain quadrature rules. As a result semi-discrete functional (3) is replaced by the discrete functional:

$$F(x_h(\xi)) \approx \sum_k \text{vol}(U_k) \sum_{q=1}^{N_k} \beta_q w_q W(C_q) = F^h(x_h(\xi))$$

Here N_k is the number of quadrature nodes per cell U_k , C_q denotes the Jacobian matrix in q -th quadrature node of U_k , while β_q are the quadrature weights and w_q are values of weight function in the quadrature nodes.

The following majorization property should hold

$$F(x_h(\xi)) \leq F^h(x_h(\xi)) \quad (4)$$

This property can be used to prove that all intermediate deformations $x_h(\xi)$ providing finite values of discrete functional are homeomorphisms [4].

Let $G_\xi(\xi)$ and $G_x(x)$ denote the metric tensors defining linear elements and length of curves in Lagrangian and Eulerian coordinates in the domains Ω_ξ and Ω_x , respectively. Then, $x(\xi)$ is the mapping between metric manifolds M_ξ and M_x . The distortion functional (1) for this mapping can be written as

$$F(x) = \int_{\Omega_\xi} w(\xi) W(Q \nabla_\xi x H^{-1}) \det H d\xi, \quad (5)$$

where

$$H^T H = G_\xi, \det H > 0, \quad Q^T Q = G_x, \det Q > 0$$

are arbitrary matrix factorizations of metric tensors G_ξ and G_x .

The corresponding discrete functional can be written as follows

$$F^h(x_h(\xi)) = \sum_k \text{vol}(U_k) \sum_{q=1}^{N_k} w_q \beta_q W(Q_q C_q H_q^{-1}) \det H_q$$

Note, that in the presence of the control metrics exact majorization inequality can be violated and one should be careful with quadrature rules in order to guarantee certain relaxed formulation for majorization, say in the form

$$F(x_h(\xi)) \leq C F^h(x_h(\xi)), \quad (6)$$

where C is a constant. This inequality should guarantee that every intermediate iteration of the mesh generation method has finite energy for mapping as a whole and not just for the finite set of quadrature nodes.

Suppose that a thin layer of hyperelastic material is glued to the surface of the body. This material is highly compressed in the direction orthogonal to the surface. Now suppose that the surface of the layer opposite to the domain boundary is freed which results in classical springback problem for pre-stressed hyperelastic material. Static springback deformation can be found as a result of minimization of stored energy.

Elastic material is modelled by the one-cell-wide layer $\mathbb{P}$ of triangular prisms. For each prism $P \in \mathbb{P}$ the target prism P_t from the known prismatic layer in certain parametric manifold is specified. In order to construct such a target prism we consider a triangle T which belongs to the oriented surface triangulation of the polyhedral surface S . We map this triangle isometrically onto the plane $x_3 = 0$ in such a way that its normal is directed upwards and build on it rectangular triangular prism with the height $H(T)$ equal to the prescribed thickness of the layer. Consider piecewise-smooth deformation $x_h : \mathbb{P}_t \rightarrow \mathbb{P}$ as a solution of minimization problem for (3) with free boundary. When equilibrium solution is attained the thickness of elastic material would approximate the prescribed one. Note that at the springback relaxation stage material self-contact is ignored hence global overlaps are allowed.

Consider auxiliary prism P^ε constructed on the same triangular base T with the vertices p_i and $p_i + \varepsilon \nu_i$, $i = 1, \dots, 3$. Here p_i are vertices of T , ε is certain small constant and ν_i is the discrete unit normal to polyhedral surface S at the vertex p_i . Elastic deformation $x_h : \mathbb{P}_t \rightarrow \mathbb{P}^\varepsilon$ is quite far from isometry since $H(T)$ generally is much larger compared to ε . The tensions inside elastic material would move free surface away from the body.

Springback computation under strong compression is rather difficult. Hence we use the set of successive target states in order to relax the stiffness of the problem. The sequence of target prisms defining deformations $x_h : \mathbb{P}_t^h \rightarrow \mathbb{P}^h$ is constructed via gradual enlargement of the target height h from ε to $H(T)$.

Initial height ε is chosen in such a way that prismatic layer $\mathbb{P}^\varepsilon$ is admissible. If for a small ε the layer still contains inverted prisms then preliminary untangling problem is solved using technique from [4].

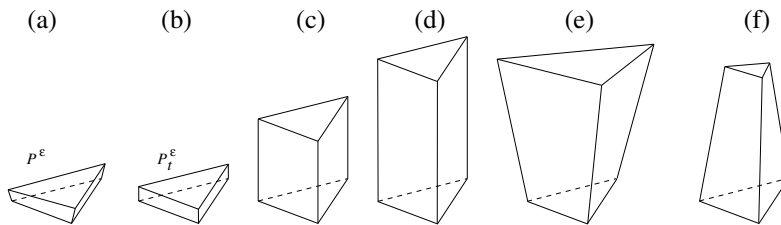


Fig. 1: (a) initial thin prism with $h = \varepsilon$, (b)-(d) height enlargement for target prisms, (e), (f) real prisms after springback

Fig. 1 illustrates the springback technique: (a) thin initial prism P^ε , (b) initial target prism P_t^ε , (c)-(d) the target prism growth, while (e)-(f) show how real equilibrium prisms near convex and convex surface fragments, respectively, look like. Note that the attainable prism thickness has a variable value. For example, an attempt to build thick layer inside a sphere may lead to real thickness which is much smaller compared to target one since quite strong compression of the upper part of thick prism prevents further growth of the layer.

One can reformulate the above procedure in algebraic terms. For each prism we specify metric tensor G_ξ in Lagrangian coordinates in such a way that after minimization of (5) thin layer is obtained. Since target shapes are orthogonal we set $(G_\xi)_{13} = (G_\xi)_{23} = 0$. Elements $(G_\xi)_{ij}$, $i, j = 1, 2$ are fixed while element $(G_\xi)_{33}$ is gradually enlarged from ε^2 to $H^2(T)$. After each enlargement step variational problem (5) is solved approximately. For largest value of $(G_\xi)_{33}$ more minimizations iterations are used.

2. Choice of the quadrature rules

In order to approximate integral (3) over the cell U_k one should use quadrature rules. In each prism elastic deformation is approximated by the bilinear mapping

$$x(\xi) = (p_0(1 - \xi_1 - \xi_2) + p_1\xi_1 + p_2\xi_2)(1 - \xi_3) + (p_3(1 - \xi_1 - \xi_2) + p_4\xi_1 + p_5\xi_2)\xi_3 \quad (7)$$

Note that this function maps rectangular prism with half of unit square as a base onto triangular prism with 3d vertices $p_0 - p_5$. The numbering scheme for vertices is shown in Fig. 2. In order to build mapping of target prism onto current cell one has to use composition of mappings $x(\xi) \circ \eta(\xi)^{-1}$ where function $\eta(\xi)$ is similar to (7).

The columns of the Jacobian matrix of mapping (7) can be written as

$$\begin{aligned} \frac{\partial x(\xi)}{\partial \xi_1} &= (p_1 - p_0)(1 - \xi_3) + (p_4 - p_3)\xi_3 \\ \frac{\partial x(\xi)}{\partial \xi_2} &= (p_2 - p_0)(1 - \xi_3) + (p_5 - p_3)\xi_3 \\ \frac{\partial x(\xi)}{\partial \xi_3} &= (p_3 - p_0)(1 - \xi_1 - \xi_2) + (p_4 - p_1)\xi_1 + (p_5 - p_2)\xi_2 \end{aligned}$$

Hence the Jacobian matrix admits representation

$$\nabla_\xi x = C_1 \Lambda_1(\xi) + C_2 \Lambda_2(\xi) + C_3 \Lambda_3(\xi), \quad \sum_{j=1}^3 \Lambda_j = I, \quad \Lambda_j \geq 0$$

$$\begin{aligned} \Lambda_1 &= \text{diag}(1 - \xi_3, 1 - \xi_3, \xi_1), \quad C_1 = (p_1 - p_0 \quad p_2 - p_0 \quad p_4 - p_1) \\ \Lambda_2 &= \text{diag}(\xi_3, \xi_3, \xi_2), \quad C_2 = (p_4 - p_3 \quad p_5 - p_3 \quad p_5 - p_2) \\ \Lambda_3 &= \text{diag}(0, 0, 1 - \xi_1 - \xi_2), \quad C_3 = (p_1 - p_0 \quad p_2 - p_0 \quad p_3 - p_0) \end{aligned}$$

Thus the majorization principle for polyconvex distortion measures [4], [6] can be applied which means that for polyconvex function (2)

$$W(\nabla_\xi x) \leq \sum_{\nu=1}^{12} a_\nu(\xi) W(\tilde{C}_\nu),$$

where $\tilde{C}_\nu$, $\nu = 1, \dots, 12$ means 3×3 “compound matrix” where k -th column is arbitrary chosen as a k -th columns of any of the basis matrices C_i , $i = 1 \dots 3$.

This inequality provides natural geometric quadratures for construction of discrete distortion measure for prism. To this end one has to consider all compound matrices $\tilde{C}_\nu$ generated by the basis matrices C_1, C_2, C_3 . Note that total number of quadrature nodes is not equal to $3^3 = 27$ as it was suggested in theorem from [4], [6]. Precise number is equal to $2 \times 2 \times 3 = 12$ since first two columns of the matrix Λ_3 are equal to zero. The presence of additional mapping $\eta(\xi)$ does not prevent using theorem from [4], [6] since for orthogonal prisms mapping (7) is affine.

The set of quadrature nodes can be split into two groups each consisting of 6 elements.

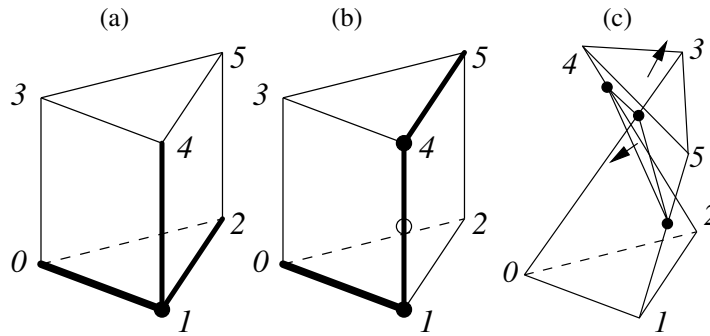


Fig. 2: (a) Quadrature node at the vertex, (b) quadrature node at the edge, (c) prism with positive Jacobian of mapping (7) at the vertices and negative Jacobian at some edges

Compound matrix where both first and second column are simultaneously taken from either matrix C_1 or from matrix C_2 while third column is arbitrary taken from third columns of C_i corresponds to vertex-based quadrature node, see Fig. 2(a). When first two columns of $\tilde{C}_v$ are chosen from different matrices C_1 and C_2 it corresponds to vertical edge-based quadrature node, see Fig. 2(b).

To summarize, one should use 6 vertex-based nodes and 6 vertical edge-based nodes in order to discretize functional (3).

For the sake of simplicity one can try to use 6-node vertex-based quadratures. Numerical experiments have shown that resulting discrete variational problems becomes much less stiff in a sense that number of iterations to reach prescribed prism thickness is reduced but resulting layer may contain considerable number of twisted prisms with degenerate mapping (7). In principle one can first construct thick vertex-based prismatic layer and then try to untangle it using 12-node quadrature approximation. We were not able to make this scheme work. Untangling problem turned out to be too stiff for solver from [4].

Fig. 2 (c) shows the prism with positive vertex Jacobians and some negative edge Jacobians. If such a prism is encountered in the layer then an attempt to cut part of the layer by cutting off the same fraction of all transverse edges leads to self-overlaps on the resulting outer boundary of prismatic layer. One should also note that problem of intersection detection for twisted prisms is not correctly posed.

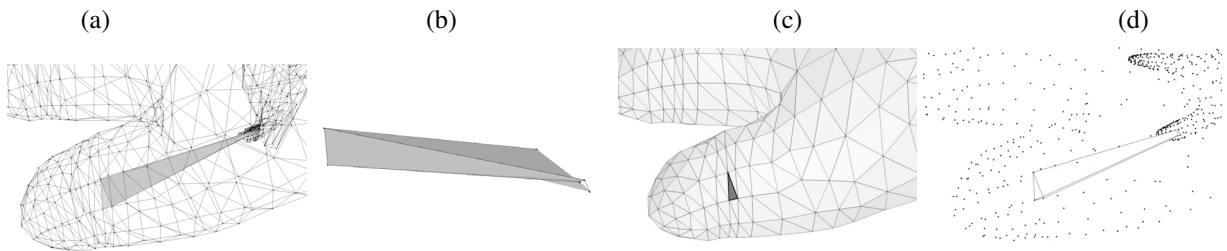


Fig. 3: (a) Twisted prism in the computed prismatic layer, (b) enlarged twisted prism viewed from the free boundary, (c) mesh fragment, (d) correct prism

Figs. 3(a)-(b) show twisted prism in the prismatic layer build inward from the surface of the “camel” model from Stanford collection. Vertex quadratures are used. Prism is viewed from inside of the camel in order to make the twist visible.

Fig. 3(c)-(d) shows the same prism for 12-node approximation. It is clear that this prism is not twisted.

Fig. 4(a)-(b) shows surface mesh on the “camel” model and cross-section of the layer, and Fig. 4(c)-(d) visualize the outer surface of prismatic layer which behaves as a generalized skeleton of the 3d domain.

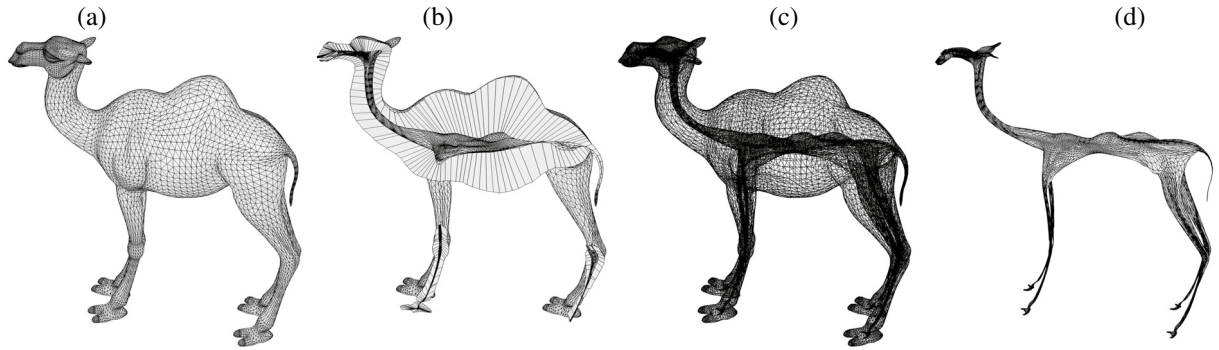


Fig. 4: (a) Initial surface mesh, (b) cross-section of prismatic layer, (c) model with its approximate skeleton, (d) approximate skeleton

Springback-based thick layer can be considered as an alternative solution to the problem of construction of medial surface. Varying weight function $w(\xi)$ in the functional (5) one can control the deviation of the layer from orthogonality and, in turn, its attainable thickness. The higher is the degree of orthogonality, the closer is the resulting layer to the medial axis-based one. Allowing larger nonorthogonality allows to diminish greatly influence of small but sharp disturbances on the surface and allows to construct rather thick layers.

3. Elimination of self-intersections

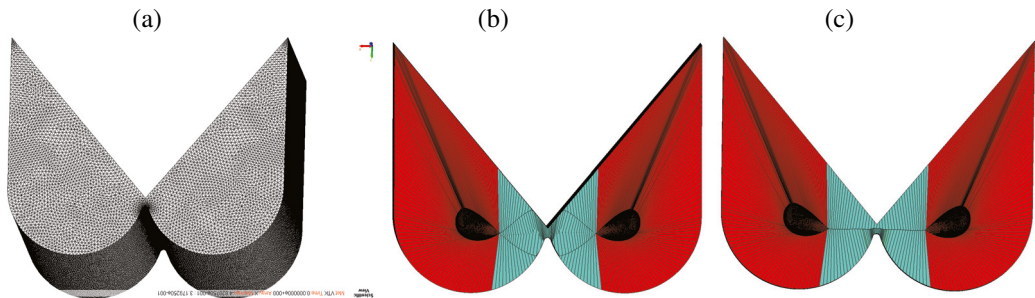


Fig. 5. (a) Surface mesh, (b) elimination of material in the exterior of the domain, (c) construction of the contact surface

We do not take into account self-contact of elastic material during springback. Instead we allow material to self-penetrate and freely intersect the boundary of domain. In order to eliminate self-intersection iterative cutting procedure is applied. First we cut off material on each prism which goes out of the computational domain. After that we build the list of all prisms which intersect other prisms and apply to the prisms in the list thickness reduction by certain relative coefficient β slightly less than unity. This procedure is repeated until all intersections are eliminated. As a result in the self-penetration zones certain “contact surface” is constructed. In the limiting case of opposite parallel planes at the distance D and with thicknesses of overlapping opposite layers equal to H_1 and H_2 , where $\delta H = H_1 + H_2 - D > 0$, this procedure will create opposite layers with thicknesses close to

$$H'_1 = \frac{H_1}{H_1 + H_2} D, \quad H'_2 = \frac{H_2}{H_1 + H_2} D$$

which means that in general found middle surface does not correspond to middle surface of overlap region.

Additional one-sided smoothing procedure is applied to the outer surface of the layer which can only reduce the thickness thus avoiding possibility of reappearance of overlaps. This process is illustrated in Fig. 5.

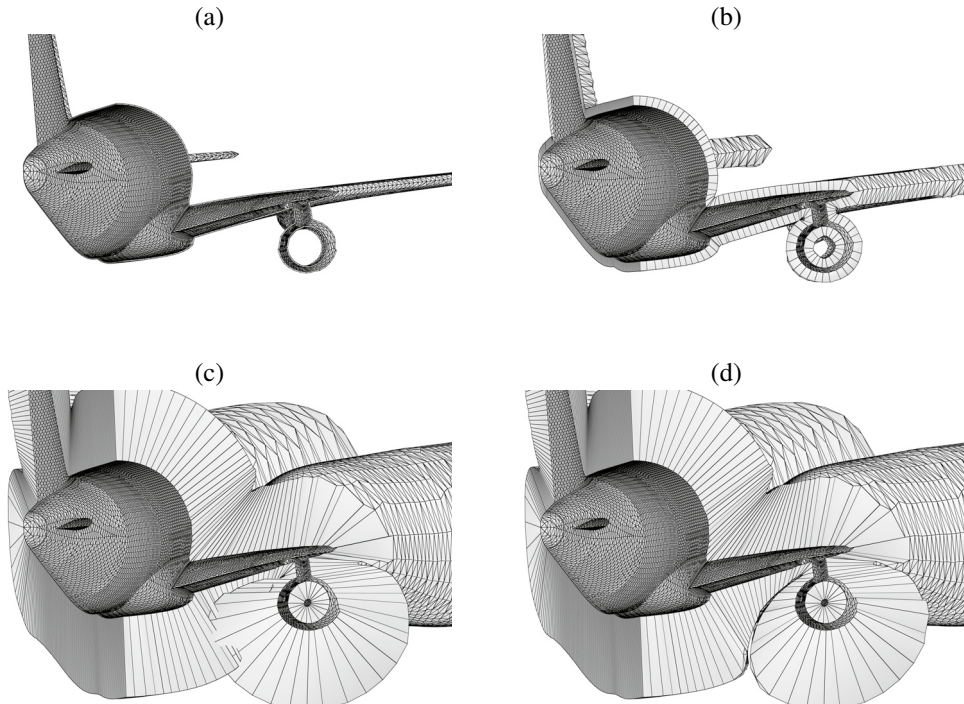


Fig. 6: (a) Initial thin layer, (b) intermediate layer, (c) thick layer, (d) thick layer without self-intersections

Algorithm stages are illustrated in Fig. 6–8 for simplified airplane model. Initial thin layer is shown in Fig. 6(a). Intermediate springback solution is shown in Fig. 6(b). Finite prismatic layer is quite thick and does not contain degenerate prisms but may contain self-overlaps and boundary overlaps as shown in Fig. 6(c). Elimination of self-intersections results in the mesh shown in Fig. 6(d).

Additional smoothing is applied to the outer surface of layer. Its vertices are allowed to move along the transverse edges of prisms. Smoothing procedure is based on the Laplace-Beltrami smoothing iterations applied to approximate distance function computed along transverse edges of the prisms. Mean value discretization [7] is used to enforce maximum principle. Smoothed surface is shown in Fig. 8(a).

4. Layer refinement and orthogonalization

As soon as the offset surface along with the set of very long prismatic cells is constructed one has to split the layer according to prescribed mesh size distribution in the direction orthogonal to the boundary.

Fig. 7(a) shows single prism of thick layer and imaginary mesh lines which are the images of the straight transverse edges of prism after refinement and orthogonalization. First layer is cut from the prism according to the prescribed mesh size distribution law (Fig. 7(b)), then, two-cell-wide layer is optimized (Fig. 7(c)). Here the weight $w(\xi)$ in the lower cell is much larger than the one in the upper cell. As a result the lower cell is orthogonalized. Since upper boundary of layer is fixed this procedure eventually leads to transfer of non-orthogonality from solid boundary to outer boundary of layer. We can either dismiss certain outer fraction of layer or use the sequence of weights where difference of weights on two layers eventually diminishes and tends to unity. The ratio of weights is largest in approximately one-quarter or one-third of the total layer. One could also consider movement of vertices on the outer boundary during optimization but here we do not use such an algorithm.

Successive splitting and orthogonalization are shown in Fig. 8(b)-(c). As one can see variational method is applied only to one-cell-wide or two-cell-wide layers, while total number of sublayers can be arbitrary large. Final layer which is orthogonal near boundary is shown in Fig. 8(d).

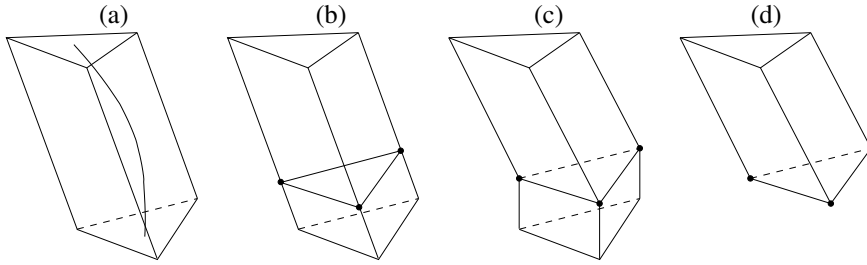


Fig. 7. Successive steps of layer splitting and orthogonalization

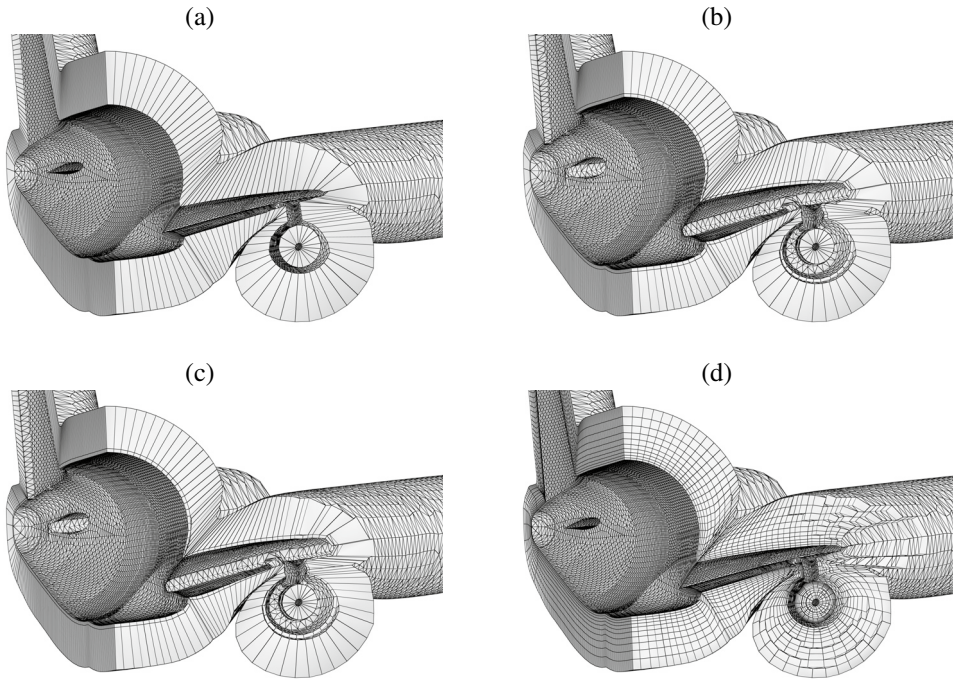


Fig. 8: (a) Layer after Laplace-Beltrami smoothing, (b) splitting into two sublayers, (c) orthogonalization of internal sublayer, (d) final prismatic layer

The above procedure guarantees the absence of degenerate elements that at each stage of the prismatic layer generation algorithm.

5. Double scaling for preconditioner

Minimization of the discrete functional can be formulated as a problem of minimization of function $F(Z)$ where argument is the vector Z such that $Z^T = (z_1^T \ z_2^T \ \dots \ z_{n_v}^T)$ where $z_k \in \mathbb{R}^3$, $k = 1, \dots, n_v$ are positions of mesh vertices. Hessian matrix $\tilde{H}$ of the function F is built of 3×3 blocks $\tilde{H}_{ij} = \frac{\partial^2 F}{\partial z_i \partial z_j^T}$. Here matrix $\tilde{H}_{ij}$ is placed on the intersection of i -th block row and j -th block column.

The Newton method for finding stationary point of the function can be written as follows

$$\sum_{j=1}^{n_v} \tilde{H}_{ij}(Z^l) \delta z_j + r_i(Z^l) = 0 \quad (8)$$

$$z_k^{l+1} = z_k^l + \tau_l \delta z_k, \quad k = 1, \dots, n_v \quad (9)$$

Here parameter τ_l is found as approximate solution of the following 1d problem

$$\tau_l = \arg \min_{\tau} F(Z^l + \tau \delta Z)$$

We use simple binary subdivision to find approximate minimum.

The following simple iterative scheme was suggested in [5], [10] by setting $\tilde{H}_{ij} = 0$ for $i \neq j$ in equations (8). Hence each iteration reduces to independent solution of 3×3 linear system with matrix $\tilde{H}_{ii}$ in i -th mesh vertex. Note that it follows from the polyconvexity of the stored energy that it is rank-one convex which in turn leads to positive definiteness of the matrices $\tilde{H}_{ii}$. Rank one convexity implicates that function F is convex as a function of three variables - the components of z_i , when all other vertices are fixed.

Unfortunately we have found that this small block Jacobi preconditioner would not allows us to attain target thickness of prismatic layer.

In 2000 Garanzha [8] proposed an alternative preconditioner. Let us set all off-diagonal elements in the matrices $\tilde{H}_{ij}$ to zero. Then linear system (8) will be partitioned into three independent linear systems with respect to three vectors δZ_m defined by the following permutation $(\delta Z_m)_i = (\delta z_i)_m$.

The size of each linear system is equal to the number of vertices n_v . Matrices of these linear systems are symmetric positive definite. It follows again from the rank one convexity, since function F is convex with respect to displacement of all vertices in the same direction.

Hence these linear systems can be approximately solved using preconditioned conjugate gradients method (PCG). This algorithm was used with certain success for many years but eventually we have found that it cannot handle stiff springback problems when target thickness of layer is quite large compared to the mesh size on the surface. Again, for stiff cases we were just not able to attain target thickness of the layer.

In order to resolve this difficulty we introduced new preconditioner which we call double scaling technique.

Let $B_i^T B_i = \tilde{H}_{ii}$ denote the factorization of 3×3 matrix $\tilde{H}_{ii}$. We apply to Hessian matrix $\tilde{H}$ the following block scaling

$$\tilde{H}^B = B_i^{-T} \tilde{H}_{ij} B_j^{-1}$$

This equality can be rewritten as $\tilde{H}^B = B^{-T} \tilde{H} B^{-1}$.

The next step is to apply permutation which allows to represent matrix $\tilde{H}^B$ as 3×3 block matrix with $n_v \times n_v$ blocks

$$\tilde{H} = P \tilde{H}^B P^T, \quad \tilde{H} = \begin{pmatrix} \tilde{H}_{11} & \tilde{H}_{12} & \tilde{H}_{13} \\ \tilde{H}_{21} & \tilde{H}_{22} & \tilde{H}_{23} \\ \tilde{H}_{31} & \tilde{H}_{32} & \tilde{H}_{33} \end{pmatrix}$$

Setting offdiagonal blocks in $\tilde{H}$ to zero, one obtains three independent linear systems with $n_v \times n_v$ matrices $\tilde{H}_{ii}$.

Preconditioned conjugate gradient technique is used for approximate solution with second order Choleski factorization [9] as a preconditioner

$$L_i L_i^T \approx \tilde{H}_{ii}$$

It is well known that approximate PCG solution to linear system

$$Ax = f$$

starting from zero initial guess which was obtained with relative error ϵ can be formally written as

$$x_\epsilon = R_\epsilon f,$$

where R_ϵ is certain symmetric positive definite matrix which in some sense approximates matrix A^{-1}

Hence we eventually replaced inverse of full Hessian matrix $\tilde{H}$ by the following matrix

$$H' = B^{-1} P^T R_\epsilon P B^{-T}$$

Thus increment vector which is used in (9) is defined by equality

$$\delta Z = H'R(Z^l), \quad R^T = (r_1^T, r_2^T, \dots, r_{n_r}^T)$$

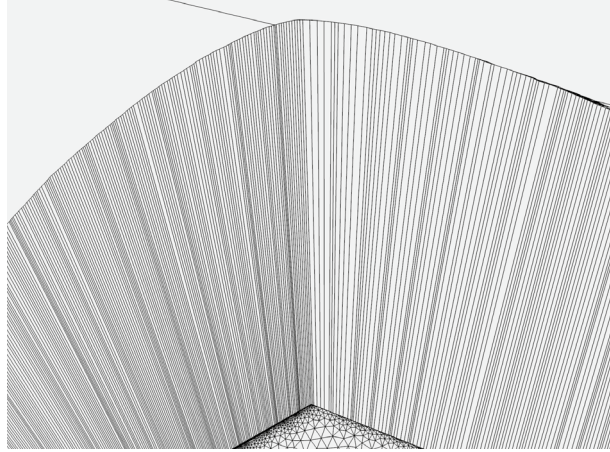


Fig. 9. Example of stiff springback problem

Fig. 9 illustrates the case of extremely thick prismatic layer where the relative thickness, i.e. the ratio of height of the prism to its base reaches 10^3 due to highly nonuniform mesh on the surface.

6. Construction of prismatic layers for polyhedral surfaces with non-Lipschitz vertices

A number of algorithms was suggested in order to assign normal vectors to the vertices of the polyhedral surfaces.

Denote such a vertex by p , then vector n_i is the unit normal to the adjacent triangle T_i with vertices $pp_i p_{i+1}$. Normal vector in p is computed via

$$v = \frac{\sum_i w_i n_i}{|\sum_i w_i n_i|}, \quad w_i > 0 \quad (10)$$

We use weights $w_i = \theta_i$, where θ_i is the angle of T_i at p . Note that well-known algorithms for computing vertex normals in general do not guarantee that discrete normal is really directed inward the domain, namely, that the following inequality holds

$$n_i^T v > 0 \quad (11)$$

Numerical experiments show that violations of this inequality are quite rare and are associated with sharp feature lines and conical vertices with complex neighborhood structure. Constrained algorithms for computation of discrete normals play important role for prismatic layer generation [3].

Assume that there exist a pair vertex p - adjacent triangle T with unit normal n , satisfying the following inequality

$$n^T v < \beta |v| \quad (12)$$

It means that discrete normal is almost tangential to the surface. In practice we use coefficient $\beta = 1 - \cos(\pi/9)$. If such a pairs are present in the mesh, then for each “suspicious” vertex we suggest to apply the following algorithm.

Consider at the vertex p a convex cone K defined as the intersection of the half-spaces $(x - p)^T n_i \geq 0$ defined by adjacent faces. If this cone is empty then normal vector satisfying constraints (11) does not exist and one cannot find coordinate frame where vertex neighbourhood is presented as Lipschitz-continuous elevation function.

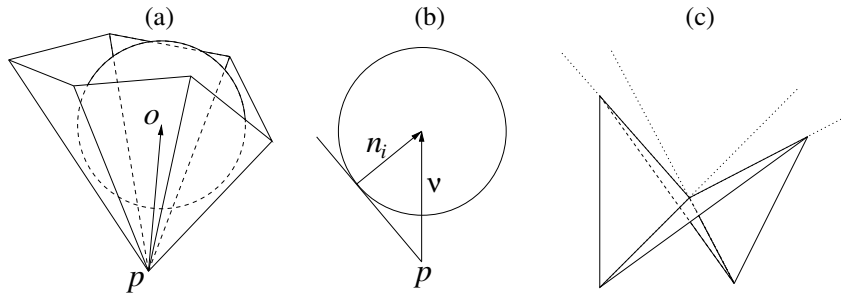


Fig. 10. (a) Ball inside convex cone, (b) ball touches the face of a cone, (c) construction of convex cone for mesh vertex

Let us try to place unit ball inside this cone such that the center o of the ball is as close to c as possible, as shown in Fig. 10 (a), while Fig. 10 (c) shows construction of convex cone for saddle-like polyhedral surface fragment.

Denote normal vector by $v = o - p$. Suppose that the ball touches the plane of the adjacent face with unit normal n_i , as illustrated in Fig. 10 (b). Then $|v|^2 - |v - n_i n_i^T v|^2 = 1$, i.e. $v^T n_i = 1$. Hence the problem of optimal ball placement is reduced to standard quadratic programming (QP) problem.

Find vector $v \in \mathbb{R}^3$ via minimization of

$$\min \frac{1}{2} |v|^2, \quad \text{s. t. } n_i^T v \geq 1, \quad i = 1, \dots, m \quad (13)$$

where m is the number of adjacent faces for vertex p .

It is geometrically evident that vector v is fully defined by 2 or 3 active faces, despite the fact that it can touch larger number of planes.

Since the number of suspicious vertices is small we use direct search method to solve QP problem instead of iterative technique. First we consider all pairs $n_i, n_j, i \neq j$ and check that the solution

$$v = \frac{n_i + n_j}{1 + n_i^T n_j}$$

satisfies all the remaining constraints (13). If such pair is not found, we search for a triple of distinct normals n_i, n_j, n_k being the solution of the linear system $(n_i \ n_j \ n_k)^T v = (1 \ 1 \ 1)^T$.

If no admissible solution is found then exterior penalty solution of the overdetermined QP problem is used to find the acceptable direction. Then several faces should be found which are not acceptable for this direction. The remaining faces are marked as active for variational method.

Simple “two cubes” model with non-Lipschitz vertices is shown in Fig. 11.

At least one of the prisms adjacent to non-Lipschitz vertex p should be degenerate and the Jacobian of mapping (7) should attain zero or negative values. For such a prism another approximation scheme for variational method should be used. Consider prism P adjacent to p and based on “bad” face. We exclude from the set of 12 quadrature nodes those tetrahedra which contain at least two prism edges originating from vertex p . Remaining quadrature nodes serve to guarantee nondegeneracy of this prism as a generalized polyhedron.

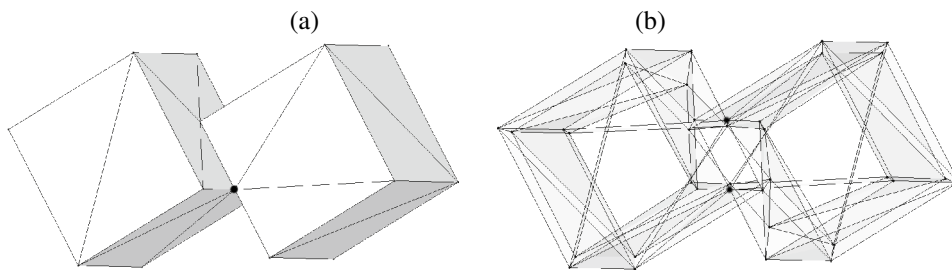


Fig. 11: (a) Model with two non-Lipschitz vertices (bold), (b) prismatic layer

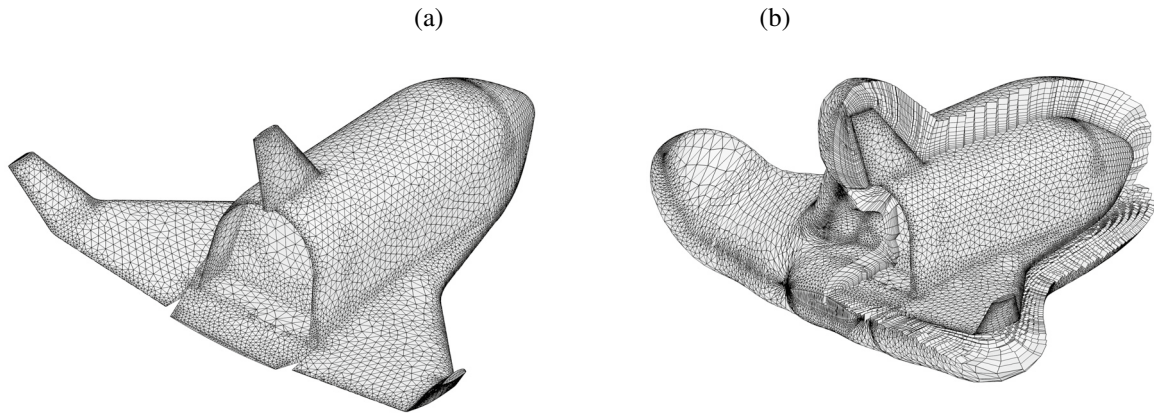


Fig. 12: (a) Surface mesh for TsAGI SRV model, (b) prismatic layer

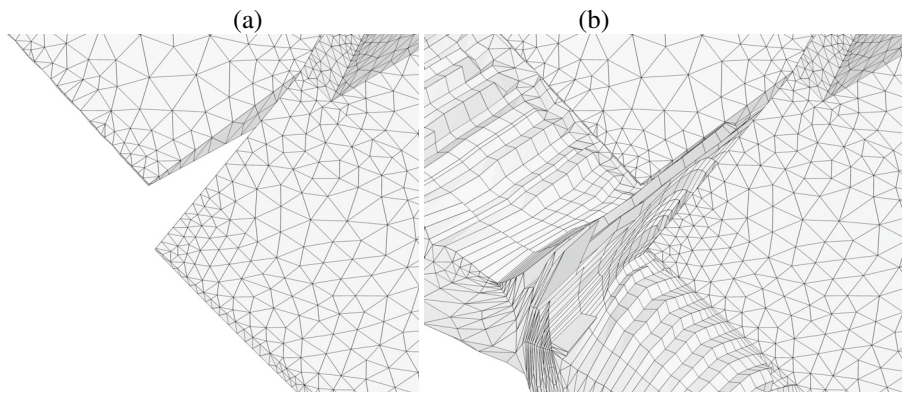


Fig. 13: (a) Complicated surface fragment, (b) fragment of the prismatic layer

Consider quite a complicated test case for prismatic meshing. The model of TsAGI RSV contains sharp tips and flaps. In order to build high quality mesh one should change topology of the layer near flap. Hence presented results illustrate the stress test where connectivity is not changed.

Fig. 12 shows surface mesh on TsAGI RSV and fragments of the prismatic layer.

Complex geometric structures on the surface of the model are shown in Fig. 13(a). Fig. 13(b) shows that very deep and thin cuts do not lead to reduction of the layer thickness.

This test case also provides nice illustration of the fact that the precise measuring of the thickness of “thick layer” is not simple problem. The most obvious solutions: true distance from the surface and length of mesh lines may provide unacceptable results.

Surface mesh on TsAGI RSV model does contain few non-Lipschitz vertices which does not prevent presented algorithm from constructing a thick layer.

7. Discussion

Numerical experiments have shown that deviation from orthogonality in resulting layers near surface is negligible, general quality of prismatic layers is quite good and it can be useful in industrial application. On the other hand variational method is time consuming if applied globally. It seems to be at least five times slower compared to the state-of-the-art industrial prismatic meshers. We briefly discuss here the way to reduce the computational costs of the algorithm. First of all better initial guess should be constructed. It can be done by smoothing the field of surface normals/transverse directions by solving simple unconstrained convex quadratic programming problem. This simple

correction of algorithm may sharply increase the admissible thickness of the vast majority of prismatic cells of the initial layer. Variational optimization can be applied locally, only around the zones where degenerate prisms are found. One should be quite careful since straightforward application of springback technique to the fragment of thin layer surrounded by thick layer results in very poor performance of iterative minimization due to the prisms with highly skewed upper lid as shown in Fig. 14(a). In order to make minimization problem less stiff one should use sliding boundary conditions as shown in Fig. 14(a).

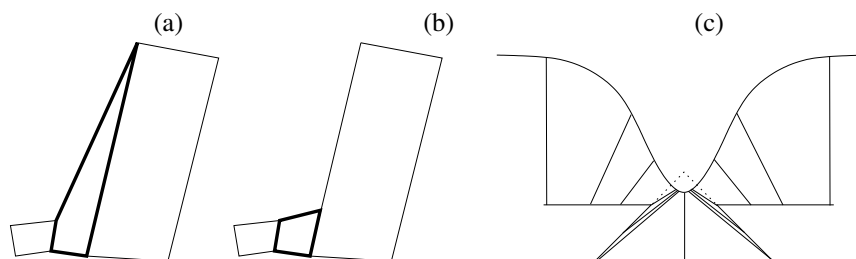


Fig. 14: (a) Bad coupling of thin and thick layer fragments, (b) coupling with sliding boundary conditions, (c) “locked” boundary configuration results in sharp local layer thickness reduction

In many cases there is no need to construct offset surfaces and one-cell-wide thick prismatic layers. Hence the above described technique while being very powerful becomes too complex and cumbersome. It has obvious drawbacks when “locking” of transverse directions leads to sharp layer thickness decrease as shown in Fig. 14(c). Note however, that second application of the same technique to the outer boundary of the first layer allows to recover thick layer, or even to mesh all the domain by the prismatic mesh. Still, the problem of smooth coupling of layers with variable thickness is not trivial and may require some additional smoothing passes with a careful choice of target cell shapes. Another drawback is related to construction of the traces of the prismatic layer on the side surfaces (“side walls”). Single-cell-wide offset is not compatible with curved side walls.

Nevertheless suggested technique still can be very useful when mesh should follow the normal mesh size distribution with very high precision irrespectively of the size and shape of the surface elements. Rather distorted cells can be created at a certain distance from the boundary but inverted mesh cells cannot appear since they result in the infinite value of the discrete stored energy.

Springback technique seems to be promising tool which can be used as a part of automatic fully hexahedral mesh generator.

Partial funding of this work by the NSH-8860.2016.1 project of the governmental program for supporting leading scientific schools is gratefully acknowledged. Research on preconditioned iterative algorithms for mesh optimization is supported by the Russian Science Foundation 16-11-10350 project.

References

- [1] V. Dyedov, D. Einstein, X. Jiao, I. A. Kuprat, J. Carson, F. del Pin, Variational Generation of Prismatic Boundary-Layer Meshes for Biomedical Computing. *Int J Numer Methods Eng.* 79(2009) 907-945.
- [2] A. Loseille, R. Löhner, Robust boundary layer mesh generation. *Proceedings of the 21st International Meshing Roundtable*. 2013 493-511.
- [3] Y. Kallinderis, S. Ward, Prismatic Grid Generation for Three-Dimensional Complex Geometries. *AIAA Journal* 31(1993) 1850-1856.
- [4] V.A. Garanzha, L.N. Kudryavtseva, S.V. Utyzhnikov, Untangling and optimization of spatial meshes // *Journal of Computational and Applied Mathematics*. 269 (2014) 24-41.
- [5] S.A. Ivanenko, Construction of nondegenerate grids. *Comput. Math. and Math. Phys.* 28(1988) 141-146.
- [6] L.V. Branets, V.A. Garanzha, Distortion measure for trilinear mapping. Application to 3-D grid generation. *Num. Linear Algebra Appl.* 9(2002) 511–526.
- [7] M.S. Floater, K. Hormann, Surface Parameterization: a Tutorial and Survey. // *Advances in Multiresolution for Geometric Modelling, Mathematics and Visualization*. Part 4. – Berlin, Heidelberg, New York: Springer, 2005. – P. 157-186.
- [8] V.A. Garanzha, The barrier method for constructing quasi-isometric grids. *Comput. Math. Math. Phys.* 40(2000) 1617-1637.
- [9] I.E. Kaporin, High quality preconditioning of a general symmetric positive definite matrix based on its $U^T U + U^T R + R^T U$ decomposition. *Numerical linear algebra with applications* 5 (1998) 483-509.
- [10] S.A. Ivanenko, A variational form of the Winslow grid generator. *Journal of Computational Physics*, 136(1997) 385-398.