Finite triangular surface mesh simplification with geometrical feature recognition

Y Zhang, H Wang, H Zhou*, and **J Li**
State Key Laboratory of Materials Processing and Mold and Die Technology, Huazhong University of Science and Technology, Wuhan, People's Republic of China

The manuscript was received on 5 January 2009 and was accepted after revision for publication on 19 May 2009.

DOI: 10.1243/09544062JMES1506

Abstract: The quantity and quality of the mesh elements are both significant factors for guaranteeing computational precision in the finite-element method (FEM). In most mesh simplification algorithms, the geometric error was thought to be the most important issue. However, the mesh quality was rarely taken into consideration. In this article, a finite triangular surface mesh simplification algorithm is proposed, in which the vertex dispersion is introduced to represent the local geometrical feature, and then either edge collapse or face collapse is carried out consequently. The aspect of the newly created face is taken into account and the position of the newly created vertex is obtained by solving an over-determined system of linear equations with respect to the aspect ratios of the newly created faces; thereupon the simplified mesh quality is improved. To obtain a further simplification and to reduce the errors in the succeeding FEM analysis, surface fitting is adopted on the surfaces with large curvature. Simplification cases had been performed in comparison with the quadric error metric method, and the results show that the definition of the local mesh density is more reasonable for the FEM analysis with the same simplification ratio while the present simplification algorithm is employed. Moreover, the mesh quality can be greatly improved on the surface with large curvature. A set of FEM numerical experiments of polymer injection moulding simulation had also been performed to determine the effect of the presented simplification algorithm on the FEM analysis. The numerical results show that the error of the injection pressure can be limited within 4 per cent, while the simplification percentage reaches 75 per cent.

Keywords: triangular surface mesh, mesh simplification, finite-element method, surface fitting, geometrical feature, mesh optimization

1 INTRODUCTION

The finite-element method (FEM) is currently the most popular approach used in structural analysis, product modelling, process simulation, and other fields. The quantity and quality of the mesh are significant factors for guaranteeing the computational precision in this approach. A model with more elements would gain a better result with high precision in general, whereas a model with refined meshes might even dramatically obtain a worse result sometimes because of elements with execrable quality. Moreover, the increasing size of the mesh consequently takes more computational time. The recent evolution of the computer-aided design technology allows the engineer to model very complex and expensive structures that produce huge surface meshes with hundreds of thousands of elements. The increasing size of the mesh still surpasses improvements in hardware performance. A simplified mesh that meets both computational precision and computational time is required.

Triangle is a kind of primitive and elementary element of the surface mesh, which is widely used in the FEM. Various surface mesh simplification algorithms have been reported in the last decade, most of which focused on triangular surface mesh. Heckbert and Garland [**1**] described an algorithm that simplified polygonal models using a quadric error metric (QEM), which is directly related to surface

**Corresponding author: State Key Laboratory of Materials Processing and Mold and Die Technology, Huazhong University of Science and Technology, 1037 Luoyo Road, Wuhan, Hubei 430074, People's Republic of China.*
email: hmzhou@hust.edu.cn; hmzhou@263.net

curvature. Hoppe [**2**] proposed an improved QEM for simplifying meshes with attributes. Lu *et al.* [**3**] proposed a mesh simplification method using half-edge collapse based on QEM. Kim *et al.* [**4**] and Jiang *et al.* [**5**] considered the curvature normal in their mesh simplification method. Borouchaki and Frey [**6**] introduced two tolerance areas to preserve the geometry of the surface in their mesh simplification method using iteratively removing and optimizing all mesh edges. Tang *et al.* [**7**] used surface moments and volume moments as the simplification metrics in edge collapse procedure. Date *et al.* [**8**] proposed a new triangular finite-element mesh generation method based on simplification of high-density mesh and adaptive level-of-detail.

A common goal of all the mesh simplification methods is how to preserve the features and properties of the initial model as far as possible within a limited face number. Besides, a finite surface mesh simplification method should consider:

(a) how to avoid sharp and narrow triangular faces and improve the quality of the simplified mesh, which are significant issues in the FEM;
(b) how to determine the local mesh density within an acceptable error tolerance, namely, the local mesh density should vary along with the variation of the unknown physical quantities.

Most of the mesh simplification methods mentioned above preserve well the geometry features of the initial model by various means, such as QEM, surface moments, and so on, whereas the qualities of the newly created faces are rarely taken into consideration. Moreover, the surface curvature is thought to be the only representation of the variation of the unknown physical quantities and a high mesh density is generally defined where the surface curvature is large. However, small-scale shape details, such as fillets or chamferings, which are negligible sometimes to the consequent FEM analysis, occupy a large proportion of the element number according to the great curvatures on these areas.

On the other hand, Yau and Menq [**9**] proposed a unified least-squares approach to the best fit of geometric feature. Sullivan *et al.* [**10**] performed parameterized surface fitting using a combination of constrained optimization and non-linear least-squares estimation techniques to minimize the mean-squared geometric distance between original vertexes and fitted surface. Lukács *et al.* [**11**] proposed a set of methods for the least-squares fitting of spheres, cylinders, cones, and tori to three-dimensional point data. If a surface fitting can be employed to describe the original model surface on the areas with a large curvature, the new vertices in the simplification algorithm can be projected onto the fitted surface and thus the geometric error can be greatly decreased. Furthermore, additional information can be obtained from the parameters of the fitted surface and a flexible mesh density can be employed in these areas.

In this article, we propose a finite triangular surface mesh simplification method with local geometrical feature recognition. First, vertex dispersion, a definition is proposed to represent the local geometric feature of the mesh surface, and it is used to control the local density of the finite triangular surface mesh (section 2). In succession, edge collapse and triangle collapse are employed to simplify the mesh according to the vertex dispersions and the amended shortest edge length. The positions of the newly created vertices during the simplification process are determined by least-squares solutions of the equations about the newly created faces aspect ratios to improve the mesh quality, rather than by the least error metric of the model geometry, such as QEM. In addition, surface fitting is imported to obtain a flexible mesh density definition and decreases the geometric error of the simplified model on areas where the surface curvature is great (section 3). Finally, several cases of the mesh simplification method presented in comparison with the QEM method are provided (section 4).

2 FEATURE REPRESENTATIONS

The first task of mesh simplification is describing features from mass discrete faces. Hamann [**12**] proposed a geometrical feature description using discrete curvature of vertices. Jiang *et al.* [**5**] and Dyn *et al.* [**13**] considered average vertex normal vector as a characterizing definition of geometrical feature. Average vertex normal is too simple to denote the geometrical feature, while discrete curvature of vertices is an excellent one but it takes a long computation time. In this article, vertex dispersion is proposed to describe local geometric features and determine the local mesh density. It is defined as a standard deviation of the average vertex normal vector and its adjacent face normal vectors, expressed as

$$A_\sigma = \sqrt{\sum_{i=0}^{n} \frac{[\theta(\boldsymbol{n}_0, \boldsymbol{n}_{\mathrm{f}i}) - \overline{\theta}]^2}{n}} \tag{1}$$

where A_σ is the vertex dispersion, $\overline{\theta}$ is the average angle between the average normal vector of the vertex and its adjacent face normal vectors, and can be calculated by

$$\overline{\theta} = \sum_{i=0}^{n} \frac{\theta(\boldsymbol{n}_0, \boldsymbol{n}_{\mathrm{f}i})}{n} \tag{2}$$

where n is the number of the vertex's adjacent faces, $\boldsymbol{n}_{\mathrm{f}i}$ is the vertex's ith adjacent face normal vector, $\boldsymbol{n}_0$ is the average normal vector of the vertex, and $\theta(\boldsymbol{n}_0, \boldsymbol{n}_{\mathrm{f}i})$

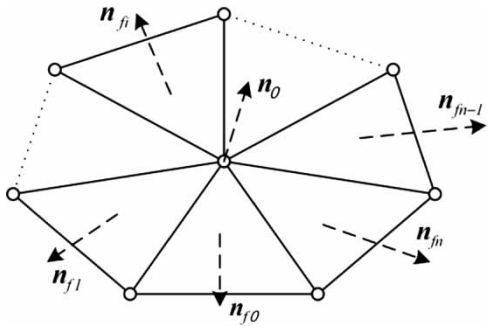

Fig. 1 Vertex normal vector and its adjacent face normal vectors

is the angle between $\boldsymbol{n}_0$ and $\boldsymbol{n}_{\mathrm{f}i}$ (see Fig. 1). $\boldsymbol{n}_0$ is calculated by area weighted average of normal vectors, expressed as

$$\boldsymbol{n}_0 = \frac{\boldsymbol{n}_\mathrm{s}}{|\boldsymbol{n}_\mathrm{s}|} \tag{3}$$

$$\boldsymbol{n}_\mathrm{s} = \frac{\sum_{i=0}^{n}(A_{\mathrm{f}i}\boldsymbol{n}_{\mathrm{f}i})}{\sum_{i=0}^{n}A_{\mathrm{f}i}} \tag{4}$$

where $A_{\mathrm{f}i}$ is the area of the vertex's ith adjacent face. The angle $\theta(\boldsymbol{n}_0, \boldsymbol{n}_{\mathrm{f}i})$ is calculated by

$$\theta(\boldsymbol{n}_0, \boldsymbol{n}_{\mathrm{f}i}) = \arccos\left[\frac{(\boldsymbol{n}_0 \cdot \boldsymbol{n}_{\mathrm{f}i})}{(|\boldsymbol{n}_0|\,|\boldsymbol{n}_{\mathrm{f}i}|)}\right] \tag{5}$$

The vertex dispersion represents the standard deviation of the vertex normal vector and its adjacent face normal vectors. Since the standard deviation is used as a measure of the dispersion or variation in a distribution, the vertex dispersion reflects the intensity of the surface normal change indeed. The greater the vertex dispersion, the more intense the change in normal at the vertex, which means that the vertex might be on an edge or on a transitional area; the smaller the vertex dispersion, the more moderate change in normal at the vertex, which means that the vertex might be on a large plane.

Non-uniform grid is adopted in the FEM, i.e. a refined grid is required on a surface with a large curvature and a reduced grid is applied on a large flat region. The longest edge length of the triangular element is a type of definition of the mesh density, and it is usually used as a control parameter in mesh generation. However, the shortest length of the triangular element is a factor to determine the simplified target, and it is usually used in mesh simplification. In this article, the vertex dispersion and the shortest length are both factors for adjusting mesh density. An amended shortest length is defined to determine the edge or face that should be collapsed. It can be calculated by

$$L_{\min} = \mathrm{e}^{k_\mathrm{m}A_\sigma}\,\min(L_i)_{i=0,1,2} \tag{6}$$

where $L_{\min}$ is the amended shortest length, e denotes the natural logarithm, A_σ is the vertex dispersion, L_i is the length of the ith edge of the triangle, and k_m is an adjustment factor.

3 SIMPLIFICATION ALGORITHM

3.1 Triangular element quality metric

A high-quality triangular element is considered as a triangle that is similar to an equilateral triangle in FEM, and a low-quality triangular element is namely a long and narrow triangle. Aspect ratio is used for measurement of the element quality and it can be defined in various ways [**1**, **14**, **15**], most nearly equivalent. An optional triangle's aspect ratio is its longest edge length divided by its corresponding height, calculated by

$$Ra = \alpha\frac{\max(L_i)_{i=0,1,2}}{H_i} \tag{7}$$

where Ra is the aspect ratio, L_i is the length of the ith edge, H_i is the corresponding height, and α is a constant factor, which is $2/\sqrt{3}$ for triangular elements. The minimum of Ra equals 1 if and only if the triangle is a regular one. It is >1 in other cases. The larger the Ra the worse the quality.

3.2 Simplification algorithm steps

First vertex dispersion of each node and aspect ratio of each face are calculated, and then all faces can be arranged into a queue in ascending order of their amended shortest edge length. The front face in the queue is then taken out and is eliminated either by edge collapse or face collapse according to its vertex dispersions and its aspect ratio. During this elimination, the initial positions of the new created vertices are optimized points on the original surface. Surface fitting is adopted where vertex dispersion is large and the optimized points are finally projected onto the fitted surfaces. Then the next face in the queue is taken out and this process is repeated until the input simplification criteria are satisfied. The steps of the mesh simplification algorithm are described as following.

Step 1: Input to the algorithm: Simplification ratio, maximum error limits, and planarity threshold are acquired from the input. The length of the shortest edge after simplification is estimated according to simplification ratio and is denoted as L_0.

Step 2: Pre-processing: All vertex dispersions are calculated respectively according to equation (1). Aspect ratios of all faces are calculated. Faces are then arranged into a queue in ascending order of their amended shortest edge length $L_{\min}$.

Step 3: Edge collapsing and face collapsing: The front face in the queue is taken out and either edge collapse

or face collapse is adopted. According to its vertex dispersions and its amended shortest edge length, the face should be one of the following four cases.

Case 1: Where all vertex dispersions of the face are less than a threshold τ_{A_σ}, it is considered to be on a near-planar surface or a surface with a small curvature. If the amended shortest edge length $L_{\min}$ is $< k_1 L_0$ ($k_1 \approx 1.5$), the face is eliminated using face collapse if its aspect is fine, otherwise the shortest edge is collapsed.

Case 2: Where there is only one vertex whose vertex dispersion is greater than the threshold τ_{A_σ}, the vertex is likely on an edge while the other two are not. The edge, which is opposite to the vertex, is collapsed if it is $< L_0$.

Case 3: Where there are two vertices whose vertex dispersion is greater than the threshold τ_{A_σ}, the two vertices are likely on an edge while the remaining one is not. If the amended shortest edge length $L_{\min}$ is $< k_2 L_0 (k_2 \approx 0.75)$, the edge is collapsed while it is just the edge between the two vertices; otherwise, the two vertices on the shortest edge are clustered and the new created vertex is one of them whose vertex dispersion is greater.

Case 4: Where all vertex dispersions of the face are greater than the threshold τ_{A_σ}, it is considered to be on a transitional surface. If the shortest edge length of the face is $< k_3 L_0$ ($k_3 \approx 0.5$), the face is collapsed while its aspect is fine, otherwise the shortest edge is collapsed.

In case 1–3, the new created vertices are optimized points during face collapsing or edge collapsing. In case 4, surface fitting is applying on nearby area where the face or edge is collapsed. The optimized points are finally projected points on the fitted surface.

Step 4: Post-processing: If the input simplification criteria are satisfied, the algorithm is finished. Otherwise, the vertex dispersions of the newly created vertex and its adjacent vertices should be recalculated. The affected items in the queue and the length of their amended shortest edge $L_{\min}$ should be updated meanwhile, then return to step 3.

3.3 New vertex position calculation

As soon as the vertex dispersions have been calculated on all vertices, the surface can be divided into three types: planar or near-planar areas, curved surfaces with a small curvature, and curved surfaces with a large curvature. There is no need to compute the fitting surfaces for planar or near-planar areas and curved surfaces with a small curvature, since the simplification ratio can be reached and the geometric figure can be well preserved if only edge collapse or face collapse is employed on these surfaces. The stress of our simplification algorithm on these surfaces is how to determine the position of the newly created vertex, which would improve the aspects of the newly created faces.

There are lots of approaches reported to compute the coordinates of the newly created vertices, in which the most effort is to obtain the most approximate points on the origin surface using distance metrics or angle metrics [**3**, **4**]. However, the quality of the newly created faces is often ignored.

As shown in Fig. 2, P_0, P_1, and P_2 are the vertices of the collapsed face (a) and P_0, P_1 are the vertices of the collapsed edge (b), $P_{N0}, P_{N1}, \ldots, P_{Nn-1}, P_{Nn}$ are their adjacent vertices, and P_b is the undetermined optimal point on the origin mesh surface. A criterion for determining the optimal point is that aspect ratio *Ra* of the newly created faces should be close to 1 as far as possible. Since the longest edge of the newly created faces is uncertain due to P_b being undetermined, the aspect ratio *Ra* of the newly created faces is unable to be exactly calculated. Suppose $P_{Ni}P_{Ni+1}$ is the longest edge of the newly created face $P_{Ni}P_{Ni+1}P_b$ and replace the height on the edge $P_{Ni}P_{Ni+1}$ by the length of the midline on the edge $P_{Ni}P_{Ni+1}$. The calculation of the aspect ratios of the new created faces can be replaced by

$$Ra_i \approx \alpha \frac{L_i'}{H_i'} = 1, \quad i = 0, 1, \ldots, n \tag{8}$$

where L_i' is the length of the edge $P_{Ni}P_{Ni+1}$ and H_i' is the corresponding midline length, which can be obtained by

$$H_i' = \sqrt{(x - x_{ci})^2 + (y - y_{ci})^2 + (z - z_{ci})^2} \tag{9}$$

where x_{ci}, y_{ci}, and z_{ci} are the coordinates of centre point P_{ci}; and x, y, and z are the coordinates of point P_b. Substituting equation (9) into equation (8), we can obtain an over-determined system of linear equations

$$\begin{cases} a_1 x + b_1 y + c_1 z = d_1 \\ \vdots \\ a_i x + b_i y + c_i z = d_i \qquad i = 0, 1, \ldots, n \\ \vdots \\ a_n x + b_n y + c_n z = d_n \end{cases} \tag{10}$$

where $a_i = x_i - x_0$, $b_i = y_i - y_0$, and $c_i = z_i - z_0$, $d_i = [(L_i'/\alpha)^2 - (L_0'/\alpha)^2 + x_i^2 + y_i^2 + z_i^2 - (x_0^2 + y_0^2 + z_0^2)]/2$. It can be written in the form of matrix expressions as

$$\mathbf{A}\boldsymbol{X} = \boldsymbol{Y} \tag{11}$$

where

$$\mathbf{A} = \begin{bmatrix} a_1 & b_1 & c_1 \\ \vdots & & \\ a_i & b_i & c_i \\ \vdots & & \\ a_n & b_n & c_n \end{bmatrix}, \quad \boldsymbol{X} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \quad \boldsymbol{Y} = \begin{bmatrix} d_1 \\ \vdots \\ d_i \\ \vdots \\ d_n \end{bmatrix}$$

The solution of equation (11) is

$$\boldsymbol{X} = (\mathbf{A}^{\mathrm{T}}\mathbf{A})^{-1}\mathbf{A}^{\mathrm{T}}\boldsymbol{Y} \tag{12}$$

where $\boldsymbol{X}$ is the coordinates vector of the undetermined optimal point P_{b}. Besides, it should also be considered that the optimal point must be on the edge collapsed or on the face collapsed. Thus the co-ordinates of the point P_{b} can be defined by parameter equations of the coordinates of P_0, P_1, and P_2. If edge collapse is employed, the parameter equation is

$$\boldsymbol{X} = \boldsymbol{X}_{p_0} + t\boldsymbol{X}_{p1}, \quad t \in [0, 1] \tag{13}$$

or if face collapse is employed, it is

$$\boldsymbol{X} = \xi_0\boldsymbol{X}_{p_0} + \xi_1\boldsymbol{X}_{p1} + \xi_2\boldsymbol{X}_{p2}, \quad \xi_0, \xi_1, \xi_2 \in [0, 1] \tag{14}$$

where $\boldsymbol{X}_{p_0}$, $\boldsymbol{X}_{p1}$, and $\boldsymbol{X}_{p2}$ are the coordinates of P_0, P_1, and P_2 respectively, t, ξ_0, ξ_1, and ξ_2 are undetermined parameters. Substituting equation (13) or equation (14) into equation (12), t or ξ_0, ξ_1, and ξ_2 can be obtained according to the solution of equation (12) and thus the coordinates of the new vertex are determined. Since the new vertex position is determined by solving an over-determined system of linear equations regarding the aspect ratios of the newly created faces, the aspect ratios of the new created faces would approach to 1 as far as possible and thus the new vertex position is optimized.

3.4 Surface fitting

The curved surfaces with a large curvature, most of which are generally small-scale shape details in industrial products, occupy a large proportion of the face number (see Fig. 3). However, they can be neglected in many FEM applications, such as flow simulation of injection moulding. Most simplification algorithms reported stressed how to keep the geometric feature on these areas, and consequently the mesh density on these areas is too high according to their curvature.

In many cases, at least in most industrial products, a great proportion of the small-scale shape details are

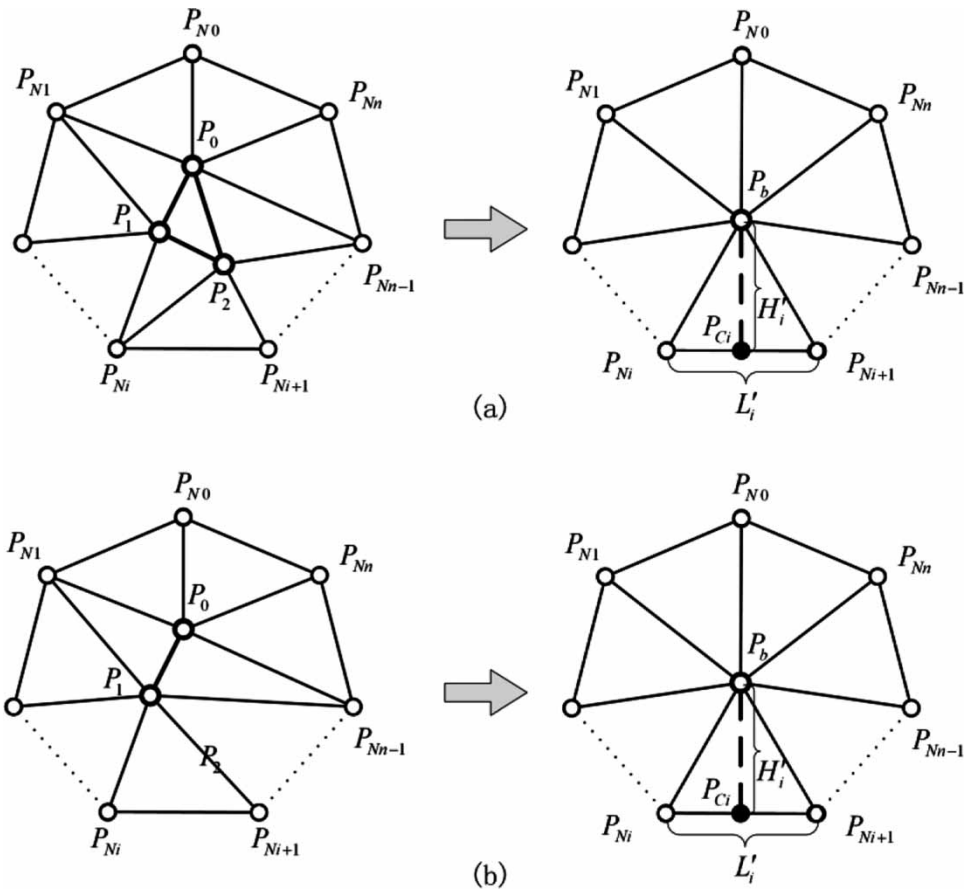

Fig. 2 Illustration of the determination of the optimal point: (a) face collapse and (b) edge collapse

composed of fillets and chamferings, which are usually used to connect two adjacent large surfaces and to occupy a large proportion of the face number as mentioned above (for example the proportion of the face number on fillets and chamferings up to about 38 per cent in the product, which is shown in Fig. 3). Since a fillet or chamfering can be approximately regarded as a part of a cylinder or sphere, sphere fitting and cylinder fitting are performed in our works. For a more general situation, a general surface fitting can be performed and Jiang and Wang [**16**] proposed an algorithm of NURBS surface fitting from discrete vertices.

3.4.1 *Sphere fitting*

In case 4 of step 3, where the vertex dispersion is large, face collapse is applied if the aspect of the eliminated face is fine. The nearby area of the eliminated face is likely to be a spherical surface or an approximate spherical surface, as shown in Fig. 3. Thereupon a trial sphere fitting is applied.

Assuming the centre coordinates of the fitted sphere are (x_0, y_0, z_0), the radius of the fitted sphere is R, the shortest distance square d^2 between the point (x, y, z) and the sphere surface can be expressed as

$$d^2 = (x-x_0)^2 + (y-y_0)^2 + (z-z_0)^2 + R^2 - 2R\sqrt{(x-x_0)^2 + (y-y_0)^2 + (z-z_0)^2} \quad (15)$$

Now, let $d' = d^2$, since $\sqrt{(x-x_0)^2+(y-y_0)^2+(z-z_0)^2} \approx R$, equation (15) can be written in the following form

$$d' = (x-x_0)^2 + (y-y_0)^2 + (z-z_0)^2 - R^2 \quad (16)$$

The polynomial expression of equation (16) is

$$d' = x^2 + y^2 + z^2 + a_1x + a_2y + a_3z + a_4 \quad (17)$$

where a_1, a_2, a_3, and a_4 are undetermined coefficients. Because d' might be negative, it should be squared in applying the least-square method. Squaring both sides of equation (17), the equivalent distance used in the least-squared method is acquired, which is denoted as d'' and can be calculated by the following equation

$$d'' = (x^2 + y^2 + z^2 + a_1x + a_2y + a_3z + a_4)^2 \quad (18)$$

The sphere fitting is illustrated in Fig. 4, P_0, P_1, P_2 are vertices of the collapsed face, the highlighted curve indicates the fitted spherical surface, and P_p is the projected point of P_b on it. The adjacent faces of triangle $P_0P_1P_2$ are taken into account in sphere fitting. The error between P_p and the original surface is defined as

$$\text{Err} = |\overrightarrow{\boldsymbol{P}_p\boldsymbol{P}_b} \cdot \boldsymbol{n}| \quad (19)$$

where Err is the error, $\overrightarrow{\boldsymbol{P}_p\boldsymbol{P}_b}$ is the vector between P_p and P_b, and $\boldsymbol{n}$ is the unit normal vector of the triangle $P_0P_1P_2$. The error Err denotes the distance between the P_p and the triangle $P_0P_1P_2$. If the error is under the tolerance limit, the newly created vertex's position is P_p, otherwise P_b.

3.4.2 *Cylinder fitting*

Fillets and chamferings are characterized by lots of long and narrow faces whose aspects are poor (see Fig. 3). A trial cylinder fitting is applied if the aspect of the eliminated face is poor in case 4 of step 3. The shortest edge of the eliminated face is collapsed in this circumstance.

Assuming the radius of the fitted cylinder is R, its centre axis is defined as

$$\boldsymbol{p} = \boldsymbol{p}_0 + t\boldsymbol{s} \quad (20)$$

where $\boldsymbol{p}_0$ is an arbitrary point on the centre axis, whose coordinates are (x_0, y_0, z_0); $\boldsymbol{s}$ is the direction of the centre axis and its direction vector is (l, m, n); t is a variable. Then the shortest distance d between the point

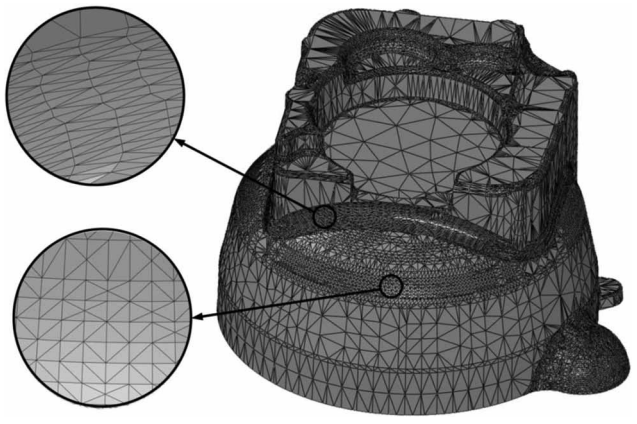

Fig. 3 Different types of triangular grid on the areas where the vertex dispersion is large

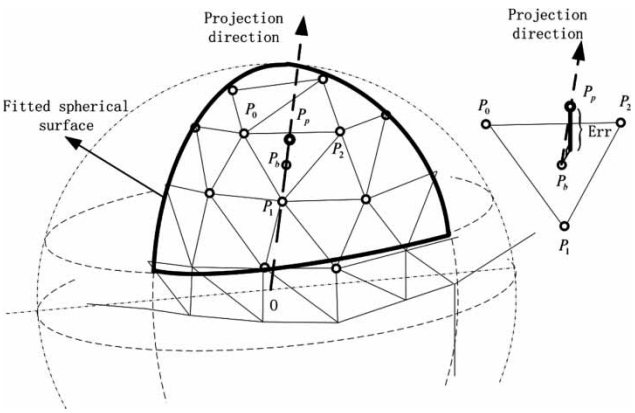

Fig. 4 Sphere fitting

(x, y, z) and the fitting cylinder surface is expressed as

$$d = \frac{\sqrt{\begin{array}{l}[m(z-z_0)-n(y-y_0)]^2+[n(x-x_0)\\ -l(z-z_0)]^2+[l(y-y_0)-m(x-x_0)]^2\end{array}}}{\sqrt{(l^2+m^2+n^2)}} - R \tag{21}$$

Similar to sphere fitting, squaring both sides of equation (17) and letting $d' = d^2$, equation (15) can be rewritten as

$$\begin{aligned} d' = {} & [m(z-z_0)-n(y-y_0)]^2+[n(x-x_0) \\ & -l(z-z_0)]^2+[l(y-y_0)-m(x-x_0)]^2 - R^2 \end{aligned} \tag{22}$$

The polynomial expression of equation (22) is

$$\begin{aligned} d' = {} & a_1x^2+a_2y^2+a_3z^2+a_3xy+a_4yz+a_5xz \\ & +a_6x+a_7y+a_8z+a_9 \end{aligned} \tag{23}$$

where $a_1, a_2, \ldots, a_9$ are undetermined coefficients. However, because each item of the right side of equation (23) contains an undetermined coefficient, there would be only zero solution in the subsequent least squaring. On the assumption of $a \neq 0$, dividing both sides of equation (23) by a and then squaring both sides, the equivalent polynomial used in least squaring can be defined as

$$\begin{aligned} d'' = {} & (x^2+b_1y^2+b_2z^2+b_3xy+b_4yz+b_5xz+b_6x \\ & +b_7y+b_8z+b_9)^2 \end{aligned} \tag{24}$$

where $b_1, b_2, \ldots, b_9$ are undetermined coefficients; if $a = 0$, it leads to m, $n = 0$ and $l = 0$; directly squaring both sides of equation (23) and simplifying, the equivalent polynomial used in least squaring is defined by

$$d'' = (y^2+z^2+c_1y+c_2z+c_3)^2 \tag{25}$$

where c_1, c_2, and c_3 are undetermined coefficients. Equations (24) and (25) are both used in the least-squared method at the same time. The cylinder fitting is illustrated in Fig. 5. The highlighted curve is the fitted cylindrical surface, P_0 and P_1 are vertices of the collapsed edge, and P_p is the projected point of P_b on it. The error between P_p and the original surface is defined as

$$\mathrm{Err} = |\overrightarrow{\boldsymbol{P}_p\boldsymbol{P}_b} \cdot n| \tag{26}$$

where Err is the error, $\overrightarrow{\boldsymbol{P}_p\boldsymbol{P}_b}$ is the vector between P_p and P_b, and $\boldsymbol{n}$ is the unit normal vector of the edge P_0P_1

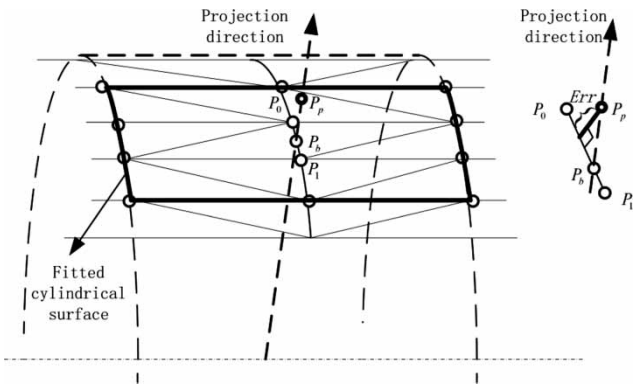

Fig. 5 Cylinder fitting

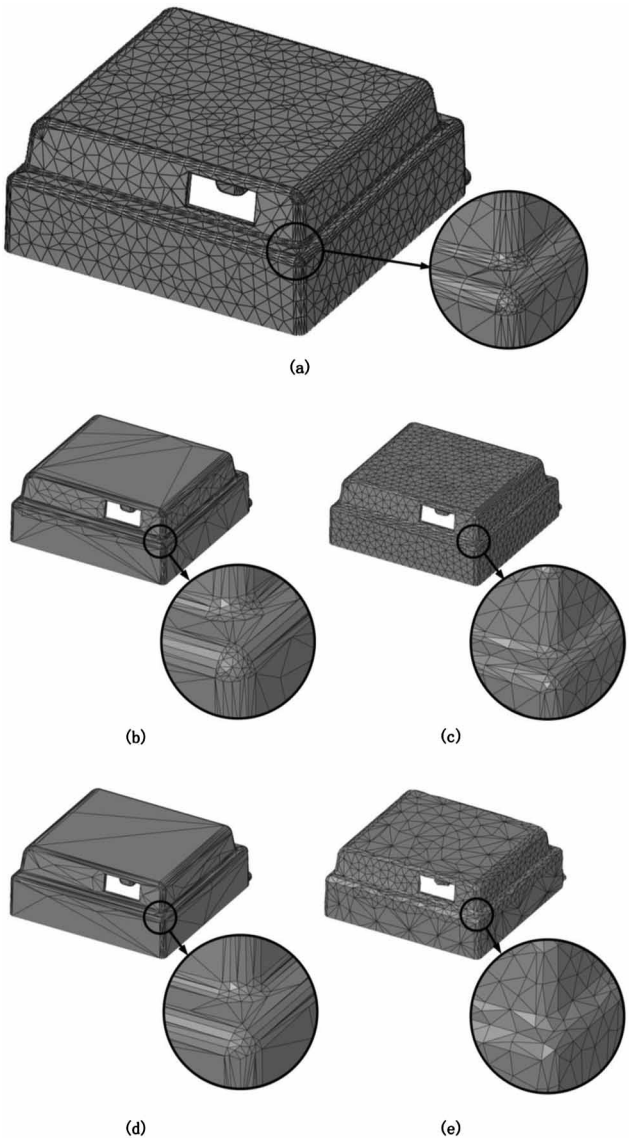

Fig. 6 The simplification results of case 1: (a) the initial mesh; (b) the mesh with a simplification percentage of 50 per cent using the QEM method; (c) the mesh with a simplification percentage of 50 per cent using the present method; (d) the mesh with a simplification percentage of 75 per cent using the QEM method; and (e) the mesh with a simplification percentage of 75 per cent using the present method

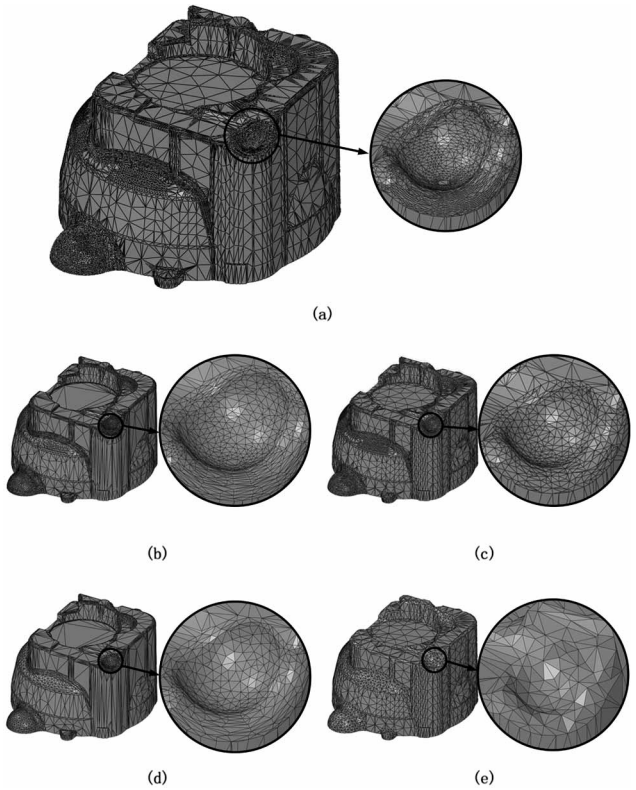

Fig. 7 The simplification results of the case 2: (a) the initial mesh; (b) the mesh with a simplification percentage of 50 per cent using the QEM method; (c) the mesh with a simplification percentage of 50 per cent using the present method; (d) the mesh with a simplification percentage of 75 per cent using the QEM method; and (e) the mesh with a simplification percentage of 75 per cent using the present method

and can be calculated by

$$\boldsymbol{n} = \frac{\boldsymbol{n}_1 + \boldsymbol{n}_2}{|\boldsymbol{n}_1 + \boldsymbol{n}_2|} \tag{27}$$

where $\boldsymbol{n}_1$, $\boldsymbol{n}_2$ are the vertex normal vectors of P_0 and P_1, respectively. If the error is below the tolerance limit, the new created vertex's position is P_p, otherwise P_b.

4 CASE STUDIES

All cases were performed on a standard Windows PC with a 2 GHz Intel E4400 processor, 2 GB of memory, and implemented in C++. The precision tolerance was limited within 0.1 mm. The initial meshes and the simplified meshes of the cases using the present simplification algorithm are shown in Figs 6 and 7 in comparison with the QEM method. The areas containing a lot of small-scale shape details occupy a great proportion of the number of faces, especially in industrial products, as shown in Figs 6(a) and 7(a). The number of faces in Fig. 7(e) is deduced to 25 per cent to the initial mesh. The aspects of the newly created faces are optimized during the simplifying process. For example the aspects of the faces on the long and narrow areas (the magnified region) are awful in Fig. 6(a), whereas they are improved in Figs 6(c) and (e). In comparison with the QEM method, the definition of the local mesh density in the present simplification algorithm is more reasonable. Moreover, the mesh quality of the simplified mesh is better than that of the QEM method, especially in the areas with small-scale shape details, as shown in Figs 6(d) and (e). Table 1 shows the mesh quality of the model using the present simplification algorithm. The mesh quality in Table 1 is defined as the percentage of faces whose aspect ratio is <20.

Table 1 Several mesh properties of the model using the present simplification algorithm

	Case 1: the shell of a relay			Case 2: the cover of a valve		
Simplification percentage	Number of faces	Number of vertices	Mesh quality	Number of faces	Number of vertices	Mesh quality
0	16 200	8096	94.4	55 170	27 588	98.5
50	8292	4142	99.3	28 242	14 124	99.3
75	4824	2408	99.5	11 390	5697	99.6

Table 2 Finite-element numerical results

	Case 1			Case 2		
Simplification percentage	Solution time (s)	Pressure (MPa)	Error (%)	Solution time (s)	Pressure (MPa)	Error (%)
0	354	20.7416	0	2480	9.211 82	0
20	240	20.9303	0.91	1816	9.258 61	0.51
35	170	21.0359	1.42	1255	8.928 94	−3.07
50	138	20.8379	0.46	816	9.133 33	−0.85
68	60	21.4798	3.56	440	9.228 76	0.18
75	51	20.1800	2.71	226	9.385 14	1.88

A set of numerical experiments had been performed to determine the effect of our simplification algorithm on FEM and are shown in Table 2. The test program is the flow module of a self-developed polymer injection moulding simulation system named HSCAE. The process conditions are uniform as following: polymer material is ABS, injection temperature is 230 °C, and mould temperature is 40 °C. The cavity pressure at the moment when injection finished is used for measuring the errors. It can be seen that the solution time will be reduced more greatly if the face number decreases. When the simplification percentage is 75 per cent, the solution time will be only about 1/7 to origin, while the error of the injection pressure is limited within 4 per cent and this means that the presented simplification algorithm meets the engineering applications.

5 CONCLUSIONS

So far, a finite triangular surface mesh simplification method has been presented. By introducing vertex dispersion, either face collapse or edge collapse could be applied according to the local geometric features. The mesh quality is considered as an important factor and the position of the newly created vertex is optimized by solving an over-determined system of linear equations regarding the aspect ratios of the newly created faces. To obtain a further simplification and meanwhile reduce the geometric errors, surface fitting is adopted on the areas with large curvature. In comparison with the QEM method, the definition of the local mesh density is more reasonable for the FEM analysis with the same simplification ratio while using the present simplification algorithm. Moreover, the mesh quality can be greatly improved on the surface with large curvature. A set of FEM numerical experiments of polymer injection moulding simulation had been also performed to determine the effect of the presented simplification algorithm on the FEM analysis. The numerical results show that the error of the injection pressure can be limited within 4 per cent while the simplification percentage reaches 75 per cent.

ACKNOWLEDGEMENTS

The authors would like to acknowledge financial support from the National Natural Science Foundation Council of China (Grant no. 50675080 and 50875095).

REFERENCES

1 **Heckbert, P. S.** and **Garland, M.** Optimal triangulation and quadric-based surface simplification. *Comput. Geom.*, 1999, **14**(1–3), 49–65.

2 **Hoppe, H.** New quadric metric for simplifying meshes with appearance attributes. In Proceedings of the IEEE Visualization 99 Conference, San Francisco, 1999, pp. 59–66.

3 **Lu, W., Zeng, D.,** and **Pan, J.** QEM-based mesh simplification with effective feature-preserving. *Lect. Notes Comput. Sci.*, 2007, **4563**, 122–131.

4 **Kim, S. J., Kim, C. H.,** and **Levin, D.** Surface simplification using a discrete curvature norm. *Comput. Graph.*, 2002, **26**(5), 657–663.

5 **Jiang, S. P., Zhou, T. M.,** and **Dai, Y.** Mesh simplification based on normals. *Chin. J. Comput.*, 1999, **22**(10), 1074–1079.

6 **Borouchaki, H.** and **Frey, P. J.** Simplification of surface mesh using Hausdorff envelope. *Comput. Methods Appl. Mech. Eng.*, 2005, **194**(48–49), 4864–4884.

7 **Tang, H., Shu, H. Z., Dillenseger, J. L., Bao, X. D.,** and **Luo, L. M.** Moment-based metrics for mesh simplification. *Comput. Graph.*, 2007, **31**(5), 710–718.

8 **Date, H., Kanai, S., Hokkaido, T. K.,** and **Nishigaki, I.** Mesh simplification and adaptive LOD for finite element mesh generation. In Proceedings of the Ninth International Conference on *Computer aided design and computer graphics*, IEEE Computer Society, Washington, DC, USA, 2005, pp. 339–344.

9 **Yau, H.-T.** and **Menq, C.-H.** A unified least-squares approach to the evaluation of geometric errors using discrete measurement data. *Int. J. Mach. Tools Manuf.*, 1996, **36**(11), 1269–1290.

10 **Sullivan, S., Sandford, L.,** and **Ponce, J.** Using geometric distance fits for 3-D object modeling and recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 1994, **16**(12), 1183–1196.

11 **Lukács, G., Martin, R.,** and **Marshall, D.** Faithful least-squares fitting of spheres, cylinders, cones and tori for reliable segmentation. *Lect. Notes Comput. Sci.*, 1998, **1406**, 671–686.

12 **Hamann, B.** A data reduction scheme for triangulated surfaces. *Comput. Aided Geom. Des.*, 1994, **11**(2), 197–214.

13 **Dyn, N., Hormann, K., Kim, S. J.,** and **Levin, D.** Optimizing 3D triangulations using discrete curvature analysis. *Math. Methods Curves Surf.*, 2001, 135–146.

14 **D'Azevedo, E. F.** and **Simpson, R. B.** On optimal interpolation triangle incidences. *SIAM J. Sci. Stat. Comput.*, 1989, **10**, 1063.

15 **Rippa, S.** Long and thin triangles can be good for linear interpolation. *SIAM J. Numer. Anal.*, 1992, **29**(1), 257–270.

16 **Jiang, D.** and **Wang, L. C.** An algorithm of NURBS surface fitting for reverse engineering. *Int. J. Adv. Manuf. Technol.*, 2006, **31**(1), 92–97.

APPENDIX

Notation

$a_1 - a_9, b_1 - b_9,$ $c_1 - c_3, t, \xi_0 \sim \xi_2$	undetermined coefficients
a_i, b_i, c_i, d_i	constants
$\mathbf{A}$	coefficient matrix
A_{fi}	area of a triangle

A_σ	vertex dispersion	$\boldsymbol{n}_0, \boldsymbol{n}_{\mathrm{f}i}, \boldsymbol{n}_{\mathrm{s}}, \boldsymbol{n}_1, \boldsymbol{n}_2, \boldsymbol{n}$	normal vectors
d	theoretic distance between two points	R	sphere radius
d'	calculated distance square between two points	Ra	aspect ratio
d'''	equivalent distance in least square	$x, y, z, x_0, y_0, z_0, x_i, y_1, z_i$	coordinate components
Err	error	$\boldsymbol{X}$	coordinates vector
H_i, H_i'	height and midline length of a triangle	$\boldsymbol{Y}$	constant vector
$k_{\mathrm{m}}, k_1, k_2, k_3$	constant coefficients	α	constant factor
L_0	length of the shortest edge	θ	angle between two vector
L_i, L_i'	length of an edge	$\overline{\theta}$	average angle
n	number of a vertex's adjacent elements	τ_{A_σ}	threshold of the vertex dispersion