Hexahedral Mesh Quality Improvement with Geometric Constraints

wei peng (✉ weipeng@jiangnan.edu.cn)

Jiangnan University <https://orcid.org/0000-0002-6605-1519>

Xinguang Wu

Xi'an North Hui'an Chemical Industry Company

Yidong Bao

Nanjing University of Aeronautics and Astronautics

Chaoyang Zhang

Jiangnan University

Weixi Ji

Jiangnan University

Original Article

Keywords: Hexahedral mesh, quality improvement, geometric constraints, feature preserve

Posted Date: June 14th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-586657/v1>

License: This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Title page

Hexahedral mesh quality improvement with geometric constraints

Peng Wei, born in 1984, is currently an associate professor at College of mechanical engineering, Jiangnan University, Wuxi, China. He received his doctor degree from Nanjing University of Aeronautics and Astronautics, China, in 2012. His research interests include digital manufacturing and finite element method.

Tel: +86-510-85910251; E-mail: weipeng@jiangnan.edu.cn

Wu Xinguang, born in 1980, is currently an engineer at Xi'an North Hui'an Chemical Industry Company, China. His research interests include numerical simulation.

E-mail: wuxinguang555@163.com

Bao Yidong, born in 1976, is currently an associate professor at College of Mechanical and Electrical Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, China. He received his doctor degree from Jilin University, China, in 2005.

baoyd@nuaa.edu.cn

Zhang Chaoyang, born in 1987, is currently an assistant professor at College of mechanical engineering, Jiangnan University, Wuxi, China. He received his doctor degree from Xi'an Jiaotong University, China, in 2016.

cyzhang@jiangnan.edu.cn

Ji Weixi, born in 1961, is currently a professor at College of mechanical engineering, Jiangnan University, Wuxi, China. His research interests include digital manufacturing and intelligent manufacturing.

ji_weixi@126.com

Corresponding author: Wei Peng E-mail: weipeng@jiangnan.edu.cn

ORIGINAL ARTICLE

Hexahedral mesh quality improvement with geometric constraints

Wei Peng^{1,2} • Xinguang Wu³ • Yidong Bao⁴ • Chaoyang Zhang^{1,2} • Weixi Ji^{1,2}

Received; revised

© Chinese Mechanical Engineering

Abstract: Hexahedral mesh is of great value in the analysis of mechanical structure, and the mesh quality has an important impact on the efficiency and accuracy of the analysis. This paper presents a quality improvement method for hexahedral meshes, which consists of node classification, geometric constraints based single hexahedron regularization and local hexahedral mesh stitching. The nodes are divided into different types and the corresponding geometric constraints are established in single hexahedron regularization to keep the geometric shapes of original mesh. In contrast to the global optimization strategies, we perform the hexahedral mesh stitching operation within a few local regions surrounding elements with undesired quality, which can effectively improve the quality of the mesh with less consuming time. A number of mesh quality improvements for hexahedral meshes generated by a variety of methods are introduced to demonstrate the effectiveness of our method.

Keywords: Hexahedral mesh • quality improvement • geometric constraints • feature preserve

1 Introduction

Finite element analysis has been widely used in the field of engineering applications, such as manufacturing process optimization, structural strength analysis and so on. Mesh generation is a pre-processing step for finite element

analysis, and has become one of its main performance bottlenecks. In finite element analysis, the quality of element mesh directly affects the efficiency and accuracy of the numerical analysis results. Therefore, the optimization of element quality is an issue that must be thoroughly studied in the field of finite element analysis. After nearly thirty years of development, mesh generation algorithm has been greatly developed, which can generate surface meshes and tetrahedral meshes with high quality. However, there are still many challenges in the generation of hexahedral meshes for complex solids, especially the quality of generated hexahedral meshes is still far from the requirements of finite element simulation [1]. Therefore, the emphasis of this paper is to improve the mesh quality of hexahedral meshes.

In recent years, a variety of methods on hexahedral mesh smoothing have been developed. Some of the existing methods include: (1) improving quality by changing the mesh connectivity, (2) mesh smoothing by geometric operations, (3) mesh optimization by various energy functions. Grid-based method is a fully automated hexahedral mesh generation method, but the quality of surface elements is poor. To solve this problem, Sun [2] proposed a grid-based quality improvement algorithm, which combines curvature-based Laplacian node smoothing and topological optimization. A fundamental hexahedral mesh is introduced in paper [3] to convert boundary sheets of existing hexahedral meshes to fundamental sheets by flipping operations. Re-meshing [4] is also a method to improve the quality of meshes, which improves the quality of local meshes through boundary feature recognition, node smoothing and topology based mesh repair operations. Wang et.al. [5] presented a novel topological optimization method for hexahedral mesh by sheet adjustment guided by frame field. The

✉ Wei Peng
weipeng@jiangnan.edu.cn

¹ College of mechanical engineering, Jiangnan University, Wuxi, China, 214122

² Jiangsu Key laboratory of Advanced Food Manufacturing Equipment and Technology, Wuxi, China, 214122

³ Xi'an North Hui'an Chemical Industry Company, Xi'an, China, 710000

⁴ College of Mechanical and Electrical Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, China, 210016

implementation of topology based hexahedral mesh quality improvement is very complex and difficult because of its specific connection. The most common used technology in geometry-based methods is Laplace smoothing [6], in which nodes are iteratively moved to the geometric center of their neighborhood. This method is easy to implement and has high efficiency. However, for some complex models, especially when the surface has local concave features, it is difficult to guarantee the mesh quality and easy to generate invalid elements. In order to ensure the validity of the smoothed elements, Freitag [7] proposes a smart Laplace method, which only displaces nodes that can improve mesh quality. Leng et.al. [8] presented a geometric flow based mesh smoothing algorithm for segmented quadrilateral/hexahedral meshes, in which different types of nodes are optimized via different geometric flow driven strategies. The geometric element transformation method (GETMe) proposed by Vartziotis et.al. [9, 10] has been proved to be effective for hexahedral mesh smoothing, which used a simple element regularizing transformation to achieve mesh optimization. While geometry-based mesh quality improvement methods are simple and easy to complement, they do not consider the integrity of the mesh, and ignore the shape features of the surface mesh, which weakens the effect of mesh quality improvement. Optimization based methods perform the quality improvement by minimizing an energy function defined by different quality evaluation criteria. Knupp [11] uses the condition number of Jacobian matrix as the quality metric to create global smoothed cost function, and obtains the global optimization mesh by conjugate gradient and line-search method. With the same objective function, particle swarm approach [12] is used to optimization the global mesh quality. Since then, the local Gauss-Seidel iteration [13] and some nonlinear methods [14] are introduced to untangle and smooth the hexahedral meshes. Livesu et. al [15] presented an untangling method by minimizing an objective function defined in terms of the edge cone descriptor that indirectly measures the distortion of the hexahedral element. With this descriptor, an optimization strategy [16] by minimizing an angle-based energy function is performed on local regions with poor quality hexahedral elements. Vartziotis et.al. [17] used a damped oscillation system to perform discrete geometric triangle transformation to relocate nodes. Peng et.al [18] proposed a local-to-global mesh quality improvement method by a combination of a geometric transformation based local regularization and a volumetric Laplacian based global optimization. While the optimization-based methods and local-to-global smoothing algorithm are effective for

most hexahedral meshes, the solution of linear equations in the global optimization process is very time-consuming, especially for the meshes with a large number of nodes, even some calculation errors may occur.

Previous local-to-global mesh quality improvement method takes into account the geometric characteristics of the mesh when optimizing the mesh as a whole, and cannot perform the mesh optimization locally, resulting in the need to solve a very large energy matrix. This paper proposes new developments related to the strategy of hexahedral mesh optimization, classification of surface nodes and quality improvement of single element. The improved algorithm consists of the following steps: (1) Node classification: classified all nodes into four categories (internal nodes, surface smooth nodes, surface crease nodes and surface corner nodes) by its normal voting tensor; (2) Element clustering: hexahedral elements with quality lower than the user-specified threshold (i.e., elements are connected with each other) are clustered into several local regions; (3) Single element regularization: Convert each hexahedral element to the optimal shape without changing the geometric features by using a geometric constraint-based element regularization algorithm; (4) Global optimization for local regions: to accelerate the computation, we perform the global optimization only in those local regions with element whose quality is lower than the user-specified threshold. The remainder of this paper is organized as follows. In Section 2, we introduce the details of the hexahedral mesh quality improvement algorithm. Several mesh-smoothing examples are given in Section 3, followed by our conclusions in Section 4.

2 Hexahedral mesh quality improvement

The goal of hexahedral mesh quality improvement is to produce a more regular mesh in the premise of ensuring the shape of the original geometry. In order to preserve the features of the original hexahedral mesh, each node can be classified into either an internal node, a surface smooth node, a surface crease node or a surface corner node. Different from the previous local-to-global mesh optimization algorithm, our method adopts different optimization strategies for different regions. For most regions with relatively high-quality hexahedral elements, we use the local hexahedral regularization method based on geometric constraints to keep the geometric features of the original model and avoid the high time-consuming caused by the global optimization operation. For local regions with low-quality hexahedral elements, the global operation is added to ensure the validity of the improved elements. For

most regions with relatively high-quality hexahedral elements, only the local hexahedral regularization method is needed to accelerate the calculation speed of mesh quality improvement. The process of our algorithm is listed in **Algorithm 1**.

Algorithm 1. Hexahedral mesh quality improvement

Input: A hexahedral mesh H_0 with vertices V_0 , elements E_0

Step1: node classification

```

for every vertex  $v_i$  in mesh  $H_0$  do
  if vertex  $v_i$  is a surface node
    Construct the local tensor matrix  $T$  (see Section 2.1)
    Set the type of vertex  $v_i$  (surface smooth, surface crease
    or surface corner)
  else
    Set the type of vertex  $v_i$  (internal)
  end for

```

Step2: element clustering

```

for every element  $E$  in mesh  $H_0$  do
  if the quality of element  $E$  is lower than the user-specified
  threshold  $q_e$ 
    Aggregate element  $E$  and its adjacent elements into a
    local region until the quality of all the peripheral
    elements in this region is greater than the given threshold
     $q_e$ .
  end for (Output several local regions  $R_i$ )

```

Step3: mesh quality improvement

$t = 0$, **repeat**

```

for every element  $E$  in mesh  $H_t$  do (local regularization)
  Align the equilateral element  $E'$  to its corresponding
  element  $E$  by minimizing the distance between element  $E$ 
  and equilateral element  $E'$ .
  Adjust the equilateral element  $E'$  to regularized element  $E''$ 
  with geometric constraints.
end for

```

end for

for every local region R_i **do** (global optimization)

Compute the displacements for all vertices of region R_i by minimizing a quadric energy function, which is established by the global elastic energy and node displacements.

end for

for every vertex v_i outside these local regions **do** (weighted average)

Calculate the optimized vertex v_{oi} of vertex v_i by weighted averaging the vertices of its adjacent regularized elements.

end for

Modify H_t with calculated displacements and updated positions v_{oi} to give H_{t+1} .

$t = t + 1$

until $\max(\text{Dis}(v_i)) < \varepsilon$ or $t > t_{\max}$ ($\text{Dis}(v_i)$ is the displacement of vertex v_i)

Output: The optimized hexahedral mesh H_t

2.1 Node classification

The premise of mesh quality optimization is that the shape of the original model cannot be changed, so the displacement of the nodes needs to be constrained. To this end, we classify the nodes into four categories: (1) Internal nodes inside the hexahedral mesh model, whose displacement is not limited. (2) Surface smoothing nodes are nodes with lower curvature on the surface of hexahedral mesh, and can only move on its tangent plane. (3) Surface crease nodes are those with high curvature in only one direction on the surface of hexahedral mesh, and they are allowed to move only along the crease direction. (4) Surface corner nodes are the intersections of at least three sharp edges that are confined to the initial position. In this paper, the types of nodes are identified by the normal tensor voting theory [19, 20]. Let v_i represents a surface node of the hexahedral mesh and $N_f(v_i)$ denotes its adjacent surface elements, the normal tensor T is defined by weighted unit normal vectors of its neighbor surface elements:

$$T_{\{v_i\}} = \sum_{l \in N_f(v_i)} \omega_l \mathbf{n}_l \mathbf{n}_l^T \quad (1)$$

here $\mathbf{n}_l$ is the unit normal vector of its neighbor surface element $t_{i,l}$ in $N_f(v_i)$, and ω_l is a weight defined by:

$$\omega_l = \frac{A(t_{i,l})}{A_{\max}} \exp\left(-\frac{g_l}{(\sigma/3)}\right) \quad (2)$$

where $A(t_{i,l})$: area of element $t_{i,l}$

$A_{\max}$: maximum element area among $N_f(v_i)$

g_l : distance from v_i to the barycenter of element $t_{i,l}$

σ : edge length of a cube that encircles the adjacent elements of each vertex

The symmetric positive semidefinite matrix T_{v_i} can be decomposed into the following form:

$$T_{\{v_i\}} = \lambda_1 \mathbf{d}_1 \mathbf{d}_1^T + \lambda_2 \mathbf{d}_2 \mathbf{d}_2^T + \lambda_3 \mathbf{d}_3 \mathbf{d}_3^T \quad (3)$$

where λ_1, λ_2 and λ_3 are eigenvalues of T_{v_i} and $\mathbf{d}_1, \mathbf{d}_2$ and $\mathbf{d}_3$ are the corresponding unit eigenvectors of λ_1, λ_2 and λ_3 respectively. We can classify surface vertices into smoothing, crease and corner nodes by relationships of the eigenvalues, described as: (1) Smooth node: λ_1 is dominant, λ_2 and λ_3 are close to 0. (2) Crease node: λ_1 and λ_2 is dominant, λ_3 is close to 0. (3) Corner node: λ_1, λ_2 and λ_3 are approximately equal. More mathematically, suppose $S_s = \lambda_1 - \lambda_2$, $S_e = \lambda_2 - \lambda_3$ and $S_c = \lambda_3$, the maximum of these values is used to classify vertices:

$$\max(Ss, \varepsilon Se, \varepsilon \eta Sc) = \begin{cases} Ss : v_i \text{ is a smooth node} \\ \varepsilon Se : v_i \text{ is a crease node lies on a} \\ \quad \text{crease curve with direction } d_3 \\ \varepsilon \eta Sc : v_i \text{ is a corner node} \end{cases}$$

Here, the parameters ε and η are used to balance the influence of noise on feature detection, which are both set to 4.0 in our experimental tests for good classification results.

2.2 Geometric constraints based element regularization

In the previously proposed local-to-global mesh optimization algorithm, each single element is converted to its regular form by using single geometric transformation. However, the constraints of geometric features are not taken into account in the local regularization operation, which results in the inappropriateness of this method in some regions where global optimization is not required. We propose a hexahedral mesh regularization algorithm based on geometric constraints. The first step is to best align an equilateral hexahedral element of equal volume to a given element. Then, according to the type of nodes, corresponding geometric constraint equations are established, including point constraints, plane constraints and vector constraints. Finally, element regularization is expressed as a quadratic optimization problem, in which the minimum energy consists of two parts: the edge direction constraints of the equilateral element and geometric constraints established according to the type of nodes, and a linear solution can be used to obtain a regularized hexahedral mesh.

Given a hexahedral element $E = (v_1, \dots, v_8)$, $v_i = (x_i, y_i, z_i) \in \mathbb{R}^3$ is the nodes of the given element, an equal volume of equilateral hexahedral element E' can be constructed. Our goal is to find a transformation that best superposes the equilateral hexahedral elements E' with E . The iterative closest point (ICP) algorithm [21] is adopted to perform the alignment, in which the source equilateral hexahedral elements E' is transformed to target hexahedral element E by minimizing the Euclidean distance between the source element and their corresponding target element. The mean square objective function to be minimized is:

$$f(\mathbf{R}, \mathbf{T}) = \frac{1}{8} \sum_{i=1}^8 |v_i - (\mathbf{R}v'_i + \mathbf{T})|^2 \quad (4)$$

where $\mathbf{R}$ is the rotation matrix and $\mathbf{T}$ is the translation matrix to transform the equilateral hexahedral elements E' to E (See Fig.1(b)). In order to obtain the minimum value of equation [4], scholars have proposed a variety of non-iterative optimization methods, such as singular value

decomposition method, quaternion method, etc. Here we use singular value decomposition method to get the rigid transformation matrix $\mathbf{R}$ and $\mathbf{T}$.

In order to keep the shape of the original hexahedral mesh and improve the quality of hexahedral element as much as possible, the equilateral hexahedral elements E' is adjusted to E'' by minimizing an energy function so that the position of node satisfies the geometric constraints. The first term of the energy function is the edge direction constraints of the equilateral element, ensuring the high quality of the regularized element E'' . For each edge $e''_{kj} = v''_k - v''_j$ of the regularized element E'' , it is parallel to the edge $e'_{kj} = v'_k - v'_j$ of the equilateral element E' , where $v''_i = (x''_i, y''_i, z''_i) \in \mathbb{R}^3$ is the nodes of the regularized element E'' . It can be formulized as:

$$(v''_k - v''_j) \times e'_{kj} = 0 \quad (5)$$

where $e'_{kj} = (e_x, e_y, e_z)$ can be easily calculated by the equilateral element E' . Eq. (5) can be written as the following equations:

$$\begin{cases} (y''_k - y''_j) e_z - (y''_k - y''_j) e_z = 0 \\ (x''_k - x''_j) e_z - (z''_k - z''_j) e_x = 0 \\ (x''_k - x''_j) e_y - (y''_k - y''_j) e_x = 0 \end{cases} \quad (6)$$

Hence, the simultaneous equations of all edges can be obtained, which is equivalent to the matrix form:

$$\mathbf{K}\mathbf{X}_E = 0 \quad (7)$$

where $\mathbf{K}$ is a 36×24 matrix; $\mathbf{X}_E = \{x''_1, y''_1, z''_1, \dots, x''_8, y''_8, z''_8\}^T$ is a 24×1 column vector of the node positions of the element E'' .

In addition to preserving the directions of equilateral element, we should add geometry constraints to keep the geometric shapes and features of original hexahedral mesh. For a vertex v''_i , a linear vertex constraint can be developed to penalize the displacement of vertices from their original positions, can be illustrated as below:

1) The geometric constraint for an internal node is $v''_i - v_i = 0$;

2) The surface smooth nodes are constrained to lie on a plane, formulized as: $ax''_i + by''_i + cz''_i + d = 0$;

3) The surface crease nodes are needed lying on a straight line, the geometric constraint is:

$$\begin{cases} a_1x''_i + b_1y''_i + c_1z''_i + d_1 = 0 \\ a_2x''_i + b_2y''_i + c_2z''_i + d_2 = 0 \end{cases}$$

4) The surface corner nodes are restricted to their original location: $v''_i - v_i = 0$;

Thus, we try to minimize the following energy function:

$$\begin{aligned} \min_{\mathbf{X}_E} \theta_K \|\mathbf{K}\mathbf{X}_E\|^2 + \theta_{in} \sum_{i=1}^{n_{in}} |v''_i - v_i|^2 + \theta_{ss} \sum_{i=1}^{n_{ss}} |ax''_i + by''_i + cz''_i + d|^2 \\ + \theta_{sc} \left(\sum_{i=1}^{n_{so}} |a_1x''_i + b_1y''_i + c_1z''_i + d_1|^2 + |a_2x''_i + b_2y''_i + c_2z''_i + d_2|^2 \right) \end{aligned}$$

$$+ \theta_{So} \sum_{i=1}^{n_{So}} |v_i'' - v_i|^2 \quad (8)$$

where θ_K is the weight of the edge direction constraints; θ_{In} , θ_{Ss} , θ_{So} and θ_{Sc} are contributions of internal nodes, surface smooth nodes, surface corner nodes and surface crease nodes respectively, while n_{In} , n_{Ss} , n_{So} and n_{Sc} are number of corresponding nodes. After many tests and adjustments, θ_K , θ_{In} , θ_{Ss} , θ_{So} and θ_{Sc} are set to be 100, 10, 200, 200, and 200 respectively in this paper. We can obtain the set of coordinates of by minimizing the above function, which results in a sparse linear system as the following:

$$AX_E = \begin{pmatrix} \theta_K \mathbf{K} \\ \theta_{In} \mathbf{I} \\ \theta_{Ss} \mathbf{S} \\ \theta_{Sc} \mathbf{C} \\ \theta_{So} \mathbf{O} \end{pmatrix} X_E = \begin{pmatrix} 0 \\ \mathbf{B}^I \\ \mathbf{B}^S \\ \mathbf{B}^C \\ \mathbf{B}^O \end{pmatrix} = \mathbf{B} \quad (9)$$

where A is a $(36+3n_{In}+n_{Ss}+3n_{So}+2n_{Sc}) \times 24$ matrix; B is a 24×1 column vector. The sparse linear system in Eq. (9) can be solved in a least square sense as:

$$X_E = (A^T A)^{-1} A^T B \quad (10)$$

Fig.1 demonstrates the regularization process of single hexahedron with a given initial element E show in Fig.1(a). The nodes are first classified to surface smooth nodes (v_0, v_1, v_2, v_3), surface crease nodes (v_4) and internal nodes (v_5, v_6, v_7). An equal volume of equilateral hexahedron E' is drawn with red lines in Fig.1(b) by minimizing the Euclidean distance between equilateral hexahedron E' and initial element E , in which $e_{4,0} = v'_4 - v'_0$, $e_{1,0} = v'_1 - v'_0$ and $e_{3,0} = v'_3 - v'_0$ are edge directions to be maintained. The positions of the final regularization hexahedron can be founded by solving Eq. (9) with $\theta_K = 100$, $\theta_{In} = 10$, $\theta_{Ss} = 200$, $\theta_{Sc} = 200$ and $\theta_{So} = 200$, plotted by black lines in Fig.1(c). After regularization operation, the scaled Jacobian value is updated from 0.39 to 0.92. We can see that our method obtains a very high quality regular element, while the geometric features of initial mesh remain unchanged.

Fig.1. Hexahedron regularization. (a) Initial hexahedron; (b) Equilateral hexahedron; (c) Final regularized hexahedron.

2.3 Mesh quality improvement

The global optimization process in previous local-to-global algorithm is to optimize all hexahedral meshes as a whole, while the global optimization process in this paper only performs in local regions. The recognition of local regions consists of the following steps:

1) When a seed hexahedral element with its scaled Jacobian lower than the user-specified threshold q_e (we set q_e to 1/4 of the maximum scaled Jacobian of all hexahedral elements) is encountered, a new local region is created, containing this hexahedral element, associated with a new label. The scaled Jacobian [22], as the main mesh quality standard, is used to measure whether the element is qualified. For a node v_i and its three connected nodes $\{v_{i,1}, v_{i,2}, v_{i,3}\}$ belonging to a hexahedron E , a 3×3 matrix can be formed by three vectors $U(v_i) = [v_{i,1} - v_i, v_{i,2} - v_i, v_{i,3} - v_i]$. By normalizing the column vectors of matrix $U(v_i)$, the scaled Jacobian value of node v_i can be defined as $J(v_i) = \det(U(v_i))$. The minimal scaled Jacobian among all nodes in hexahedral element is the scaled Jacobian of element E , expressed as $J(E)$.

2) Then a recursive process extends this region: for each hexahedral element with quality lower than the user-specified threshold in the local region, the adjacent elements are integrated into the local region, associated with new labels. The termination condition of this recursive process is that there is no element lower than the user-specified threshold at the periphery of the local region.

3) The recognition is repeated for every other hexahedral element with quality lower than the user-specified threshold and still unlabeled.

2.3.1 Global optimization for local regions

To prevent the generation of invalid elements, a linear elastic energy based optimization is used to relocate the nodes belonging to the local regions. The process of local hexahedral regularization can be seen as a process in which the initial hexahedral element is elastically deformed into a regularized hexahedral mesh, and the internal force of the node is bound to occur during the deformation. Thus, the global optimization can be recast into that of solving the deformation of hexahedral meshes under the internal force. For an initial hexahedron $E = (v_1, \dots, v_8)$, $v_i = (x_i, y_i, z_i) \in \mathbb{R}^3$ and its corresponding regularized hexahedron $E'' = (v''_1, \dots, v''_8)$, $v''_i = (x''_i, y''_i, z''_i) \in \mathbb{R}^3$, the displacement of hexahedron is $q^e = \{\Delta x_0, \Delta y_0, \Delta z_0, \dots, \Delta x_7, \Delta y_7, \Delta z_7\}$, can be calculated as follows: $\Delta x_i = x''_i - x_i$, $\Delta y_i = y''_i - y_i$, $\Delta z_i = z''_i - z_i$, ($i=0, \dots, 7$). Assume that regular hexahedron E'' is obtained by linear elastic deformation of initial hexahedron E , the eight-node

hexahedron isoparametric element is used to calculate the internal forces:

$$\mathbf{F}_{in}^e = \mathbf{B}^T \mathbf{D} \mathbf{B} \mathbf{q}^{eT} = \mathbf{k}^e \mathbf{q}^{eT} \quad (11)$$

where matrix $\mathbf{k}^e$ is the stiffness matrix of hexahedral element, which consists of the strain matrix $\mathbf{B}$ and the linear elastic matrix $\mathbf{D}$ of the hexahedral element [23]. After calculating the internal forces of each element by formula [11], suppose that $L(i) = \{l_1, \dots, l_{m(i)}\}$ denotes $m(i)$ hexahedrons adjacent to the node v_i . The internal force of node v_i is obtained by average of its adjacent elements:

$$\mathbf{F}_{in}^i = \frac{1}{m(i)} \sum_{j \in L(i)} \mathbf{F}_{in}^{eL(i)} \quad (12)$$

The energy equation under node load is as follows:

$$\min_{\Delta \mathbf{X}} E(\Delta \mathbf{X}) = \frac{1}{2} \Delta \mathbf{X}^T \mathbf{K} \Delta \mathbf{X} + \mathbf{F}_{in} \Delta \mathbf{X} \quad (13)$$

where $\mathbf{K}$ is the global stiffness matrix; $\Delta \mathbf{X} = \{\Delta x_0, \Delta y_0, \Delta z_0, \dots, \Delta x_{n-1}, \Delta y_{n-1}, \Delta z_{n-1}\}$ is the displacement vectors of all nodes in a local region. The energy equation [13] is quadratic and the displacements can be calculated by solving a sparse linear system:

$$\mathbf{K} \Delta \mathbf{X} = \mathbf{F}_{in} \quad (14)$$

2.3.2 Update node location

The new mesh node positions are determined by different strategies. For nodes in local regions, linear elastic energy based optimization is used to compute the displacement of the nodes simultaneously, and the optimized node location is $v_{oi} = v_i + \Delta \mathbf{X}_i$. Otherwise, the new mesh node position will be directly derived from the following weighted averages as:

$$v_{oi} = \frac{\sum_{i \in L(i)} \mu_i v''_i}{\sum_{i \in L(i)} \mu_i} \quad (15)$$

where $L(i)$ represents all elements surrounded to node v_i ; v''_i is the position of local regularized hexahedral element E'' ; and μ_i is the weight of node v''_i , defined as $\mu_i = 1.0 - J(E_i)$, $J(E_i)$ is the scaled Jacobian value of given element E_i .

3 EXPERIMENTAL RESULTS

We have applied our hexahedral mesh improvement algorithm to several hexahedral meshes generated by different methods, running on a machine with Intel Core i7 2.6 GHz CPU and 8 GB memory. To evaluate the mesh quality, the minimal and mean mesh quality value are defined as:

$$J_{\min} = \min_{i \in \{1, L, n_H\}} J(E_i),$$

$$J_{\text{mean}} = \frac{1}{n_H} \sum_{i \in \{1, L, n_H\}} J(E_i)$$

where $J(E)$ is the scaled Jacobian value of element E_i , and n_H is the element number of hexahedral meshes. In the following test examples, the mesh quality is visually illustrated by color map according to the scaled Jacobian value – the color of a hexahedron is blue if its scaled Jacobian value is 1.0 and red if its scaled Jacobian value is 0.0. Additionally, the mesh quality is quantitatively measured by histograms (in percentage) with respect to scaled Jacobian values. We also compare the method proposed in this paper with smart Laplace [7], GETMe [9] and local-to-global smoothing [18] methods to verify its effectiveness.

Fig.2 Mesh quality improvement of a torque arm model

Fig.2 is a hexahedral mesh quality improvement of a torque arm model. The given original mesh is generated by sweep method [24, 25], displayed in Fig.2(a). Fig.2(b), (c), (d) and (e) are improved results obtained by smart Laplace, GETMe, local-to-global and our method. Distributions of scaled Jacobian (in percentage) are shown in Fig.3 to compare the quality of different hexahedral meshes in Fig.2. The minimal mesh quality value of the smoothed mesh achieved by smart Laplace,

GETMe, local-to-global and our method is elevated to 0.24, 0.24, 0.5 and 0.51 respectively. We can see that smart Laplace and GETMe are failed to improve the minimal mesh quality because they did not consider the influence of the surface node on mesh quality. The proposed method achieves the same mesh quality as the local-to-global method, which meets the requirements of finite element analysis.

Fig.3 Comparison of the mesh quality of a torque arm model

Fig.4 Mesh quality improvement of an eight-shape model

Fig.5 Comparison of the mesh quality of an eight-shape model

Automation of hexahedral mesh generation has always been a major problem in the field of mesh generation. At present, the two most automated methods are tetrahedral splitting method and grid-based method, which often lead to a number of horrible quality hexahedral elements. Fig. 4 is a hexahedral mesh optimization of an eight-shape model obtained by tetrahedral splitting method. The corresponding quality distributions of Fig.4 (a), (b), (c), (d) and (e) is displayed in Fig.5, in which the minimal mesh quality is 0.02, 0.04, 0.07, 0.17 and 0.15 respectively.

Fig.6 Mesh quality improvement of a rod model

Fig.7 Comparison of the mesh quality of a rod model

The grid-based method [26, 27] is one of the most widely used methods, which is suitable for any shape of solid model, and can obtain high quality internal mesh and low quality surface mesh. The following two examples of mesh smoothing are performed for hexahedral meshes generated by grid-based method. Fig.6(a) shows the original unstructured hexahedral mesh of a rod model. Through the smart Laplace, GETMe, local-to-global and our method, the improved hexahedral meshes show in Fig.6(b), (c), (d) and (e) are obtained. Fig.7 is a comparison of mesh quality of different meshes, in which the minimal mesh quality is 0.08, 0.09, 0.09, 0.14 and 0.19 respectively and our method has produced the largest proportion of regular elements.

Fig.8 is a mesh quality improvement example of a base model with 129148 hexahedrons and 146546 nodes. Due to the number of nodes in this model is too large, the sparse matrix established by local-to-global method is too huge, resulting in errors in mesh quality improvement. However, the method in this paper only performs global optimization in local region, which saves the storage space of sparse matrices and avoids calculation errors. The maximum and minimal scaled Jacobian values of original hexahedral mesh show in Fig.(a) are 0.01 and 1.0, and a reasonable threshold for local region recognition can be set to 0.25 (i.e. 1/4 to 1/3 of the maximum and minimal Jacobian interval). With this specified threshold, 187 local regions are clustered in Fig.8 (b). The improved hexahedral meshes with a minimal mesh quality of 0.02, 0.02 and 0.12 are plotted in Fig.8 (c), (d) and (e) respectively, and the mesh quality comparison of different improved results is shown in Fig.9.

Fig.8 Mesh quality improvement of a base model

Fig.9 Comparison of the mesh quality of a base model

Table 1 gives the mesh quality statistics of improved hexahedral mesh on Torque arm, Eight-shape, Rod and Base models and the computational costs are given in Table 2. As can be seen from Table 1 and Table 2 that compared with smart Laplace and GETMe method, our method can achieve much better improved results in almost the same amount of time-consuming. Compared with local-to-global method, our method is faster and achieves almost the same effect of mesh quality improvement.

Table 1 Comparison of mesh quality

Methods	Torque arm		Eight-shape		Rod		Base	
	$J_{\min}$	J_{mean}	$J_{\min}$	J_{mean}	$J_{\min}$	J_{mean}	$J_{\min}$	J_{mean}
Initial mesh	0.24	0.97	0.02	0.39	0.08	0.83	0.01	0.81
Smart Laplace	0.24	0.97	0.03	0.46	0.09	0.83	0.02	0.81
GETMe	0.24	0.98	0.07	0.51	0.09	0.83	0.02	0.81
Local-to-global	0.50	0.99	0.17	0.53	0.14	0.85	-	-
Our method	0.51	0.99	0.16	0.53	0.19	0.85	0.12	0.83

Table 2 Comparisons of computational time

Models	Nodes	Elements	Runtime(s)			
			Smart Laplace	GETMe	Local-to-Global	Our
Torque arm	36215	31356	7.80	15.03	71.15	22.32
Eight-shape	16501	13024	3.83	5.64	31.83	9.57
Rod	29534	25909	5.92	12.23	60.28	18.18
Base	146546	129148	53.61	112.94	-	113.78

4 CONCLUSION

- (1) Reference to the previous local-to-global hexahedral mesh smoothing algorithm, we introduce a new geometric constraints based hexahedral mesh quality improvement method.
- (2) The tensor analysis procedure is used to determine the type of nodes, and different geometric constraints are added to obtain a regular form for each hexahedral element by minimizing a quadratic functional in a least squares sense.
- (3) A global hexahedral mesh optimization is performed locally for a number of regions with element whose quality is lower than the user-specified threshold. The mesh nodes outside these local regions are simultaneously relocated by weighting the contribution of new temporary node in each attached element.
- (4) As compared to other two geometry-based and

local-to-global hexahedral mesh-smoothing methods, the optimization results obtained by this method are similar to those obtained by the local-to-global method and are superior in computational efficiency.

- (5) The parameters ε and η in node classification process are sensitive to the noise of meshes, which may lead to errors in judgment of node type. For some hexahedral meshes with complex topology and large size change, the improvement of mesh quality by this method is not obvious enough.

5 Declaration

Funding

This work was funded by the Project of the National Natural Science Foundation of China (No.11402264) and Open Fund of Jiangsu Key Laboratory of Precision and Micro Manufacturing Technology. These supports are gratefully acknowledged.

Availability of data and materials

The datasets supporting the conclusions of this article are included within the article.

Authors' contributions

The author's contributions are as follows: Wei Peng was in charge of the whole trial and wrote the manuscript; Xinguang Wu and Yidong Bao assisted with model testing and data analyses; Chaoyang Zhang and Weixi Ji assisted with writing and revising the paper.

Competing interests

The authors declare no competing financial interests.

Consent for publication

Not applicable

Ethics approval and consent to participate

Not applicable

References

- [1] T. Blacker. Meeting the challenge for automated conformal hexahedral meshing, in: 9th International Meshing Roundtable, 2000: 11–20.
- [2] Sun L, Zhao GQ, Ma XW. Quality improvement methods for hexahedral element meshes adaptively generated using grid-based algorithm. International Journal for Numerical Methods in Engineering, 2012, 89(6):726-761.
- [3] Ledoux F, Shepherd J. Topological modifications of hexahedral meshes via sheet operations: a theoretical study. Engineering with Computers, 2010, 26(4):433-447.

- [4] Nielsen C V, Fernandes J L M, Martins P A F. All-hexahedral meshing and remeshing for multi-object manufacturing applications. *Computer-Aided Design*, 2013, 45(5):911-922.
- [5] Wang R, Gao S, Zheng Z, Chen J. Hex mesh topological improvement based on frame field and sheet adjustment. *Computer Aided Design*, 2018, 103:103-117.
- [6] Field D A. Laplacian smoothing and Delaunay triangulations. *Communications in Applied Numerical Methods* 1998, 4(6): 709-712.
- [7] Freitag L A. On combining Laplacian and optimization-based mesh smoothing techniques. *Trends in Unstructured Mesh Generation*. 1997: 37-43.
- [8] Leng J, Xu G, Zhang Y, Qian J. Quality improvement of segmented hexahedral meshes using geometric flows. In: *Image-based geometric modeling and mesh generation*. Springer; 2013: 195-221.
- [9] Vartziotis D, Wipper J, Papadrakakis M. Improving mesh quality and finite element solution accuracy by GETMe smoothing in solving the Poisson equation. *Finite Elements in Analysis and Design*, 2013, 66:36-52.
- [10] Vartziotis D, Wipper J. A dual element based geometric element transformation method for all-hexahedral mesh smoothing. *Computer Method in Applied Mechanics and Engineering*, 2011; 200(9-12): 1186-1203.
- [11] Knupp P M. Achieving finite element mesh quality via optimization of the Jacobian matrix norm and associated quantities. Part II—a framework for volume mesh optimization and the condition number of the Jacobian matrix. *International Journal for Numerical Methods in Engineering* 2000, 48(3):1165-1185.
- [12] Yilmaz A E, Kuzuoglu M. A particle swarm optimization approach for hexahedral mesh smoothing. *International Journal for Numerical Methods in Fluids* 2009; 60(1): 55-78.
- [13] Ruiz-Gironés E, Roca X, Sarrate J. Optimizing mesh distortion by hierarchical iteration relocation of the nodes on the cad entities. *Proc Eng* 2014, 82:101-13.
- [14] Ruiz-Gironés E, Roca X, Sarrate J, Montenegro R, Escobar JM. Simultaneous untangling and smoothing of quadrilateral and hexahedral meshes using an object-oriented framework. *Adv Eng Softw* 2015, 80:12-24.
- [15] Livesu M, Sheffer A, Vining N, et al. Practical hex-mesh optimization via edge-cone rectification. *ACM Transactions on Graphics*, 2015, 34(4):141:1-141:11.
- [16] Xu K, Gao X, Chen G. Hexahedral mesh quality improvement via edge-angle optimization. *Computers & Graphics*, 2017, 70: 17-27.
- [17] Vartziotis D, Bohnet D. A geometric mesh smoothing algorithm related to damped oscillations. *Computer Methods in Applied Mechanics and Engineering*, 2017, 326: 102-121.
- [18] Peng W, Lu D, Huang T, et al. Hexahedral mesh smoothing via local element regularization and global mesh optimization. *Computer-Aided Design*, 2015, 59:85-97.
- [19] Page DL, Sun Y, Koschan AF, et al. Normal vector voting: crease detection and curvature estimation on large, noisy meshes. *Graphical Models* 2002; 64: 199-229.
- [20] Kim HS, Han KC, Lee KH. Feature detection of triangular meshes based on tensor voting theory. *Computer Aided Design*, 2009, 41: 47-58.
- [21] Besl P J, McKay N D. A method for registration of 3-D shapes. *IEEE Trans. Pattern Anal Mach Intell*, 1992, 14(2): 239-256.
- [22] Knupp P M. A method for hexahedral mesh shape optimization. *International Journal for Numerical Methods in Engineering*, 2003, 58(2): 319-332.
- [23] Dhatt G, Lefrançois E, Touzot G. *Finite Element Method*, 1st ed., Hoboken: John Wiley & Sons, 2012.
- [24] Peng W, Lu D, Huang T, et al. As-rigid-as-possible mesh deformation and its application in hexahedral mesh generation. *Advances in Engineering Software*, 2013, 65:158-167.
- [25] Lu H C, Quadros W R, Shimada K. Evaluation of user-guided semi-automatic decomposition tool for hexahedral mesh generation. *Journal of Computational Design and Engineering*, 2017:330-338.
- [26] Huang L, Zhao G, Wang Z, et al. Adaptive hexahedral mesh generation and regeneration using an improved grid-based method. *Advances in Engineering Software*, 2016, 102:49-70.
- [27] Maréchal L. Advances in Octree-Based All-Hexahedral Mesh Generation: Handling Sharp Features. *Proceedings of the 18th International Meshing Roundtable*. 2009: 65-84.

Biographical notes

Peng Wei, born in 1984, is currently an associate professor at College of mechanical engineering, Jiangnan University, Wuxi, China. He received his doctor degree from Nanjing University of Aeronautics and Astronautics, China, in 2012. His research interests include digital manufacturing and finite element method. Tel: +86-510-85910251; E-mail: weipeng@jiangnan.edu.cn

Wu Xinguang, born in 1980, is currently an engineer at Xi'an North Hui'an Chemical Industry Company, China. His research interests include numerical simulation. E-mail: wuxinguang555@163.com

Bao Yidong, born in 1976, is currently an associate professor at College of Mechanical and Electrical Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, China. He received his doctor degree from Jilin University, China, in 2005. baoyd@nuaa.edu.cn

Zhang Chaoyang, born in 1987, is currently an assistant professor at College of mechanical engineering, Jiangnan University, Wuxi, China. He received his doctor degree from Xi'an Jiaotong University, China, in 2016. cyzhang@jiangnan.edu.cn

Ji Weixi, born in 1961, is currently a professor at College of mechanical engineering, Jiangnan University, Wuxi, China. His research interests include digital manufacturing and intelligent manufacturing. ji_weixi@126.com

Figures

Figure 1

Hexahedron regularization. (a) Initial hexahedron; (b) Equilateral hexahedron; (c) Final regularized hexahedron.

Figure 2

Mesh quality improvement of a torque arm model

Figure 3

Comparison of the mesh quality of a torque arm model

Figure 4

Mesh quality improvement of an eight-shape model

Figure 5

Comparison of the mesh quality of an eight-shape model

Figure 6

Mesh quality improvement of a rod model

Figure 7

Comparison of the mesh quality of a rod model

Figure 8

Mesh quality improvement of a base model

Figure 9

Comparison of the mesh quality of a base model