Engineering with Computers (2004) 20: 184–192
DOI 10.1007/s00366-004-0286-2

ORIGINAL ARTICLE

Zhijian Chen · Joseph R. Tristano · Wa Kwok

Construction of an objective function for optimization-based smoothing

Received: 28 July 2003 / Accepted: 12 April 2004 / Published online: 19 August 2004

Abstract An objective function for optimization-based smoothing is proposed for both linear and quadratic triangular and quadrilateral elements. Unlike currently published objective functions that are used to perform smoothing or untangling separately, this objective function can be used to untangle and smooth a mesh in a single process. The objective function is designed in such a way that it is easy and straightforward to be extended to higher order elements. The objective function has higher order continuous derivatives that make it suitable for optimization techniques. It has been shown empirically that the proposed function only has one minimum. With the integration of the proposed new objective function into our optimization-based smoothing algorithm, our combined Laplacian/optimization smoothing scheme provides us with satisfactory high quality meshes.

Keywords Optimization-based smoothing · Constrained-Laplacian smoothing · Smoothing objective function · Surface mesh

1 Introduction

Mesh quality is a key factor in finite element method (FEM) analysis. There are numerous ways to achieve a high quality mesh [1], such as controlling the discretization size, controlling the edge valence of mesh nodes, and controlling the distortion of the individual element shapes. Mesh smoothing (relaxation) [2] improves quality by adjusting node locations to reduce the distortion of the element shapes without changing the topology of the mesh. In general, mesh smoothing can be classified into two major groups [3]: local and global. In the local smoothing, nodes are moved one by one, while the global smoothing changes all the nodal locations in a mesh simultaneously.

The most commonly used smoothing technique is Laplacian smoothing [4], which moves a given node to the geometric center of its incident nodes. Various weighted Laplacian smoothing algorithms have been developed to improve the performance of the original smoothing technique. Laplacian smoothing is computationally inexpensive, but it does not guarantee an improvement in mesh quality. In fact, it is possible to create inverted or invalid elements with this technique. A valid mesh is one whose elements have acceptable quality metrics [2]. Constrained-Laplacian smoothing [5] overcomes this problem by placing a node at a new location only when the mesh quality is improved. This method successfully prevents the degradation of mesh quality, but does not always improve the quality of the mesh or place nodes at their best locations.

In recent years, optimization-based smoothing algorithms have been drawing the attention of the mesh generation community. Several optimization-based smoothing algorithms have been developed [2–4]. These algorithms integrate some mesh quality measures into objective functions. In general, optimization techniques yield better meshes when an objective function is properly formulated. Optimization-based smoothing schemes vary based on: (1) the type of mesh being smoothed, (2) the optimization method used, and (3) the distortion metric selected to construct the objective function.

One of the keys to the success of an optimization-based smoothing algorithm is to define an appropriate objective function. An inappropriate objective function might not only waste time in the optimization but also cause smoothing to fail to improve the mesh quality. Most of the efficient optimization algorithms [6] require an objective function to be C^1 continuous.

Various measures for element quality [7] have been used in the objective function, such as distortion metrics, aspect ratio, minimum angle, etc. Recently, the inspiring

Z. Chen (✉) · J. R. Tristano · W. Kwok
Ansys Inc., 275 Technology Drive,
Canonsburg, PA 15317, USA
E-mail: james.chen@ansys.com
E-mail: joe.tristano@ansys.com
E-mail: wa.kwok@ansys.com

work of Knupp [8] derived an objective function from the condition number of an element Jacobian matrix. His work, along with the work of Freitag [3, 9], has lead to mesh quality improvement algorithms for 2D and 3D linear elements. Paoletti [10] stated that an interpolation tensor could be applied to various polyhedral meshes in 2D and 3D. Even though the published works show enormous potential, there are two general limitations in these algorithms:

1. They are only applied to linear elements
2. They require valid initial meshes

It is not unusual that a mesh to be smoothed is not guaranteed to be valid. Most of the existing objective functions used in optimization-based smoothing schemes mentioned cannot guarantee a converged solution for an invalid mesh. For this reason, untangling techniques [9] have been proposed to remove invalid elements from the mesh before executing optimization-based smoothing.

In general, optimization-based smoothing is very slow compared with Laplacian smoothing. Canann et al. [2] combined the use of Laplacian and optimization-based smoothing to speed up the smoothing process, along with the benefit of better mesh quality from optimization smoothing. In his work, the distortion metrics, α for a triangle [11] and β for a quadrilateral [2], are used in the objective function, optimizing the infinity norm. Based on our experience, reasonably good meshes have been achieved with this approach. However, there are some cases when smoothing nodes are near to curved boundaries, the final mesh quality around these elements is not always satisfactory (Fig. 1). For this reason, we recently improved our smoothing algorithm to make it suitable for working with quadratic elements by developing a new objective function to be used for optimization-based smoothing [12]. Current work is limited to isotropic meshes, such as the ones for structural analyses.

In what follows, Sect. 2 will discuss issues related to the creation of objective functions; Sect. 3 will address the proposed objective function; and Sect. 4 will present some examples to illustrate the effectiveness and robustness of the proposed objective function. In the last section, we will conclude our discussion and present future work in the area.

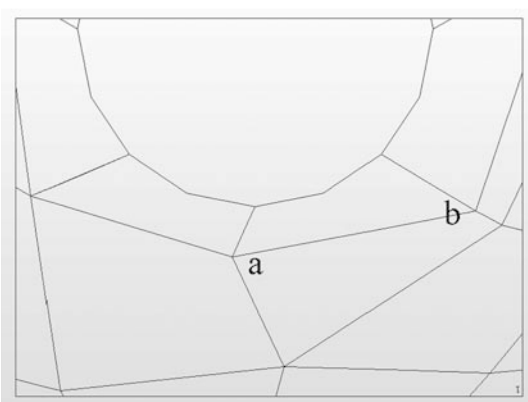


Fig. 1 A simple mesh with poor smoothing. Better locations exist for nodes *a* and *b*

2 Objective function creation

2.1 Basic criteria for objective functions

A distortion metric is a measure of a mesh's quality. Therefore, an objective function for smoothing is usually constructed based on some distortion metrics or a combination thereof. A metric is suitable for use in an objective function if the following criteria are met:

1. Continuity: an effective way to speed up an optimization process is to utilize more sophisticated algorithms. Usually, more efficient algorithms [6] require the use of higher order derivatives. An objective function with higher order continuous derivatives tends to faster convergence of an optimization process.
2. Monotonically decreasing: an objective function with multiple local minima will be difficult for the optimization algorithm to find the optimal solution. If the objective function is not monotonically decreasing, the optimized location may vary based on the initial location of the node to be smoothed. An objective function that is not monotonically decreasing is highly unlikely to succeed in an untangling process.
3. Shape independence: it is favorable for the objective function to be defined and normalized in such a way that the metrics for all element shapes can work together. Therefore, the scale of the metrics for different element shapes should be at a similar level.

In practice, efficiency is also critical. Because of the huge number of objective function evaluations, a computationally expensive objective function will make an optimization-based smoothing slower: a small inefficiency will add up to be very expensive in the end.

Historically, we at Ansys have used two shape metrics: α for triangular elements and β for quadrilateral elements. Let us now discuss their properties.

2.2 α for triangular elements

The triangular metric, α, [11] is defined as:

$$\alpha = \pm \frac{2\sqrt{3}\left\| \overrightarrow{\mathbf{BC}} \times \overrightarrow{\mathbf{AC}} \right\|}{l_{\mathrm{AB}}^2 + l_{\mathrm{BC}}^2 + l_{\mathrm{CA}}^2} \tag{1}$$

where l_{AB}, l_{BC}, and l_{CA} are the edge lengths of the triangle ΔABC, and $\overrightarrow{\mathbf{BC}}$ and $\overrightarrow{\mathbf{AC}}$ are the edge vectors of the triangle, as shown in Fig. 2a. The metric is signed negative when an element is inverted. For a linear

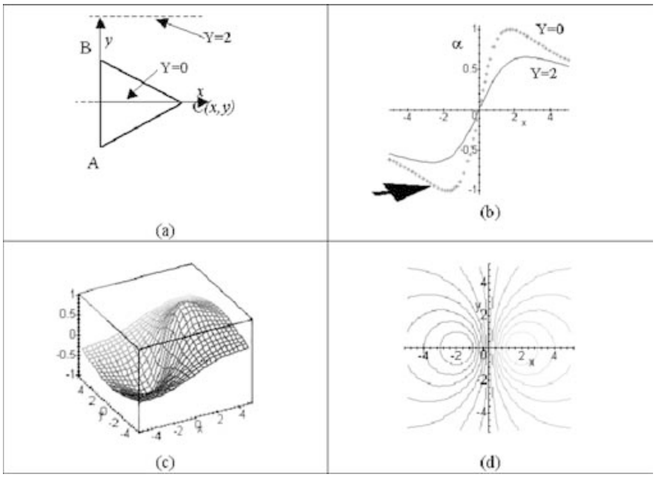


Fig. 2a–d α for a triangle. **a** A triangle with point C as a smoothing node that is moving over the x–y plane. **b** The change of α on $y=0$ and $y=2$. **c** The 3D plots of α. **d** The contour of α on the x–y plane

triangular element, the numerator of Eq. 1 is directly related to the area of the triangle, while the denominator is the sum of the squared edge lengths of the triangle. The shape metric α is bounded by $\alpha \in [-1, 1]$. A value of 1 corresponds to the best triangle, an equilateral triangle, while a value of -1 indicates an inverted equilateral triangle. When all the three points of the triangle are co-linear, the triangle has a zero area, which yields a value of $\alpha = 0$. α has only one maximum and one minimum.

When α is used alone as an objective function for optimization-based smoothing, it will be difficult to untangle inverted triangles. For example, if an initial point is placed at the location as shown by the arrow in Fig. 2b, the shape metric α would tell us to move the point in the negative x-direction to improve mesh quality.

2.3 β for quadrilateral elements

Quality metric β [2], used to measure quadrilaterals, is comprised of a combination of the αs of the triangular elements derived from the quadrilateral. The basic concept is to split a quadrilateral into four different triangles, ΔABD, ΔABC, ΔBCD, and ΔCDA (Fig. 3). Each of these triangles has a quality metric, α_1, α_2, α_3, and α_4, respectively, such that:

$$\beta = \frac{\min(\alpha_1, \alpha_2, \alpha_3, \alpha_4) - n_{\text{neg}}}{\alpha_{90}} \tag{2}$$

where α_{90} is the quality metric for a right-angled triangle having unit base and height lengths, and n_{neg} is the number of negative sub metrics. α_{90} is used as a normalization factor to have β be the ideal value of 1 when the quad is square, while n_{neg} is a heuristic value that is used to help the smoothing algorithm un-invert tangled meshes such that inverted elements would have a large

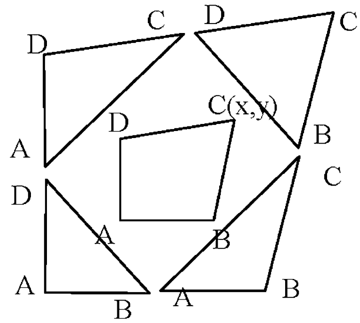


Fig. 3 Sub-divide a quad into four triangles. Node C is the smoothing node

weighting in the objective function. More details of this procedure can be found in Canann et al. [2]. Figure 4b is a 3D diagram of the metric β. It is quite obvious that the function has many steps that are caused by the term n_{neg}. Such step functions are undesirable properties for optimization. β has similar problems to α, Fig. 2, regarding untangling.

If the n_{neg} term is removed from the Eq. 2, then the function is simply:

$$\beta' = \frac{\min(\alpha_1, \alpha_2, \alpha_3, \alpha_4)}{\alpha_{90}} \tag{3}$$

This modified metric β' is illustrated in Fig. 5. This figure clearly shows that the objective function has multiple local minimums. These are obviously not desirable to an optimization process.

2.4 Problems with higher order elements

It is quite often observed that an element is of acceptable quality when it is linear but is of unacceptable quality when it is quadratic with curved edges. This phenomenon is illustrated in Fig. 6, where the dotted lines are geometric boundaries. The elements in Fig. 6a, c are of acceptable quality when the elements are linear. However, with the introduction of mid-side nodes, the quadratic elements in Fig. 6b, d show interior angles close to 0° and 180°, which are invalid elements.

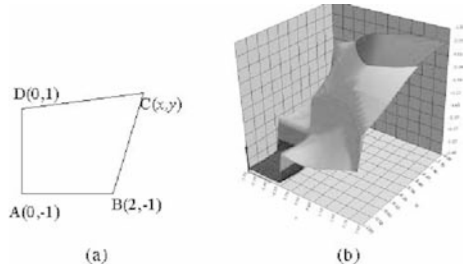


Fig. 4a, b β for a quad. **a** A quadrilateral with point C set to be the free node. **b** The change of β when point C is moving in the x–y plane

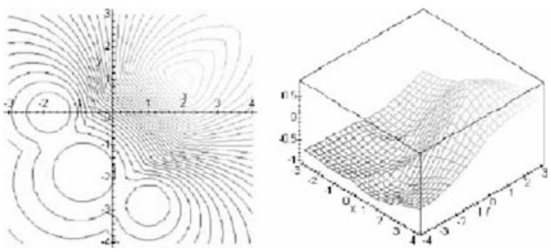


Fig. 5a, b Modified β for a quad. **a** The contour. **b** 3D distribution when point C is moving in the x–y plane

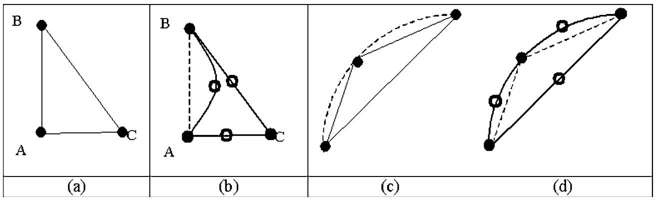


Fig. 6 The difference in element quality for linear and quadratic elements

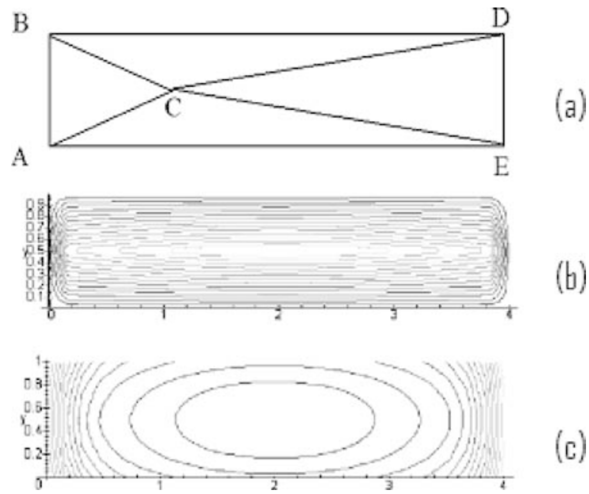


Fig. 7a–c Comparison between the contours of a simple mesh. **a** The mesh: a rectangle is divided into four triangles. **b** The contour of objective function is created in the infinity norm sense. **c** The contour of the objective function formed in the L2 norm sense

A curved element edge usually occurs when the element edge is on a curved boundary. In the above example, the element in Fig. 6b might be improved by moving node C, if node C is an interior node. However, the element in Fig. 6d has no node to move to improve its quality. The quality improvement for this class of problems is beyond the scope of smoothing.

Smoothing for linear simplex elements (three-node triangle for 2D and four-node tetrahedral for 3D) has been studied substantially with fruitful achievements [3, 4, 8, 9]. However, simple straightforward means to extend these studies to quadratic triangular and quadrilateral have not appeared in the literature.

2.5 Infinity norm or L2 norm sense

The use of the infinity norm in the construction of the objective function is a very popular approach. As an example, the authors [2] attempt to optimize:

$$\max \rightarrow \min(\alpha_i) \tag{4}$$

where α_i is one of the distortion metrics related to the smoothing node. The impetus for this method is to target the worst element.

One may also use the L2 norm to construct the objective function, defined as:

$$\max \rightarrow \sum_i \alpha_i^2 \tag{5}$$

where, similar to the infinity norm, α_i is comprised of a distortion metric related to the node to be smoothed. The impetus for the L2 norm is to handle all elements attached to the node.

Figure 7 illustrates the contour of the objective function for the smoothing node C in the simple mesh. It is easily observed that the objective function formed in the L2 norm sense has the better shape for optimization with a well-defined target zone.

Another advantage of using the L2 norm is that it is easier to add new terms to the objective function for special purposes, such as the penalty terms, which will be described below. In this work, the L2 norm is used for constructing the proposed objective function.

2.6 Three triangles or four triangles?

A quadrilateral is sub-divided into four triangles, as shown in Fig. 3. In the definition of β, it is obvious that, if ΔABD is the worst triangle in the set, the value of β is not affected by perturbations of node C. Therefore, ΔABD is a "dead zone" with respect to the smoothing node C.

For this reason, the proposed objective function is defined based only on three triangles, ΔABC, ΔACD, and ΔBCD. The dead zone, ΔABD, is not used in the objective function proposed in this paper.

3 The proposed objective function

3.1 Terms of the objective function

Experience has shown that the previous objective function in Eq. 5 does not enable satisfactory optimization-based smoothing for the general case of mixed-shape quadratic elements. The distortion metrics, α and β, cannot be the lone contributors to the objective function.

By observation, an invalid element is recognized by its inverted interior element angles. Once an element angle is less than 0° or larger than 180°, the element becomes invalid. Therefore, it is quite natural to add a penalty term derived from the interior angles. When considering quadratic elements, the effects of the presence of curved element edges can also be covered by element angles. Poor elements in quadratic meshes are often caused by the bad angles introduced by the curved element edges with mid-side nodes, as shown in Fig. 6.

By introducing an angle penalty term, the proposed objective function is made up of two parts:

$$f = f(\alpha) + f(\theta) \tag{6}$$

where $f(\alpha)$ is based on the shape metric α and $f(\theta)$ is an angle penalty term based on element angles computed at the opposite two nodes. The purpose of the $f(\theta)$ term is to prevent the element from inverting and to smooth the element when it is quadratic. It is this penalty term that makes the smooth algorithm smooth quadratic elements as well.

3.2 α term

For a given node:

$$f(\alpha) = \sum_{i=1}^{n_e} f_i(\alpha) \tag{7}$$

where n_e is the number of elements adjacent to the smoothing node and $f_i(\alpha)$ is the contribution from the ith element to the objective function.

For triangular elements:

$$f_i(\alpha) = (1 - \alpha_i)^2 \tag{8}$$

where α_i is the shape metric α of the ith element. Similarly, for quadrilateral elements:

$$f_i(\alpha) = \sum_{j}^{3} \left(1 - \alpha_{ij}\right)^2 \tag{9}$$

where α_{ij} is the ith quad's jth sub-triangle.

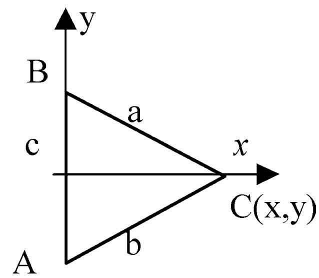


Fig. 8 The α term of the quad as shown in Fig. 4 with a rotation of 90° to put node A on the x-axis

Figure 8 shows the objective function with only the α term, as defined by Eq. 6. It is obvious that this part of objective function looks similar to that in Fig. 2.

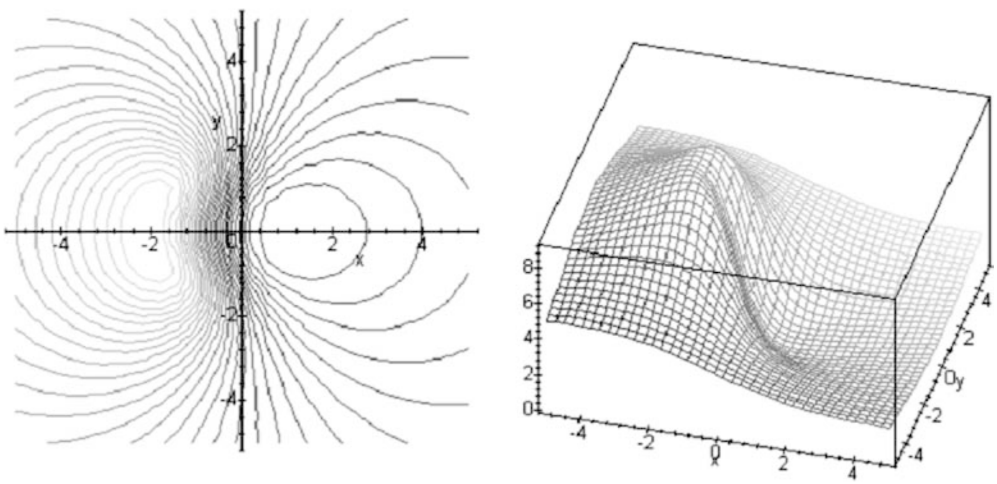
3.3 Penalty term

The penalty term $f(\theta)$ gives the objective function a monotonically decreasing property with one minimum, which is shown empirically. It is this property that makes the optimization-based smoothing algorithm able to untangle invalid meshes, as well as smooth elements. Interior angles at nodes A and B (Fig. 9) are convenient means to determine the validity of a triangle. When smoothing quadratic elements, the quadratic edge tangent vectors at nodes A and B are used to compute the angles at A and B.

However, angle A and angle B in Fig. 9 are not continuous functions on the x–y plane, Fig. 10a. There is a discontinuity on the line $\{x = 0\}$, where the angle jumps between $-\pi$ and π. Looking at node B in Fig. 9, when point C is on the positive x side near the line, the angle is π, while the other side makes the angle be $-\pi$. Therefore, the angle on the line of $\{y > 1,\ x = 0\}$ is undefined.

As discussed previously, discontinuous functions are not desired for optimization algorithms. To overcome

Fig. 9 Triangle with node C as a moving node



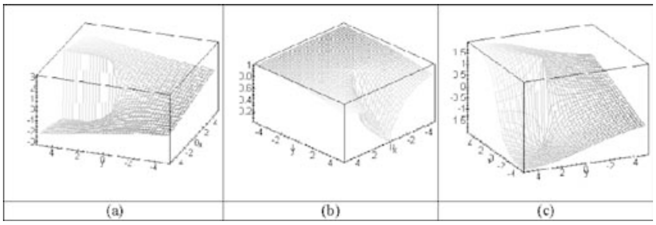


Fig. 10a–c Angle is not a continuous function in the x–y plane. **a** Angle distribution over x–y plane. **b** Weight function introduced to make adjusted angle smooth. **c** The adjusted angle

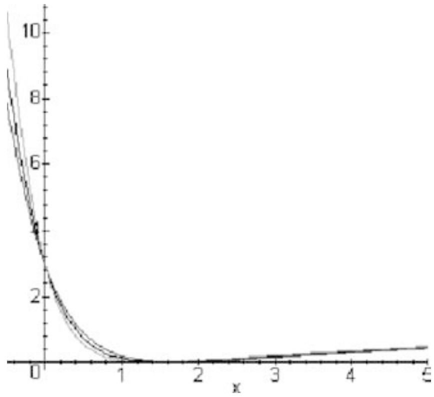


Fig. 11 Penalty term. The penalty term increases rapidly when the point is approaching $x \le 0$, which will make the triangle be an invalid element

this problem, a weight w is introduced as a multiplier to the angles to generate an adjusted angle, ω:

$$w = 1 - \mathrm{e}^{-2(\pi - |\theta|)} \tag{10}$$

$$\omega = w\theta \tag{11}$$

The adjusted angle is continuous everywhere but at points A and B. Figure 10c shows the adjusted angle for angle $\angle$ABC. As seen in the figure, there is a "jump" at the location of point B. For the angle $\angle$ABC, the penalty term is defined as:

$$f_{\mathrm{B}}(\theta) = \left(k_1 \mathrm{e}^{k_2 \omega \sqrt{\frac{a}{c}}} - 1\right)^2 \tag{12}$$

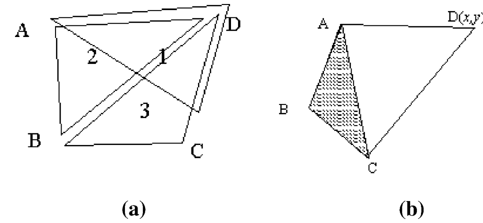


Fig. 13a, b ΔABC is the dead zone for the quadrilateral ABCD during smoothing

where $a = \left\|\overrightarrow{\mathrm{BC}}\right\|$ is the edge length of edge BC, $c = \left\|\overrightarrow{\mathrm{AB}}\right\|$ is the edge length of edge AB, and k_1 and k_2 are two constants for normalizing the metric such that $f(\theta)$ is 0 when C is moved to the position that makes the triangle an equilateral triangle.

The penalty term is a smooth function with higher order continuous derivatives. This smooth penalty term increases rapidly when an element angle approaches 0 and the negative angle region. When an element is in the valid region, the penalty is relatively small compared to the near-invalid and invalid regions, as shown in Fig. 11. The 3D plot of the penalty term is shown in Fig. 12a.

For a triangle, two penalty terms are needed, one for angle $\angle$ABC and the other one for angle $\angle$BAC. The whole penalty term is:

$$f(\theta) = f_{\mathrm{A}}(\theta) + f_{\mathrm{B}}(\theta) \tag{13}$$

The final objective function is as illustrated in Fig. 12b, c. This objective function is smooth and increases rapidly when the triangle is near inversion.

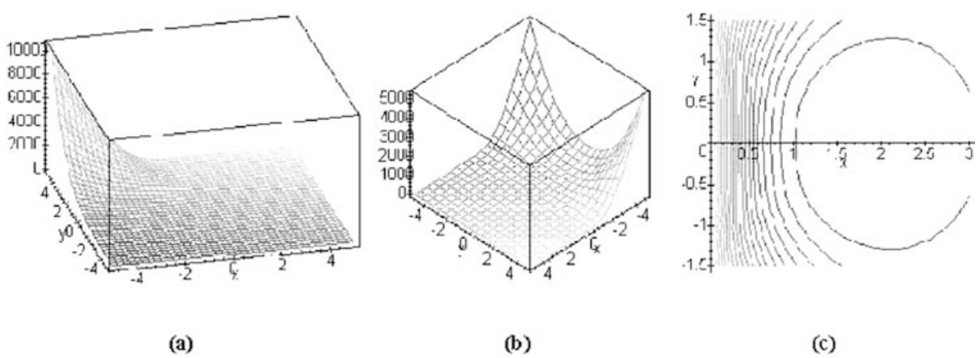
3.4 Three triangle scheme

For quadrilateral elements, the objective function, f_q, is formulated as:

$$f_q = f_1 + f_2 + f_3 \tag{14}$$

where f_1, f_2, and f_3 are the objective functions from triangles ΔADC, ΔADB, and ΔBDC, respectively, as shown in Fig. 13, where node D is the node to be smoothed.

Fig. 12a–c The penalty term and the objective function. **a** The penalty term when considering only one angle of the triangle. **b** The final shape of the objective function. **c** The contours of the final objective function in the feasible region



3.5 Mixed triangular and quadrilateral elements

The issue of how to combine the nodal metrics for a mixed mesh arises when triangular and quadrilateral elements are adjacent to the node being smoothed. The resolution of this issue is actually quite simple and straightforward. As shown in Eq. 14, the objective function for a quadrilateral is made up of three parts derived from three sub-triangles. Therefore, for a mixed mesh, the contribution of each triangular element to the total objective function should be multiplied by a factor of 3 in order to evenly weight the triangles and quadrilaterals.

4 Result and examples

Currently, the optimization-based smoothing algorithm with the proposed objective function was integrated with the Laplacian smoothing scheme, which is similar to the work in the [2]. In this section, the robustness of this kind of the combined approach with the new objection function will be demonstrated by solving some problems.

4.1 Quadratic elements

The effectiveness of the above formulation can be demonstrated by the example in Fig. 14. As clearly indicated in Fig. 14a, when the top edge is curved towards the center, the best location for the center node is below the intersection of the two dashed lines. The other two illustrations show similar results for quadrilateral elements.

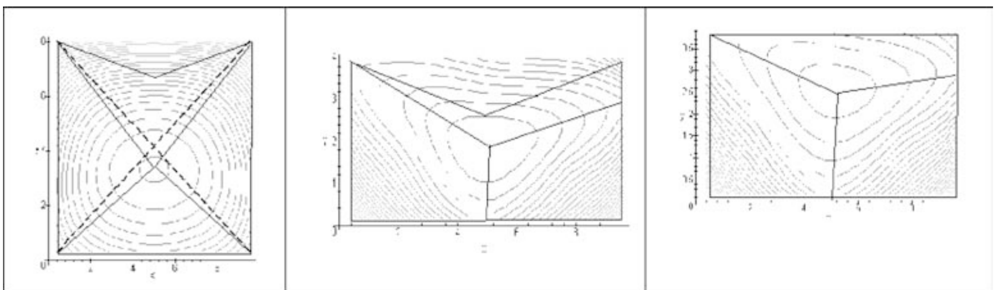
4.2 Near Convex Level Set

An example configuration similar to the one presented in [3] is used to show the result of our current work. Figure 15 illustrates that, even though the level sets outside of the valid region for the center node are non-convex, the near-convexity of the set enables faster convergence of the optimization algorithm.

4.3 Untangling

Figure 16 is an example for the untangling of our "plate-with-a-hole" model. The tangled mesh (Fig. 16a) is created by perturbing the nodal locations of each interior node (a node not on any surface boundaries) randomly from the mesh in Fig. 16b. Just like almost all the local smoothing algorithms, our smoothing works in iterations. It takes three iterations for this model to be untangled, as shown in Fig. 17a–c, and one more iteration to turn the mesh back to the one shown in Fig. 16b.

In our current implementation, a node is smoothed with optimization-based smoothing if the node is con-

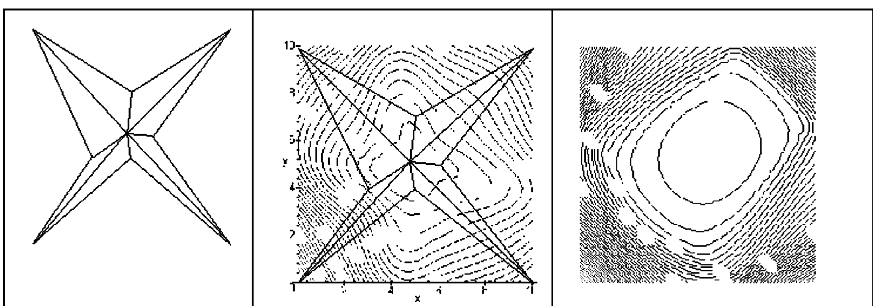


(a) a rectangular plate made up of 4 triangular elements with the top edge curved

(b) a rectangular plate made up with 3 quad elements with top edge curved

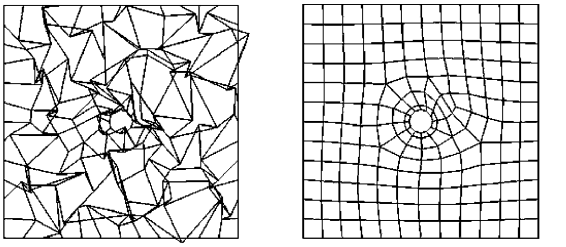
(c) a rectangular plate made up with 3 quad elements with all straight edges

Fig. 14a–c Configuration with one quadratic edge



(a) (b) (c)

Fig. 15a–c An example with a star-shaped configuration to illustrate the level sets in the feasible region



(a) tangled plate with a hole (b) untangled plated with a hole

Fig. 16a, b Untangling example

nected to an inverted element. Therefore, in untangling, based on this rule, the majority of the nodes in Fig. 17 are actually untangled by the use of optimization-based smoothing.

4.4 More quadratic elements

Figure 18 is a unit test case where a node is connected to a curved boundary sharing two quadratic boundary elements. This is a difficult case for most smoothing algorithms that deal only with linear elements. Figure 18a is the result with only constrained-Laplacian smoothing while Fig. 18b is the result using the proposed objective function with optimization-based smoothing.

Figure 19 illustrates another unit test example. Similar to the above example, a node is connected to some curved quadratic boundary elements. The optimization-based smoothing yields a satisfactory result.

Figures 20 and 21 show examples of high quality meshes generated using the new smoothing algorithm.

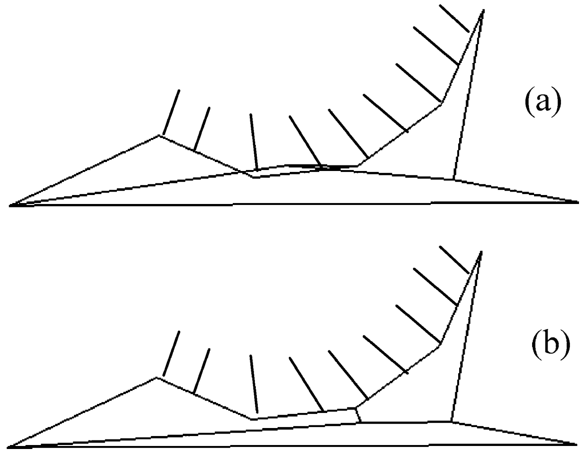
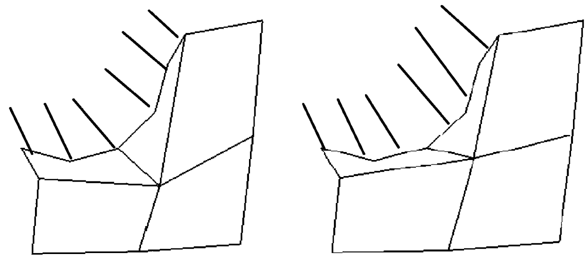


Fig. 18a, b A node with three edges close to a curved boundary. **a** Smoothed by Laplacian smoothing. **b** Smoothed with our new objective function



(a) (b)

Fig. 19a, b Quadratic curved boundary quads smoothing. **a** Smoothed by our new objective function. **b** Smoothed with Laplacian smoothing

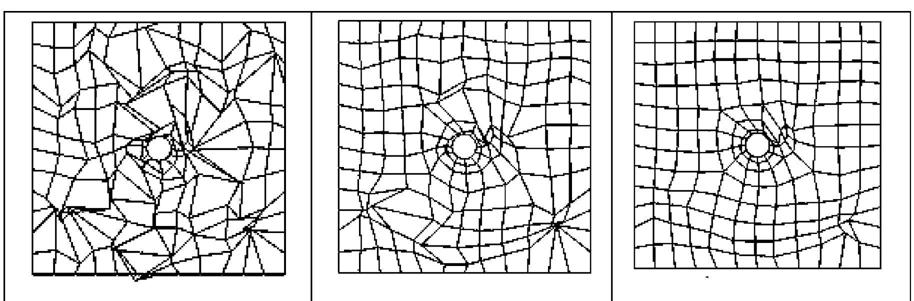
5 Conclusions and future work

A new objective function for optimization-based smoothing is proposed for linear and quadratic triangular and quadrilateral elements. Unlike the current popular approaches, the new objective function makes it possible to untangle and smooth elements in a single process. The objective function has higher order continuous derivatives and only one minimum, if any. This makes it suitable for optimization techniques. The results show that the Laplacian/optimization smoothing scheme with the proposed objection function is not only able to untangle invalid elements, but also produce high quality meshes.

Future work in this area may include:

- Speed improvement on metric calculation so that we can use optimization smoothing more often
- Mathematically prove properties of the objective function
- Extend the objective function to solid elements

Fig. 17a–c Untangling results after each iteration



a) iteration 1 b) iteration 2 c) iteration 3

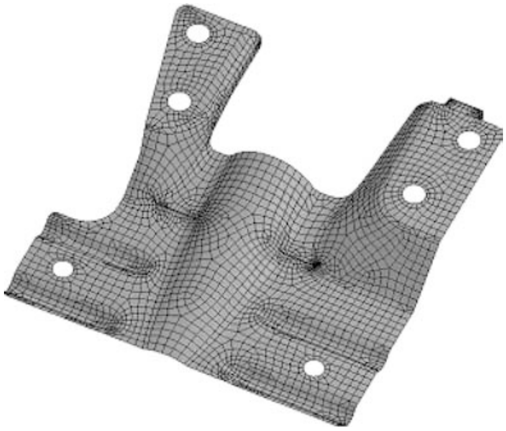


Fig. 20 Example surface mesh

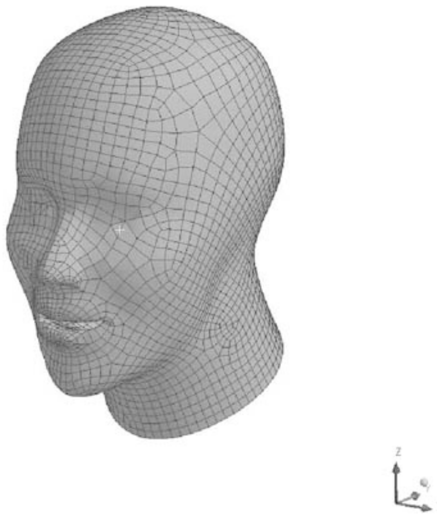


Fig. 21 Example surface mesh: human head

References

1. Owen SJ (1998) A survey of unstructured mesh generation technology. In: Proceedings of the 7th international meshing roundtable, Dearborn, Michigan, October 1998. Sandia National Laboratories, pp 239–267
2. Canann SA, Tristano JR, Staten ML (1998) An approach to combined Laplacian and optimization-based smoothing for triangular, quadrilateral, and quad-dominant meshes. In: Proceedings of the 7th international meshing roundtable, Dearborn, Michigan, October 1998. Sandia National Laboratories, pp 479–494
3. Freitag LA (2001) Local optimization-based untangling algorithms for quadrilateral meshes. In: Proceedings of the 10th international meshing roundtable, Newport Beach, California, October 2001. Sandia National Laboratories, pp 397–406
4. Field D (1988) Laplacian smoothing and Delaunay triangulations. Commun Numerical Methods 4:709–712
5. Freitag LA (1997) On combining Laplacian and optimization-based smoothing techniques. In: Trends in unstructured mesh generation. ASME Applied Mechanics Division AMD-vol 220, pp 37–44
6. Fletcher R, Reeves CM (1964) Function minimization by conjugate gradients. Comput J 7:149
7. Kwok W, Chen Z (2000) A simple and effective mesh quality metric for hexahedral and wedge elements. In: Proceedings of the 9th international meshing roundtable, New Orleans, Louisiana, October 2000. Sandia National Laboratories, pp 325–333
8. Knupp PM (1999) matrix norms and the condition number: a general framework to improve mesh quality via node-movement. In: Proceedings of the 8th international meshing roundtable, South Lake Tahoe, California, October 1999. Sandia National Laboratories, pp 13–22
9. Freitag LA , Plassmann PE (1999) Tetrahedral element shape optimization via the Jacobian determinant and condition number. In: Proceedings of the 8th international meshing roundtable, South Lake Tahoe, California, October 1999. Sandia National Laboratories, pp 247–258
10. Paoletti S (2002) Polyhedral mesh optimization using the interpolation tensor. In: Proceedings of the 11th international meshing roundtable, Ithaca, New York, September 2002. Sandia National Laboratories, pp 19–28
11. Lee CK, Lo SH (1994) A new scheme for the generation of a graded quadrilateral mesh. Comput Struct 52(5):847–857
12. Chen Z, Tristano J, Kwok W (2003) Combined Laplacian and optimization-based smoothing for quadratic mixed surface meshes. In: Proceedings of the 12th international meshing roundtable, Santa Few, New Mexico, September 2003. Sandia National Laboratories, pp 361–370