LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-CONF-753019

An Angular Method with Position Control for Block Mesh Squareness Improvement

J. Yao, D. Stillman

June 13, 2018

27th International Mesh Roundtable
Albuquerque , NM, United States
October 1, 2018 through October 5, 2018

Disclaimer

This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

An Angular Method with Position Control for Block Mesh Squareness Improvement

Jin Yao, Douglas Stillman

Abstract We optimize a target function defined by angular properties with a position control term for a basic stencil with a block-structured mesh, to improve element squareness in 2D and 3D. Comparison with the condition number method shows that besides a similar mesh quality regarding orthogonality can be achieved as the former does, the new method converges faster and provides a more uniform global mesh spacing in our numerical tests.

1 Introduction

Mesh orthogonality, if achieved, reduces computational errors by eliminating the cross-terms in the truncation error therefore a more accurate result can be obtained in a numerical simulation. There are various way to approach mesh orthogonality and the condition-number [1] mesh smoothing is among the most widely employed in practices. It minimizes a target function defined by the ratio between the sum of element edge length squared and a power of the volume consistent with the dimension of length squared, defined for a corner for each corner in an element. When orthogonality is achieved, the volume of a quad (or hex) element takes its maximum possible value thus the target function is minimized. In general, the condition-number method often provides good element shapes and it also works in the case of an arbitrary mesh.

The condition-number method provides square elements when the boundary of mesh is consistent with orthogonality. However it also provides relatively small sizes

Jin Yao
Lawrence Livermore National Laboratory, 7000 East Avenue, Livermore, CA 94551, U.S.A. e-mail: yao2@llnl.gov

Douglas Stillman
Lawrence Livermore National Laboratory, 7000 East Avenue, Livermore, CA 94551, U.S.A.e-mail: stillman2@llnl.gov

around a reduced-connectivity point. Sometimes it pulls the mesh surfaces toward a concave boundary and causes thin or even flipped elements. These are not desired and may limit the time step in a simulation with the Courant-Friedrichs-Levy (CFL) condition to control instability, or even crush the run. We have also observed a relatively slow convergence with the condition-number method, especially when the initial mesh is twisted.

A relatively fast mesh-smoothing approach is proposed. The method is almost parallel to the condition-number method in the two-dimensional case, except with a difference target function to minimize. In three-dimensions the new method sums up target functions similar to the 2D one, defined on the three logically 2D stencils shared by a given center node. Therefore the new method is easier to implement than the condition-number method in 3D.

Besides providing element squareness as the condition-number method does, the new method converges much faster and provides globally more uniform mesh sizes.

Abstract A quartic target function for optimization with an additional position control term is introduced.

2 An angle-based quartic target function

In (fig. 1) a basic two-dimensional stencil is shown on the left. The red node at the center is to move and one may attempt to make the sum of squared angle cosines of the α, β angles as small as possible for orthogonality. In the ideal case, the four α angles (with the center node as the tip) and the eight β angles (with middle nodes on

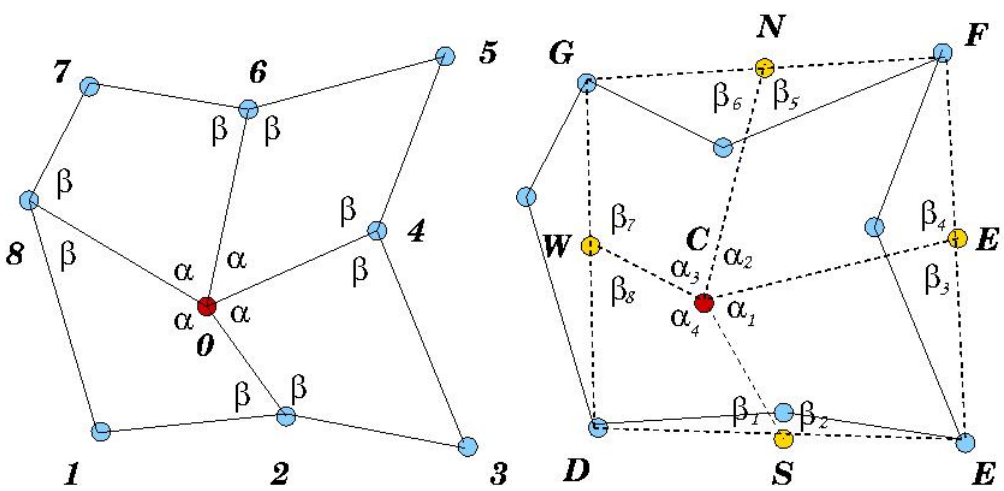

Fig. 1 Left: A normal stencil in 2D with 9 nodes; right: a simple stencil modified from the left. Yellow points S(south), N(north), W(west), and E(east) are mid-points on faces of the simplified stencil DEFG (a quad defined by dashed lines).

each wall of a regular stencil as tips) would all be $\pi/2$ and the sum of their cosines is *zero*. However, with an initially much twisted mesh a Newton's method for minimizing the sum of cosine squared often diverges or finds undesired solutions.

To simplify the problem we consider that a mesh would have nearly straight mesh lines after smoothing. Thus, we modify the basic stencil in fig. 1 left by ignoring the middle nodes on the four walls of the stencil. To provide some position control we take the four midpoints on the faces of the resulting quad and call them S(south), E(east), N(north), and W(west) (fig. 1, right). Then we link the node at the center node C to S,E,N, and W (as desired node positions). Then 12 angles are formed with $\alpha_i (i = 1,2,3,4)$ stand for the corners with C as their tip and $\beta_i (i = 1,2,3,...8)$ with the *four* mid-points (S,E,N, and W) on faces as corner tips. Since in the ideal case with orthogonality all the α and β are right angles. We write a target function as the follows

$$T = \frac{1}{2}\left(\sum_{i=1}^{4}\cos^2(\alpha_i) + \sum_{i=1}^{8}\cos^2(\beta_i)\right). \tag{1}$$

For each angle contained in the above function, the square of cosine is computed by the square of the *inner-product* of the two vectors defined by a corner tip and the two closest nodes in the stencil that define the corner, divided by the product of the length squared of the two vectors. For examples (in fig. 1)

$$\cos^2(\alpha_1) \equiv \frac{(\mathbf{CS}\cdot\mathbf{CE})^2}{|CS|^2\cdot|CE|^2},$$

and

$$\cos^2(\beta_1) \equiv \frac{(\mathbf{SC}\cdot\mathbf{SD})^2}{|SC|^2\cdot|SD|^2}.$$

The target-function defined above, when being minimized by a Newton's method, still sometimes diverges or finds spurious roots. To achieve stability we simplify the target function further by fixing the denominators in the Newton's iterations. This is to say the the leg-lengths in a smoothing iteration are taken to their values of the previous iteration. As the smoothing converges, the length of a given leg gradually reaches its limiting value. Thus, if the proposed method converges, the above approximation of cosine with the target function in (eq. 1) shall not change the solution, however, it simplifies the algebra quite a bit.

Then we are left with a positively definite quartic function (eq. 1) to minimize. This is a well defined algebraic problem and the target function behaves well in an optimization by the Newton's method.

2.1 A local optimization with the Newton's method

In a smoothing step of the proposed method we loop over all internal nodes (boundary nodes are assumed fixed) and for each internal node, the simplified target func-

tion can be written as

$$T = \frac{1}{2}\sum_{i=1}^{12} c_i\left((x-a_{1i})(x-a_{2i})+(y-b_{1i})(y-b_{2i})\right)^2. \tag{2}$$

The subscript 'i' stands for the contribution of angle 'i' and there are 12 angles in total. However when a corner-node is also a reduced connectivity point in a regular stencil, because the mesh-line there is not expected to be straightened, we ignore contributions of corners with the reduced-connectivity as their tip. The constants a_{1i}, a_{2i}, b_{1i}, b_{2i}, and c_i are all in terms of coordinates of the simplified stencil (fig. 1, right). The iterator 'i' counts α angles then β ones. For example the first term in the sum would be

$$\cos^2(\alpha_1) = \frac{((x-x_S)(x-x_E)+(y-x_S)(y-y_E))^2}{[(x_C-x_S)^2+(y_C-y_S)^2][(x_C-x_E)^2+(y_C-y_E)^2]}$$

which is in the format of (eq. 2) with

$$a_{11} = x_S, a_{21} = x_E; b_{11} = y_S, b_{21} = y_E; \text{ and}$$

$$c_1 = \omega_i[(x_C-x_S)^2+(y_C-y_S)^2]^{-1}[(x_C-x_E)^2+(y_C-y_E)^2]^{-1}.$$

In the above expression x, y are the coordinates to be updated of the center node C. x_C, y_C are the coordinates of C at the last smoothing iteration, similar with x_E, y_E and x_S, y_S. ω_i are weights. In this paper we have taken $\omega_i = 1$ an constant unless with a reduced connectivity corner-tip, for which ω_i is taken to 0.

At a minimizer one must have $\partial T/\partial x = 0$ and $\partial T/\partial y = 0$ where

$$\frac{\partial T}{\partial x} = \sum_{i=1}^{12} c_i(2x+a_{1i}+a_{2i})\left((x+a_{1i})(x+a_{2i})+(y+b_{1i})(y+b_{2i})\right)$$

$$\frac{\partial T}{\partial y} = \sum_{i=1}^{12} c_i(2y+b_{1i}+b_{2i})\left((x+a_{1i})(x+a_{2i})+(y+b_{1i})(y+b_{2i})\right) \tag{3}$$

With the Newton's method for optimization the second derivatives of the target function are also needed

$$\frac{\partial^2 T}{\partial x^2} = \sum_{i=1}^{12} c_i[(2x+a_{1i}+a_{2i})^2+2\left((x+a_{1i})(x+a_{2i})+(y+b_{1i})(y+b_{2i})\right)]$$

$$\frac{\partial^2 T}{\partial y^2} = \sum_{i=1}^{12} c_i[(2y+b_{1i}+b_{2i})^2+2\left((x+a_{1i})(x+a_{2i})+(y+b_{1i})(y+b_{2i})\right)]$$

$$\frac{\partial^2 T}{\partial xy} = \sum_{i=1}^{12} c_i[(2x+a_{1i}+a_{2i})(2y+b_{1i}+b_{2i})] \tag{4}$$

Because the target function is a sum of squares, a minimizer must exists. The position of the minimizer is not related to the original position of a given node. To start with, the initial guess of the minimizer is taken to the geometrical center of a quad formed by S, E, N, and W

$$x_0 = \frac{1}{4}(x_S + x_E + x_N + x_W),$$

$$y_0 = \frac{1}{4}(y_S + y_E + y_N + y_W).$$

This can be justified by considering that when mesh orthogonality and even mesh-spacing are achieved, this initial guess would be exactly the solution point.

A Newton's iteration is then carried out with

$$x_1 = x_0 - \left[\frac{\partial^2 T}{\partial x \partial y}\frac{\partial T}{\partial y} - \frac{\partial^2 T}{\partial y^2}\frac{\partial T}{\partial x}\right] / \left[\frac{\partial^2 T}{\partial x^2}\frac{\partial^2 T}{\partial y^2} - (\frac{\partial^2 T}{\partial x \partial y})^2\right]$$

$$y_1 = y_0 - \left[\frac{\partial^2 T}{\partial x \partial y}\frac{\partial T}{\partial x} - \frac{\partial^2 T}{\partial x^2}\frac{\partial T}{\partial y}\right] / \left[\frac{\partial^2 T}{\partial x^2}\frac{\partial^2 T}{\partial y^2} - (\frac{\partial^2 T}{\partial x \partial y})^2\right]. \tag{5}$$

We perform the above Newton's iteration (eq. 5) only once in a smoothing step for each internal node to obtain its improved position (x_1, y_1). Then a loop over all the nodes moves each node to its improved position. There maybe nodes at singular connectivity points that do not own a normal stencil. In this case we simply take the geometrical average of the three nodes directly linked to the center node by simple legs. We stop the smoothing when the global L_2 difference between the results of two consecutive smoothing steps is smaller than a preset threshold, or a preset limit of the number of iterations is reached.

2.2 An additional position control term

Minimizing the target function defined in the last section usually give good mesh quality that is comparable to the condition-number method with fewer steps and better mesh-spacing. However, when the element aspect ratio is big, the contribution from corners that have short sides sometimes cannot balance contributions from other corners. As a result the minimizer maybe located exterior to the stencil while the mesh squareness is reasonably achieved locally. To avoid this situation, we add a position control term to the target function proposed in (eq.1) defined by the distances from center node to the mid-face points squared

$$U = \frac{1}{2}[(x-x_S)^2 + (y-y_S)^2 + (x-x_E)^2 + (y-y_E)^2 + (x-x_N)^2 + (y-y_N)^2 + (x-x_W)^2 + (y-y_W)^2], \tag{6}$$

and the target function to minimize is then becoming

$$T + \sigma U. \tag{7}$$

The factor σ is proportional to the aspect ratio defined by $|NS|^2/|WE|^2$ or its reverse, whichever is bigger. If minimizing T alone takes a node out of its stencil, the term σU will move the node back in the stencil. There are other possibilities for a position control function. We use the current one because it works well in all the cases we have tested. $\sigma = 1$ is taken in the numerical tests but it is certainly an adjustable parameter and can be taken to zero in suitable cases.

It should be pointed out that simply taking the geometric average of S, E, N and W (the initial guess chosen above) also smooths the mesh (and it can be seen as a Laplacian smoothing [7]), but does not provide a more uniform mesh spacing. An example about this will be showed in a later section.

3 The three-dimensional case

In three dimensions the target function is the sum of three target functions defined on logically orthogonal two-dimesional stencils.

3.1 The stencil simplified with direction-nodes

In three-dimensions, a given regular interior node (i.e., not at a singular point) is shared by three logically two-dimensional regular stencils perpendicular to each other (fig. 2, left), one in each logical direction. To achieve three-dimensional mesh orthogonality, it is necessary that the two-dimensional mesh orthogonality is obtained in each of the logically 2D spatial normal stencil. In each logical direction, we define a pair of direction-nodes (similar to the S, E, N, W nodes in the 2D case). Each direction node is the geometrical average of the yellow nodes in the 2D case to allow each pair of perpendicular logically 2D stencils sharing the same direction nodes (in order to achieve a even mesh-spacing). The three modified logically 2D stencil are shown in (fig. 2, right).

3.2 The target function

Thus, we propose to sum the three two-dimensional target functions associated with a given regular node for a three-dimensional target function to minimize. The two vectors involved in an inner product (to compute cosine of an angle) are both three-dimensional in this case. The target function has *three* independent variables x, y, z instead of only *two* in the previously described two-dimensional case.

We pick an initial guess which is the geometrical average of the positions of the direction-nodes, similar to the 2D case.

A Newton's iterative scheme can again be used to minimize the quartic target function. Although it is a common practice, we still write down the standard procedure for minimizing a general function $F(x,y,z)$ for reference. Let the initial guess of the minimizer be (x_0,y_0,z_0). Because the condition $\nabla F = 0$ must be satisfied at a minimizer, we expand the gradient of F around (x_0,y_0,z_0) and keep only the linear terms, then solve the following 3 by 3 linear system

$$\begin{bmatrix} \partial^2 F/\partial x^2 & \partial^2 F/\partial x\partial y & \partial^2 F/\partial x\partial z \\ \partial^2 F/\partial x\partial y & \partial^2 F/\partial y^2 & \partial^2 F/\partial y\partial z \\ \partial^2 F/\partial x\partial z & \partial^2 F/\partial y\partial z & \partial^2 F/\partial z^2 \end{bmatrix} \cdot \begin{bmatrix} \delta x \\ \delta y \\ \delta z \end{bmatrix} = - \begin{bmatrix} \partial F/\partial x \\ \partial F/\partial y \\ \partial F/\partial z \end{bmatrix}. \tag{8}$$

Then the position of the solution gets updated by $x_1 = x_0 + \delta x$, $y_1 = y_0 + \delta y$, and $x_1 = x_0 + \delta z$.

The above Newton's scheme is performed for a single time in each smoothing step. The three-dimensional target function taken for minimization is defined as the follows

$$F = (P_I + \sigma_I U_I) + (P_J + \sigma_J U_J) + (P_K + \sigma_K U_K), \tag{9}$$

where I, J, and K stand for the three logical directions, P, σ, and U have the same definitions as in the two-dimensional case except that the position of a point has *three* components x, y, and z in this case.

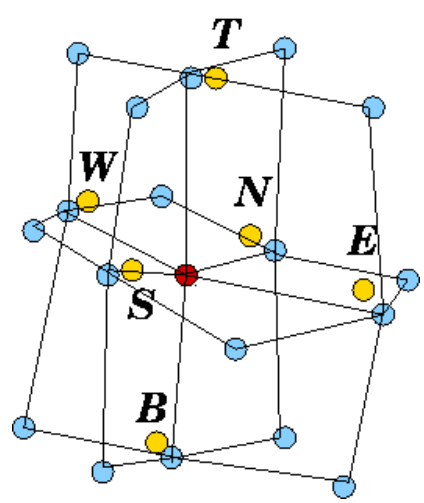

Fig. 2 A regular node in three-dimensions is shared by three logically two-dimensional regular stencils. The left figure shows the said stencils defined with original nodes (blue). The yellow nodes B(bottom), T(top), W(west), E(east), S(south), and N(north) are 'direction-nodes' with each marks a logical direction and defined by the geometrical average of the four blue nodes directly linked to a corresponding blue node at the center of a wall of the stencil. The target function is defined for the modified stencil showed on the right.

4 Comparison with the condition-number method

The proposed method can somehow be seen as a condition-number method with its target function replaced by (eq. 1) or (eq. 9). Therefore it should be easy to modify an existing condition number implementation to perform the proposed method.

In all numerical tests we have tested, the proposed method provides a faster convergence and a globally more uniform mesh with better spacing around reduced connectivity points. The mesh-squareness provided by the proposed method is at least comparable to the condition number method in all the cases.

In the following test problems we compare the results obtained with the proposed method the condition-number method. Results obtained with an angle-based method [2], the equi-potential method [3], and an equal-distance method [6] are also provided for references.

The treatment for nodes at reduced connectivity points is the same with all the methods, by taking the geometrical center of the triangle formed by the nodes directly linked to a reduced-connectivity point.

4.1 A deformed butterfly mesh

A two-dimensional five-block butterfly configuration is contained in a perfect 6 by 6 square. Each block are assigned an algebraic mesh by a bi-linear mapping from an ideal mesh in the parametric space to the physical space.

In the first case the center block is twisted counter-clock-wise by 60 degrees (fig. 3). Each block is assigned a 15 by 15 mesh. Fig. 4 shows the comparison between the proposed method and other well known methods.

Clearly, with a fixed smoothing steps, the proposed method produced symmetry and the largest mesh size around a reduced connectivity point. The condition num-

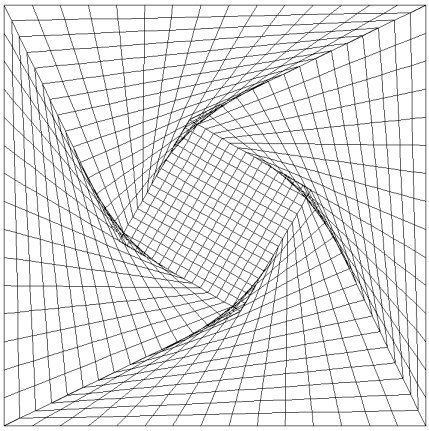

Fig. 3 A butterfly mesh on a 6 by 6 square with the center block twisted counter-close-wise by $\pi/3$.

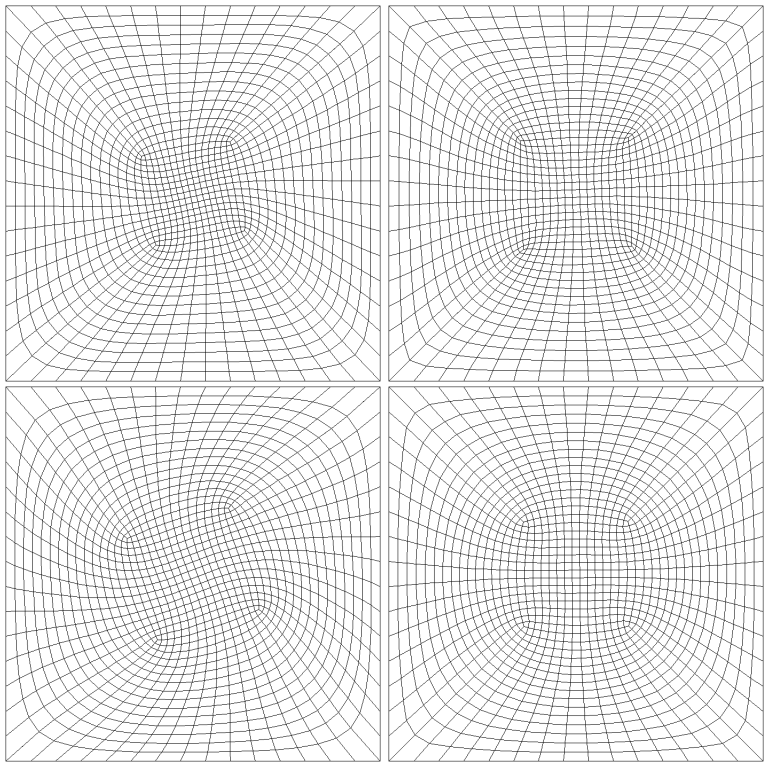

Fig. 4 The initial mesh in fig. 3 smoothed with 160 iterations: upper left: with the angle-based method; upper right: with the Winslow-Crowley method; lower left: with the condition-number method; lower right: with the proposed method.

ber method and the angle-based method converge slower. The Winslow-Crowley

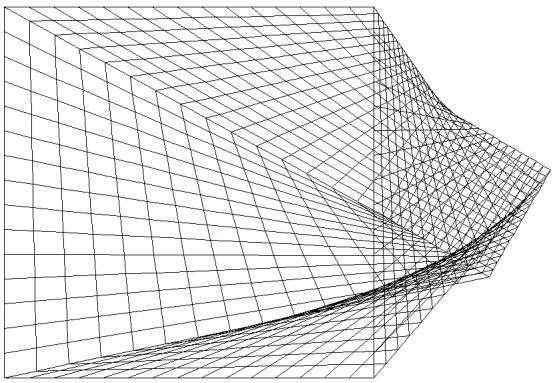

Fig. 5 A butterfly mesh on a 6 by 6 square with the center block twisted by 60 degree and shifted by 9 units in the x-direction.

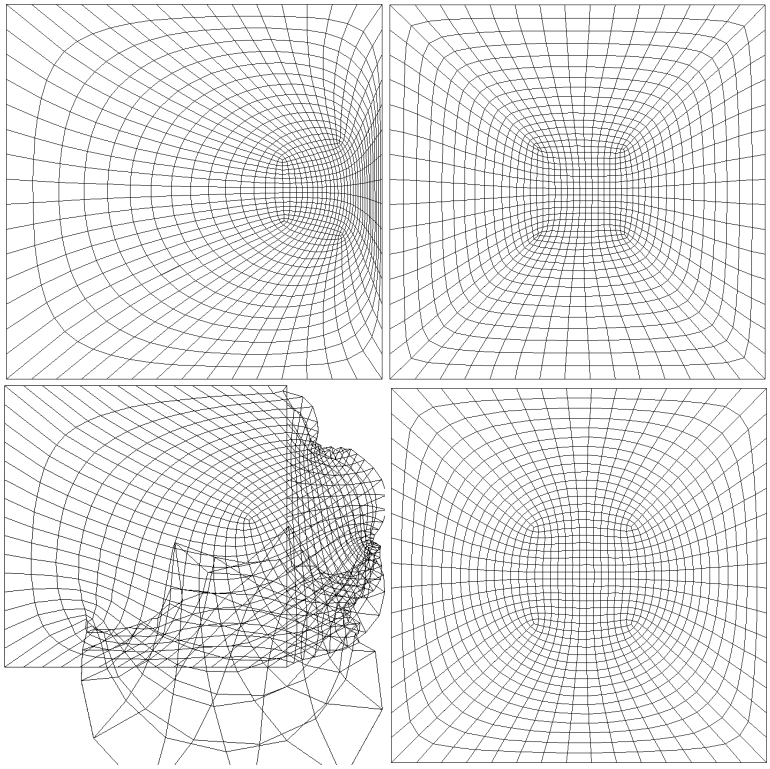

Fig. 6 The initial mesh in fig. 5 smoothed with 800 iterations: upper left: with the angle-based method; upper right: with the Winslow-Crowley method; lower left: with the condition-number method; lower right: with the proposed method.

method, although converges not as slow (but slower than the proposed method), produces undesirably small elements around the reduced connectivity points.

In the second case the twisted center then is shifted in the x-direction with a distance of 9 units (fig. 5).

Fig. 6 shows the comparison between the proposed method and other well known methods. In this the condition-number method did no converge. The proposed method not only converges, but also provides the best mesh quality among all the methods.

To demonstrate the contribution of the target function that include the angular terms, we compare the results between simply taking the geometric average of S, E, N, and W in fig. 1 (the initial guess), against with the Newton's step for minimizing the target function. Fig. 7 show their difference. The result with minimizing the target function clearly has a bigger center box which is consistent with globally more uniform mesh sizes.

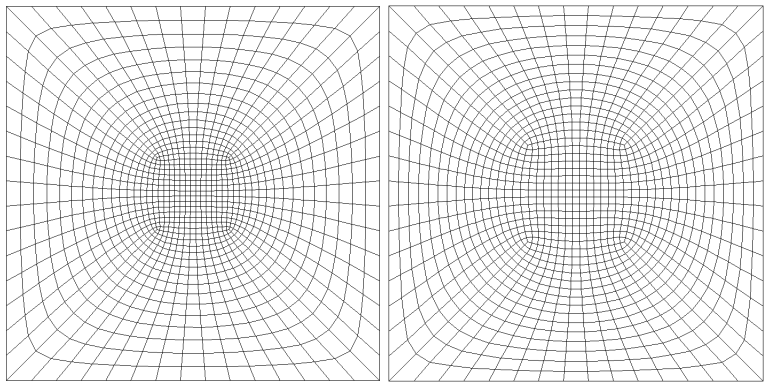

Fig. 7 The initial mesh in fig. 5 smoothed with 2000 iterations for convergence: left: without the Newton's step for minimizing the target function (taking the geometric average of S, E, N, and W in fig. 1); right: with the Newton's step (the proposed method).

4.2 On a curved surface

Fig. 8 shows a three-block initial mesh which is on the inner side of a $1/8$ spherical shell with a radius of 15 units centered at the origin. We perform smoothing with four approaches: the condition number [1]; the equipotential [3]; the equal-distance [6]; and the proposed method for orthogonality and even mesh spacing. Each smoothing method is applied for 500 times. The stencil associated with each node is projected on a plane tangential to the sphere $r = 15$ at the given point to allow a 2D smoothing step, then the improved node position is projected back to the original surface at the end of each iteration for updating the node later with another loop over all nodes. The results of smoothing are shown in fig. 9. The original

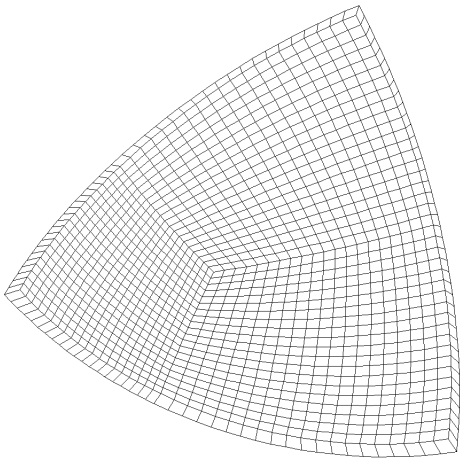

Fig. 8 The initial three-block mesh on the inner side of a $1/8$ spherical shell. The smaller block has element dimensions nearly half of the dimensions of elements in rest of the blocks.

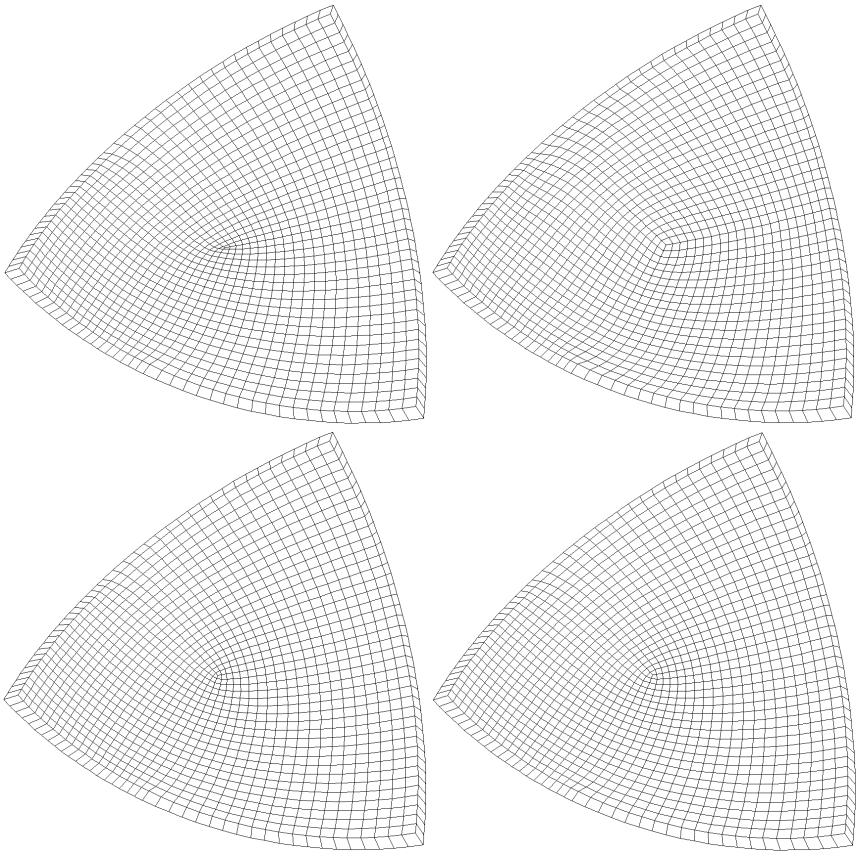

Fig. 9 The initial mesh in fig. 8 smoothed with 500 iterations: upper left: with the original equi-potential method; upper right: with the equal-distance method. lower left: with the condition-number method; lower right: with the proposed method;

equi-potential method (upper left figure) does a fair job for mesh squareness, however it also produces relatively small mesh-sizes around the reduced-connectivity point. The equal-distance method (upper-right figure), which is aimed at producing a globally uniform mesh size, produces the largest mesh sizes around the reduced-connectivity point. It does not produce the best squareness of elements because orthogonality is not a concern with the method. The condition-number method (lower left figure) does provide mesh-squareness, but still generates small mesh-sized around the reduced-connectivity point. The proposed orthogonality method also provide mesh-squareness, nevertheless, its smallest mesh-size is larger than produced by the condition-number method. This is consistent with the previous examples.

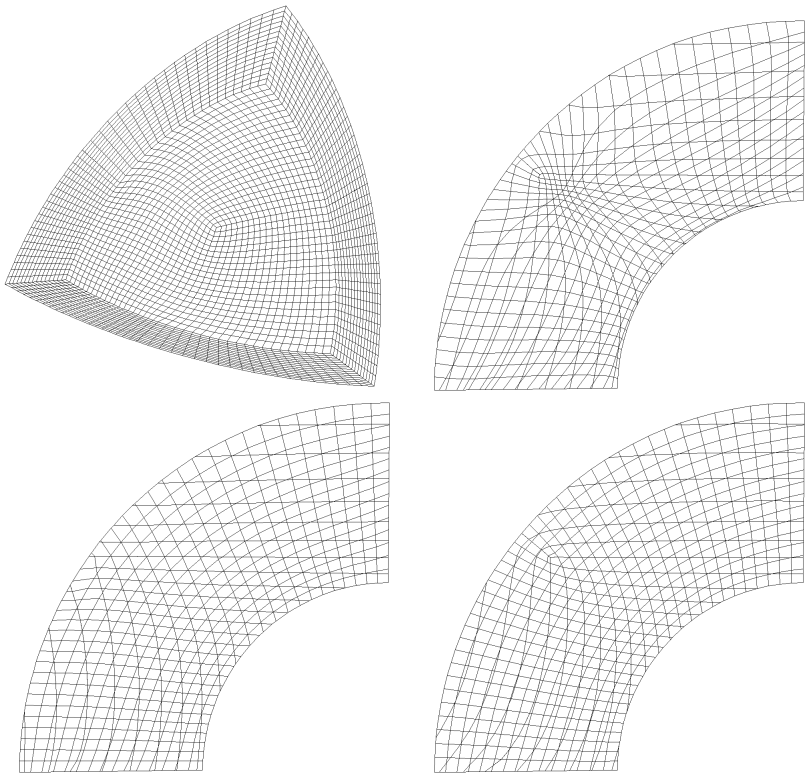

Fig. 10 A 3D shell-mesh is smoothed with 800 iterations for convergence with four methods: upper left: the eqal-distance method; upper right: condition-number method; lower left: the angle-based method; lower right: the proposed method. The surface mesh with the equal-distance smoothing is shown for providing a $3D$ view of the region. The meshes smoothed with the other methods are sliced with the $y-z$ plane passing the origin. The three surface meshes on the sides of the upper left figure are fixed.

4.3 A three-dimensional shell mesh

In fig. 10 (upper-left), a *one-eighth* spherical shell is meshed with three blocks and this is essentially the region in the previous test with a thickness of 5 units in the radius direction. The inner radius is 10 units. The condition number method generates thin elements near the inner surface and this is can be seen with slicing the mesh after smoothing with the $x = 0$ plane (fig. 10, upper right). The angle-based method does a better job (fig. 10 lower left), but the proposed method behaves the best (fig. 10, lower right). This example clearly demonstrates that the proposed method is able to generate mesh surfaces that resist the attraction of a concave boundary, and provide a better global mesh-quality than the condition-number method.

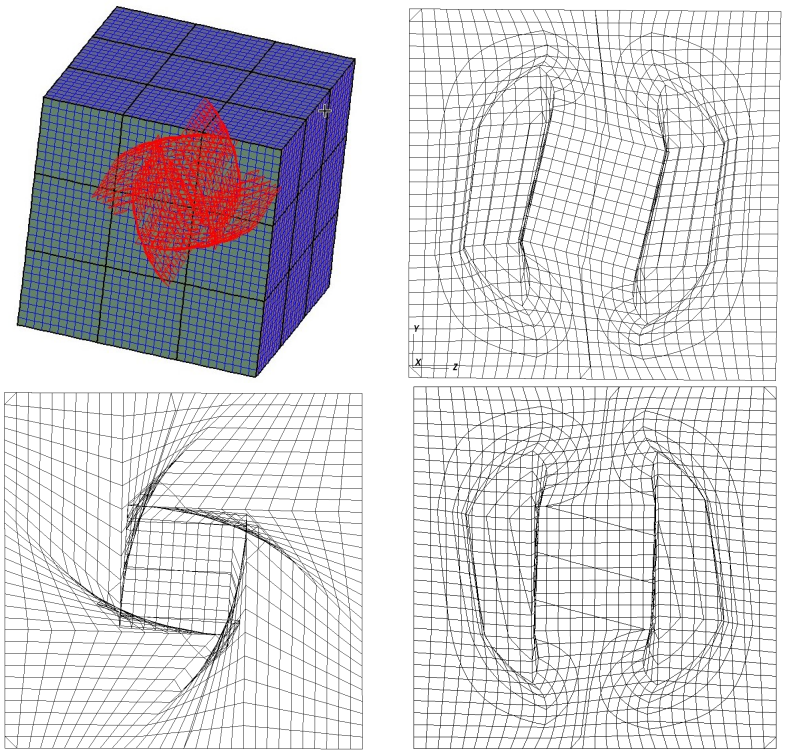

Fig. 11 In the upper left figure, the elements marked by red color are flipped because of GG consecutive $(5\pi/12)$ rotations applied to the center block of a 3 by 3 by 3 perfect cubic block-mesh. Upper right: sliced by the $x = 0$ plane; lower left: sliced by the $y = 0$ plane; lower right: sliced by the $z = 0$ plane.

4.4 A three-dimensional block mesh with center twisted

We show here the comparison between various smoothing methods with three mesh-quality metrics for another three-dimensional test problem. They are chosen to be: a) the relative minimum element size; b) the smallest two dimensional angle; and c) the aspect ratio of an element.

The minimum element size is computed with the volume of a given element divided by the maximum face area of the same element. The smallest angle is computed by finding the largest inner product of a pair of element-edges that share the common tip, divided by the products of the two edge-lengths, then taking its inverse cosine. The aspect ratio is computed with the longest diagonal divided by the smallest edge size for each element, then taking the maximum among them.

We take a cubic block structure with *3 by 3 by 3 = 27* blocks centered at the origin. Each block is a cube with an edge-length of 2. Initially, the block at the center is rotated by $5\pi/6$ around the x-axis, then rotated another $5\pi/6$ around the

y-axis, finally the center is again rotated by $5\pi/6$ around the z-axis. Each block is then assigned a *10 by 10 by 10* algebraic mesh.

The ideal metrics for a perfectly smoothed cubic mesh would be the smallest relative element size, the smallest angle, and the largest aspect ratio equal to $(1, \pi/2, \sqrt{3})$. The initial mesh metric is $(0.0000188721, 3.214596, 1505562)$. Among the $27,000$ elements in total, 490 of them are flipped (fig. 11).

Table 1 shows comparison between mesh metrics for four different smoothing methods including the condition-number method and the proposed method for orthogonality with even mesh-spacing. The other two are the equal-potential method and the angle-based method.

Clearly, the proposed method provides the best mesh metrics with fixed numbers of the smoothing steps. The equi-potential comes the second, followed by the angle-based method. The condition-number method converges the slowest. Further computation shows that for the smallest mesh size to get within a relative threshold of 10^{-3} to the ideal case, 180 smoothing steps is sufficient with the proposed method, in contrast, about 1000 steps with the condition-number method are necessary to achieve the same accuracy.

Table 1 Mesh metrics against the number of iterations regarding the initial mesh in fig. 11. 'CND' stands for condition number; 'OTR' for the proposed method; 'ANG' for angle-based; and 'EQP' for equi-potential (Winslow-Crowley). The first number in the metrics is the smallest relative side length, the second number is the minimum 2D angle in degrees, the last one is the aspect ratio defined by the longest element-diagonal divided by the shortest side-length. The ideal metrics is $(1.0, 90.0, 1.732)$.

Iters	*CND*	*OTR*	*ANG*	*EQP*
2^2	(0.001, 1.10, 9088)	(0.24, 24.3, 11.9)	(0.001, 3.52, 8395)	(0.09, 6.02, 36.5)
2^3	(0.045, 7.89, 23.2)	(0.60, 52.7, 3.39)	(0.16, 12.9, 16.9)	(0.37, 27.9, 6.24)
2^4	(0.417, 37.0, 4.77)	(0.76, 69.6, 2.42)	(0.42, 35.4, 5.07)	(0.63, 50.1, 3.33)
2^5	(0.670, 62.3, 2.83)	(0.89, 80.9, 1.99)	(0.65, 58.0, 3.02)	(0.81, 68.0, 2.38
2^6	(0.805, 76.5, 2.20)	(0.97, 87.2, 1.80)	(0.816, 74.7, 2.24)	(0.924, 80.6, 1.96)
2^7	(0.890, 84.2, 1.92)	(0.997, 89.6, 1.74)	(0.912, 84.5, 1.89)	(0.979, 87.4, 1.78)

5 Conclusion

We propose a mesh smoothing algorithm which is parallel to the condition-number method, with the target function replaced by the sum of an asymptotic expression of cosine squares of angles defined in a simplified regular stencil, plus a position control function similar to used in the Laplacian smoothing. Numerical results show that while achieving a similar quality of element squareness, the proposed method converges much quicker, and provide globally more uniform mesh size than the condition-number method. In addition, the proposed method also behaves better

than the angle-based method and the Winslow-Crowley method in our numerical tests for fixed numbers of smoothing steps.

6 Acknowledgment

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

References

1. Patrick M. Knupp, “A method for hexahedral mesh shape optimization”, Int. J. Numer. Meth. Eng. 58, pp 319-332, 2003.
2. Tian Zhou, and Kenji Shimad, “An Angle-based Approach to Two-dimensional Mesh Smoothing”, Proceedings, 9th International Meshing Round-table, Sandia National Laboratories, pp. 373-384, 2000.
3. Winslow, A. M., “Numerical Solution of the Quasi-linear Poisson Equation in a Nonuniform Triangular Mesh”, Journal of Computational Physics, Vol. 1, pp. 149-172, 1967.
4. Tipton, R. E., “Grid Optimization by Equipotential Relaxation”, Lawrence Livermore National Laboratory Report, July 15, 1992.
5. Patrick Knupp, and Stanly Steinberg, “Fundamentals of Grid Generation”, CRC Press, 1993.
6. Yao, Jin, and Douglas Stillman, ”An Equal-space Algorithm for Block-mesh Improvement”, Procdeia Engineering 163: 199-211, 2016.
7. Canann, S. A., Tristano, J., R., Staten M. L., “ An approach to combined laplacian and optimization-based smoothing for triangular, quadrilateral, and quad-dominant meshes”, Proceedings of the 7th International Meshing Roundtable, 1998.
8. Jun, B.I. “A Modified Equipotential Method for Grid Relaxation”, Lawrence Livermore National Laboratory Report, UCRL-JC-138277. March 21, 2000.
9. Tipton, R. E., “Mesh Relaxation in Elliptical Coordinates and Notch Points for ALE Hydro-Codes”, Lawrence Livermore National Laboratory Report, Nov.1, 2004.
10. Yao, Jin, “An Efficient Line-Sweep Mesh Relaxation Method with Locally Conservative Interface Reconstruction ”, Multimat Workshop, Acachon, France, 2011
11. Yao, Jin,“A Mesh Relaxation Study and Other Topics”, Lawrence Livermore National Laboratory Technical Report, LLNL-TR-637101, 2013.