

Research Article

An Iterative Mesh Untangling Algorithm Using Edge Flip

Jibum Kim

Department of Computer Science and Engineering, Incheon National University, Incheon, Republic of Korea

Correspondence should be addressed to Jibum Kim; jibumkim@inu.ac.kr

Received 4 April 2017; Revised 2 August 2017; Accepted 15 November 2017; Published 10 December 2017

Academic Editor: Stefano de Miranda

Copyright © 2017 Jibum Kim. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Existing mesh untangling algorithms are unable to untangle highly tangled meshes. In this study, we address this problem by proposing an iterative mesh untangling algorithm using edge flip. Our goal is to produce meshes with no inverted elements and good element qualities when inverted elements with poor element qualities are produced during mesh generation or mesh deformation process. Our proposed algorithm is composed of three steps: first, we iteratively perform edge flip; subsequently, optimization-based mesh untangling is conducted until all inverted elements are eliminated; finally, we perform mesh smoothing for generating high-quality meshes. Numerical results show that the proposed algorithm is able to successfully generate high-quality meshes with no inverted elements for highly tangled meshes.

1. Introduction

Geometric domains are often discretized by meshes for partial differential equation- (PDE-) based applications. Mesh qualities affect both the speed and accuracy of PDE solutions. However, poor quality and inverted elements often occur during mesh generation [1], mesh optimization [2], and mesh deformation [3]. For isotropic PDE problems such as Poisson's equation, skinny elements with very large or very small angles are not desired and are often considered as poor quality elements. An inverted element is defined as an element with negative signed volume. A tangled mesh is a mesh with any inverted elements. Inverted elements are not suitable for solving PDEs, since they result in erroneous PDE solutions. For many scientific applications such as Arbitrary-Lagrangian-Eulerian (ALE) simulations and biomedical applications, geometric domains (i.e., domain boundaries) deform as time varies. When these geometric domains deform, the meshes should be updated appropriately such that the deformed meshes have good element qualities with no inverted elements. However, when huge deformations occur, many poor-quality elements and inverted elements are produced. These poor-quality elements (also, inverted elements) deteriorate the accuracy and efficiency of PDE solutions. Most existing mesh untangling methods give high cost to inverted elements and minimize the cost function by solving nonlinear optimization problems [4–6]. However,

existing mesh untangling algorithms have limitations for highly tangled meshes and do not always untangle inverted elements.

In this study, we propose an iterative mesh untangling and smoothing method using mesh modification methods when inverted elements are produced during mesh generation or mesh deformations. Of the various mesh modification methods, edge flip minimizes the modification of mesh topology but is able to improve mesh quality. The proposed mesh untangling and smoothing method first performs edge flip to minimize the number of skinny triangles in the mesh. We numerically show that edge flip is also effective in eliminating the inverted elements in the mesh. As a second step, we perform an optimization-based mesh untangling method to eliminate inverted elements in the mesh. Our proposed algorithm iteratively repeats Steps 1 and 2 until all inverted elements are eliminated in the mesh. When all inverted elements are removed, we perform optimization-based mesh smoothing to further improve the element quality in the mesh. The effectiveness of the proposed method is demonstrated on several challenging tangled meshes, which are produced during mesh deformation process. Numerical experiments show that the proposed mesh untangling and smoothing method efficiently eliminates inverted elements in the mesh while maintaining good element qualities in the mesh with huge boundary deformations.

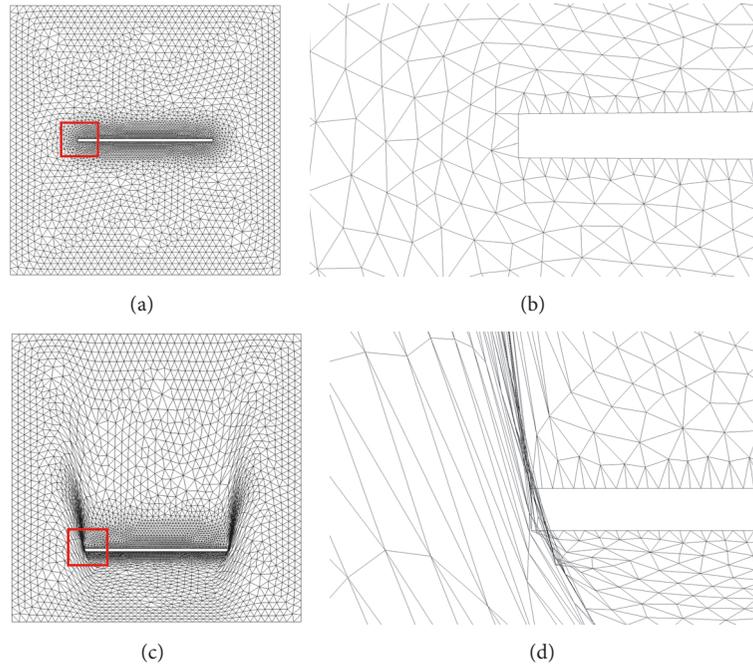


FIGURE 1: Mesh deformation example: (a) initial mesh with no inverted elements on the bar domain; (b) close-up view of the mesh in (a); (c) deformed mesh on the bar domain deformed with FEMWARP [7]; (d) close-up view of the mesh in (c).

2. Previous Work

Mesh optimization, a widely studied research topic for computer graphics, computational simulation, and biomedical applications, includes two mesh optimization methods for improving mesh quality. The first method is mesh smoothing. Mesh smoothing methods improve mesh quality by relocating vertex positions while fixing mesh topology. Mesh smoothing methods focus on improving the element qualities by optimizing the mesh quality improvement and solve (minimize) it by employing nonlinear solvers. The second approach is called mesh modification, which is based on topological changes such as edge flip, edge collapse, and edge split. Mesh modification methods have more flexibility in improving mesh qualities, as compared with mesh smoothing methods, since they permit change in the mesh topology to improve mesh quality. Several mesh optimization methods that combine both mesh smoothing and topological changes are proposed to improve the mesh quality.

On the other hand, mesh untangling is an active research topic among meshing communities. Knupp proposed the untangling-beta method that assigns high penalty to inverted elements [4, 5]. Freitag and Plassmann proposed another optimization-based mesh untangling method by maximizing the minimum area on the mesh [9]. In addition, Bhowmick and Shontz proposed a graph-based mesh untangling method [10]. However, pure mesh untangling methods focus on untangling inverted elements and often result in meshes with poor element qualities. Also, most existing untangling methods have limitations to untangle highly tangled meshes. Several simultaneous untangling and smoothing methods

are proposed [6, 11]. These methods are preferred to pure optimization-based untangling methods since they are able to untangle inverted elements while improving element qualities. Similarly, multiobjective mesh optimization method is proposed to simultaneously improve multiple aspects (e.g., shape, size, and untangling) of the mesh [3, 12]. A log-barrier mesh optimization method is proposed for mesh quality improvement and mesh untangling [11]. However, the effect of using mesh modification methods (especially, edge flip) to eliminate inverted elements on the mesh remains unclear.

3. Mesh Deformation and Inverted Elements

We focus on tangled meshes with poor element qualities. These tangled meshes with inverted elements are often produced during a mesh deformation problem. An initial mesh has good element qualities but updated (deformed) meshes on deformed domains using mesh deformation algorithms possess poor-quality inverted elements. The goal of the mesh deformation algorithm is to preserve good element qualities on deformed meshes with no inverted elements. However, for large boundary deformations, mesh deformation algorithms often produce both skinny and inverted elements on the deformed meshes [3, 7, 13]. Recently, Staten et al. compared several existing mesh deformation algorithms for real 3D mesh deformation problems [13].

Figure 1 shows a moving bar deformation example where the geometric domain undergoes a huge deformation with respect to time. Initial bar mesh with no inverted element is shown in Figure 1(a). Figure 1(b) shows a close-up view

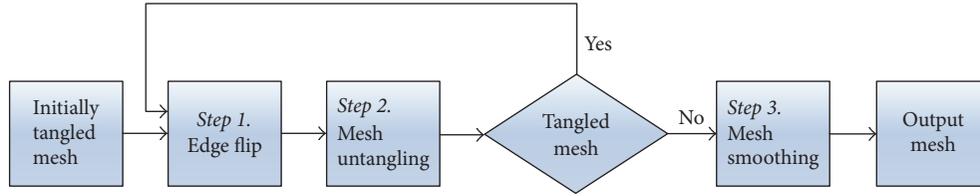


FIGURE 2: A flowchart of the proposed iterative mesh untangling algorithm.

of the mesh in Figure 1(a) (red box). Figure 1(c) shows an example of output mesh by performing the mesh deformation algorithm on Figure 1(a). We use FEMWARP mesh deformation algorithm to automatically update the mesh on the deformed domain [7]. For this problem, the output mesh by performing FEMWARP algorithm has 27 inverted elements with very skinny inverted elements on the deformed domain. Most poor-quality elements (also, inverted elements) are located near the inner boundary due to a huge deformation. Figure 1(d) shows a close-up view of the mesh in Figure 1(c) (red box) with several inverted elements and skinny elements.

4. Proposed Mesh Untangling and Smoothing Method

For highly tangled meshes with skinny elements, interior vertices are often highly constrained to move due to geometric constraints. Figure 1(c) shows one example of these cases. For these cases, performing optimization-based mesh untangling is not sufficient to eliminate inverted elements in the mesh. This is because optimization-based mesh untangling attempts to eliminate inverted element by just relocating vertex positions, while fixing mesh topology. For these cases, we propose to first perform edge flip to reduce the number of skinny elements before performing mesh untangling. Edge flip is expected to give interior vertices more spaces to move and improve element qualities before optimization-based mesh untangling is performed. Edge flip is also able to achieve large improvements in the element quality.

We accordingly propose a mesh untangling and smoothing method, as illustrated in Figure 2. When an initially tangled mesh with poor element qualities is given as an input mesh, we first perform edge flip as the first step. Our first step is designed to improve the mesh qualities by reducing the number of skinny triangles and also accelerate the second step, which eliminates inverted elements in the mesh. We numerically show that performing edge flip is effective in reducing the number of inverted elements in the mesh. As a second step, we perform optimization-based mesh untangling method to eliminate inverted elements on the deformed mesh. If the output mesh from Step 2 is a valid mesh with no inverted elements, we perform optimization-based mesh smoothing (Step 3) to further improve the mesh quality. Otherwise, we repeatedly perform Steps 1 and 2 until all inverted elements are eliminated. Details on each step are shown in Figure 2.

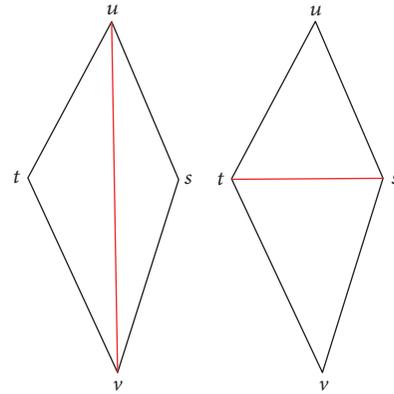


FIGURE 3: An example of performing edge flip.

Step 1 (edge flip). Edge flip is commonly used in mesh generation and mesh optimization for improving element qualities [14]. Edge flip is often preferred to other mesh modification methods (e.g., edge collapse and edge split) since it minimizes topological changes. It maintains the number of total vertices and edges in the mesh but is able to significantly improve mesh qualities by simply changing edge connectivities. Figure 3 shows an example of performing edge flip. Edge flip replaces the edge $\{u, v\}$ with $\{t, s\}$. For this example, two triangles ($\{t, s, u\}$ and $\{v, s, t\}$) become Delaunay triangles after performing edge flip.

A Delaunay flipping criterion is used to perform edge flip on a given edge in a tangled mesh. From [15], for each edge $\{u, v\}$ with opposite vertices $\{t, s\}$, $\{u, v\}$ is flipped if the Delaunay flipping criterion (i.e., $\angle utv + \angle usv > \pi$) is satisfied. As shown in our previous study [15], a greedy strategy is used to flip edges in decreasing order of opposite angle [15]. Edge flip is allowed to occur only once for a given edge. Also, edge flip is not performed for edges located on the boundary and edge flip that results in inverted elements [15].

Step 2 (mesh untangling). We employ the optimization-based mesh untangling method proposed by Knupp for its simplicity [4], but other mesh untangling methods [11, 12] can be used for eliminating inverted elements. Knupp's untangling beta quality metric is designed to give high

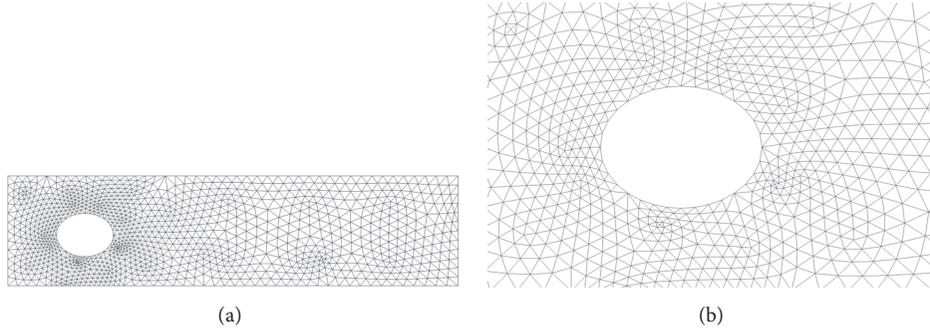


FIGURE 4: Moving circle domain: (a) initial mesh and (b) zoomed-in initial mesh on the cylinder domain.

penalty (cost) for inverted elements. The element quality for i th element in the mesh is defined as

$$q_i = \frac{1}{2} (|A - \beta| - (A - \beta)), \quad (1)$$

where A is a signed area of the element and β is a user-defined parameter, which is close to zero. In our numerical experiments, we set the β value as 0.01. Note that q_i is zero for valid (noninverted) elements. Then, the optimization problem is formulated by

$$\min \left\{ \sum_{i=1}^N q_i^2 \right\}, \quad (2)$$

where N is the number of elements. We use a nonlinear conjugate gradient method implemented in Mesquite [8] to solve (2). More information on the nonlinear conjugate gradient method for solving mesh optimization problems is provided in our previous paper [3].

Step 3 (mesh smoothing). After all inverted elements are eliminated, we perform optimization-based mesh smoothing for further improving the mesh quality. We employ an inverse mean ratio (IMR) quality metric in order to improve the element quality [4, 16]. The ideal element of the IMR quality metric is an equilateral triangle for isotropic PDEs. Let a , b , and c be the three vertices of a triangle. Next, define the incidence matrix A by $[b - a, c - a]$. Let W be the incidence matrix for an ideal element. For many applications, the ideal element is an equilateral triangle. In this case,

$$W = \begin{bmatrix} 1 & \frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} \end{bmatrix}. \quad (3)$$

The IMR quality metric is defined as

$$q_i = \frac{\|AW^{-1}\|_F^2}{2|\det(AW^{-1})|}, \quad (4)$$

where $\|\cdot\|_F$ is the Frobenius norm. The range of the IMR quality metric is between 1 and ∞ , where a value greater than one indicates that the element shape is different from

TABLE 1: Description of each step.

	Description of each step	Language/software
Step 1	Perform edge flip	C/C++
Step 2	Perform mesh untangling	Mesquite [8] (C++)
Step 3	Perform mesh smoothing	Mesquite [8] (C++)

the ideal element (e.g., equilateral triangle). The IMR quality metric has the value of 1 for the ideal element. Moreover, the IMR quality metric is invariant to translation, rotation, and uniform scaling of the element [16]. The overall mesh quality is denoted by $\sum_{i=1}^N q_i^2$. Then, the optimization problem is formulated by (2). Similar to the mesh untangling problem, we use the nonlinear conjugate gradient method implemented in Mesquite [8] to solve the optimization problem in (2).

5. Numerical Experiments

We perform numerical experiments to show the effectiveness of the proposed algorithm. We tested the ability of our proposed algorithm to generate high-quality meshes with no inverted elements on several extremely large boundary deformations. Finite element-based mesh warping (FEMWARP) mesh deformation algorithm is used to perform mesh deformation, which updates meshes on the deformed domain [7]. FEMWARP is used to perform mesh deformation since it is easy to implement and maintains good element quality after mesh deformation [13]. FEMWARP is further described in a previous report [7].

Table 1 summarizes each step with software/language used. GRX5 [17] is used to perform edge flip in Step 2. GRX 5 is used to flip interior edges that do not satisfy Delaunay criteria [15]. We repeatedly perform Steps 1 and 2 until all inverted elements are eliminated. The machine employed for this study is equipped with an AMD Opteron processor 6174 (2.2 GHz) and 6.5 GB of RAM.

5.1. Moving Circle Example. We first consider an example of moving circle. The initial mesh has 2,240 elements with no inverted elements as shown in Figure 4. We move the circle approximately 4 diameters to the right. The deformed mesh using FEMWARP has 53 inverted elements with many skinny

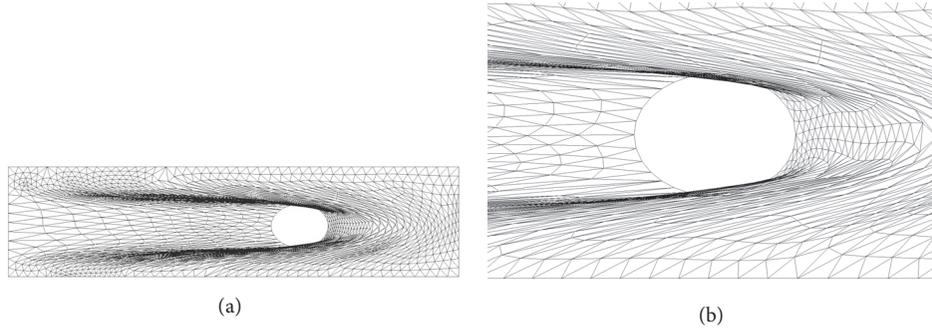


FIGURE 5: Moving circle domain: (a) deformed mesh and (b) zoomed-in deformed mesh on the cylinder domain.

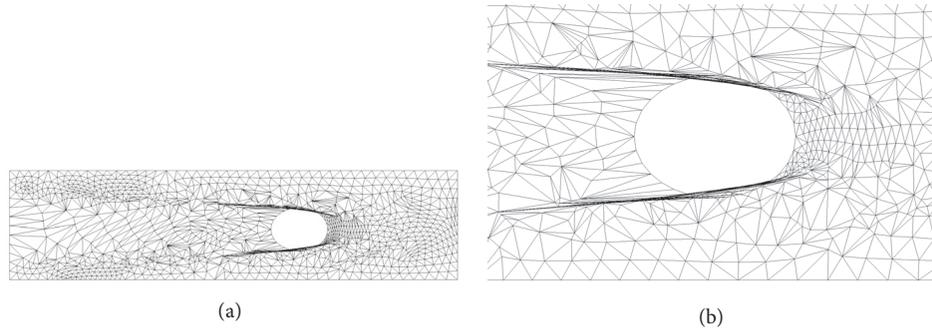


FIGURE 6: Moving circle domain: (a) output mesh after initial edge flip and (b) zoomed-in output mesh after initial edge flip on the cylinder domain.

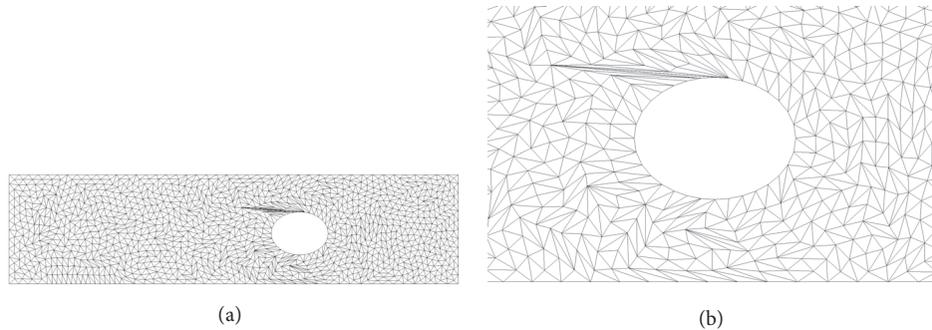


FIGURE 7: Moving circle domain: (a) untangled mesh and (b) zoomed-in untangled mesh on the cylinder domain.

triangles, as shown in Figure 5. Most inverted elements are produced around the circle due to the large deformation. The output mesh of initial edge flip is shown in Figure 6. We observe that the number of skinny triangles is significantly reduced after performing edge flip. As we expected, initial edge flip eliminates 14 inverted elements. Figure 7 shows untangled meshes after repeatedly performing Steps 1 and 2. As a result, all inverted elements are successfully eliminated. For this example, a total of three iterations of Steps 1 and 2 are required to eliminate all inverted elements. The untangled mesh has no inverted elements but has poor element qualities. We finally perform optimization-based mesh smoothing (Step 3) to improve element qualities. The final output mesh is shown in Figure 8.

Table 2 shows mesh quality statistics and the number of inverted elements of the initial mesh and output meshes. We observe that the overall mesh quality of the final mesh is comparable to the initial mesh considering the huge boundary deformation. In order to see the effectiveness of performing initial edge flip for tangled meshes, we compare the output mesh using the proposed method with the output mesh that only performs Steps 2 and 3 of the proposed method. The output mesh, which only performs Steps 2 and 3, fails to remove inverted elements and also results in the output mesh with poor element qualities.

5.2. *Moving Bar Example.* Next example is a moving bar with large inner boundary deformations. Initial undeformed

TABLE 2: Mesh quality statistics and the number of inverted elements on the moving circle domain. The inverse mean ratio quality metric is used to measure the mesh quality.

Mesh quality	Average	Maximum	Number of inverted elements
Initial mesh	1.038	1.944	0
Deformed mesh	23,681	$1e + 06$	53
Output mesh after initial edge flip	3.274	346.5	39
Untangled mesh	1.829	938.8	0
Final mesh	1.083	6.358	0
Output mesh with only Steps 2 and 3	17.54	1,228	54

TABLE 3: Mesh quality statistics and the number of inverted elements on the moving bar domain. The inverse mean ratio quality metric is used to measure the mesh quality.

Mesh quality	Average	Maximum	Number of inverted elements
Initial mesh	1.075	1.649	0
Deformed mesh	2.019	623.425	27
Output mesh after initial edge flip	1.283	71.158	26
Untangled mesh	1.191	3.987	0
Final mesh	1.061	2.075	0
Output mesh with only Steps 2 and 3	2.052	571.588	30

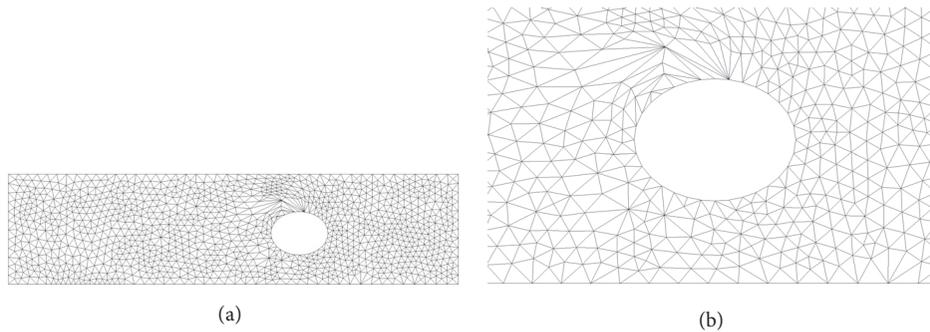


FIGURE 8: Moving circle domain: (a) final mesh and (b) zoomed-in final mesh on the cylinder domain.

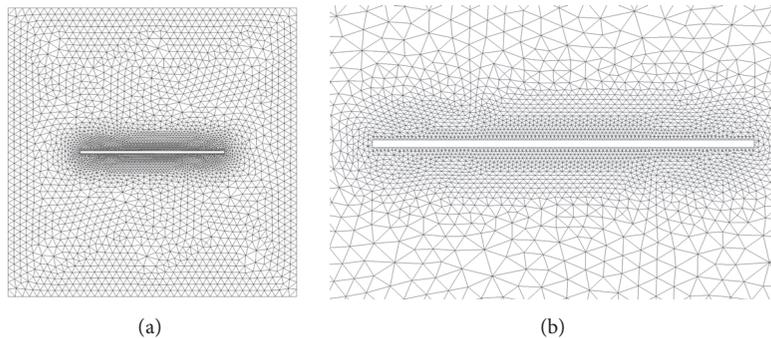


FIGURE 9: Moving bar domain: (a) initial mesh and (b) zoomed-in initial mesh on the bar domain.

mesh with good element quality is shown in Figure 9. The deformed mesh using FEMWARP is shown in Figure 10. For this example, FEMWARP results in 27 inverted elements after performing mesh deformation. Inverted elements are produced near inner boundary due to geometric constraints. The output mesh of initial edge flip is shown in Figure 11.

Edge flip successfully eliminates skinny triangles. Inverted elements are successfully eliminated after performing mesh untangling (see Figure 12). For this example, a total of three iterations of Steps 1 and 2 are required to eliminate all inverted elements. Figure 13 shows the final output mesh with good element qualities after mesh smoothing. Table 3 shows mesh

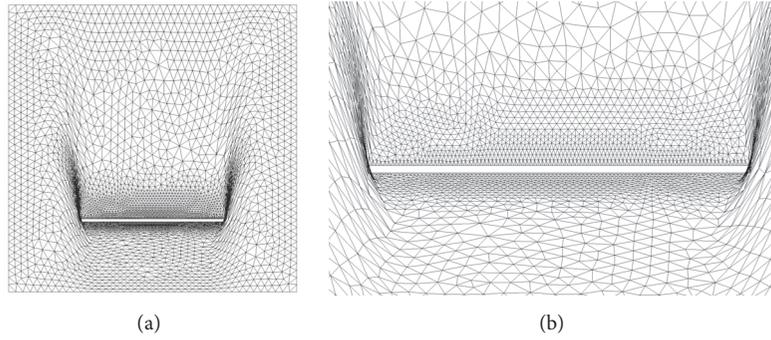


FIGURE 10: Moving bar domain: (a) deformed mesh and (b) zoomed-in deformed mesh on the bar domain.

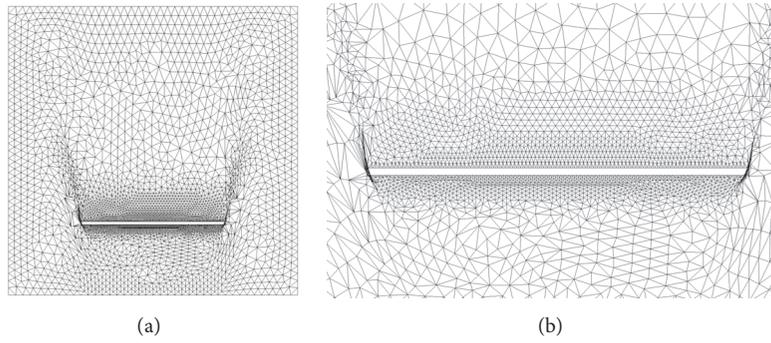


FIGURE 11: Moving bar domain: (a) output mesh after initial edge flip and (b) zoomed-in output mesh after initial edge flip on the bar domain.

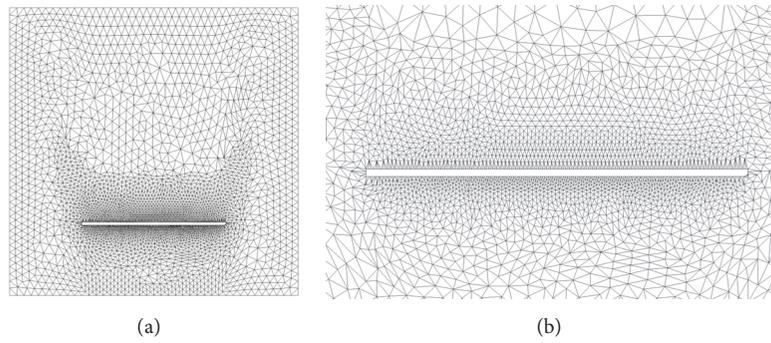


FIGURE 12: Moving bar domain: (a) untangled mesh and (b) zoomed-in untangled mesh on the bar domain.

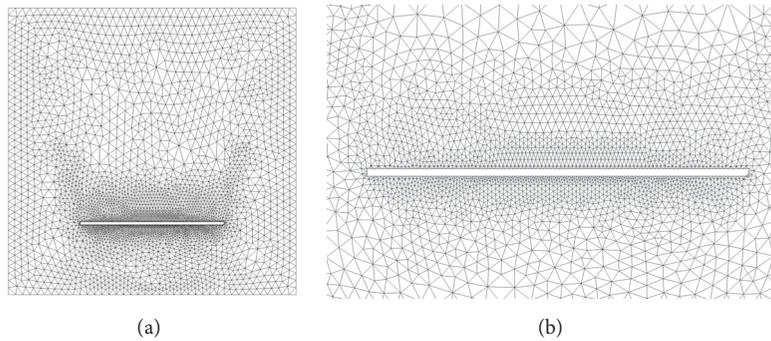


FIGURE 13: Moving bar domain: (a) final mesh and (b) zoomed-in final mesh on the bar domain.

TABLE 4: Mesh quality statistics and the number of inverted elements on the moving gate domain. The inverse mean ratio quality metric is used to measure the mesh quality.

Mesh quality	Average	Maximum	Number of inverted elements
Initial mesh	1.057	1.779	0
Deformed mesh	1.253	26.368	19
Output mesh after initial edge flip	1.145	16.378	20
Untangled mesh	1.136	2.540	0
Final mesh	1.065	1.979	0
Output mesh with only Steps 2 and 3	1.325	77.267	14

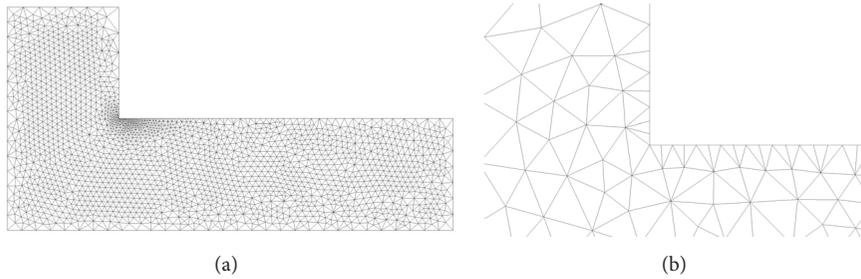


FIGURE 14: Moving gate domain: (a) initial mesh and (b) zoomed-in initial mesh on the gate domain.

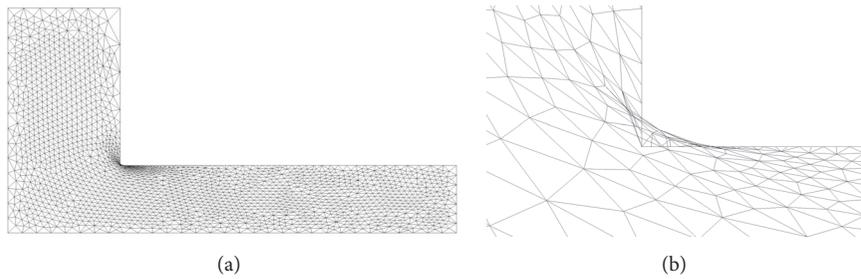


FIGURE 15: Moving gate domain: (a) deformed mesh and (b) zoomed-in deformed mesh on the gate domain.

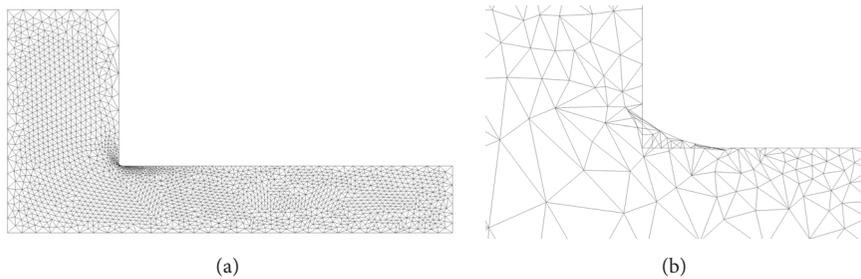


FIGURE 16: Moving gate domain: (a) output mesh after initial edge flip and (b) zoomed-in output mesh after initial edge flip on the gate domain.

quality statistics and the number of inverted elements of the initial mesh and output meshes after performing each step. The final output mesh quality is comparable to the initial mesh with no inverted elements. Similar to the previous example, first performing edge flip is effective in eliminating inverted elements.

5.3. Moving Gate Example. The final example is a moving gate example with outer boundary deformations. The initial mesh is shown in Figure 14. The deformed mesh using

FEMWARP (see Figure 15) has 19 inverted elements after mesh deformation. Elements with very large and small angles are produced after performing mesh deformation (see Figure 15). Edge flip significantly improves the overall element quality. The output mesh of edge flip (iteration 1) is shown in Figure 16. Remaining inverted elements are finally removed by repeatedly performing Steps 1 and 2. For this example, total 5 iterations of Steps 1 and 2 are required to eliminate all inverted elements (see Figure 17). Final output mesh after performing Step 3 is shown in Figure 18. Table 4 shows mesh

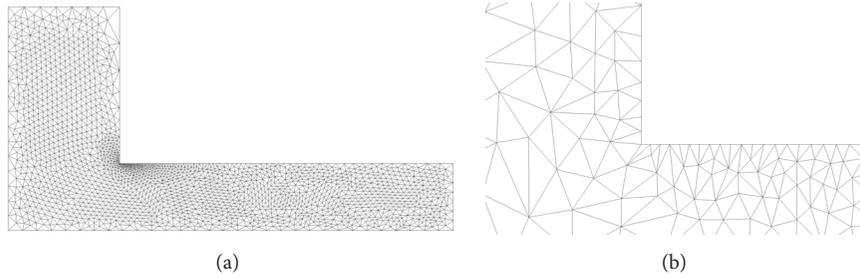


FIGURE 17: Moving gate domain: (a) untangled mesh and (b) zoomed-in untangled mesh on the gate domain.

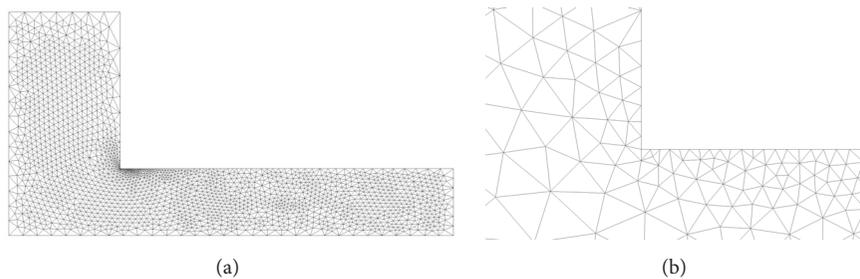


FIGURE 18: Moving gate domain: (a) final mesh and (b) zoomed-in final mesh on the gate domain.

quality statistics and the number of inverted elements of the initial mesh and output meshes of each step. The output using the proposed method results in output meshes with no inverted elements but the output mesh with only Steps 2 and 3 fails to untangle the mesh.

6. Conclusions

We propose an iterative mesh untangling algorithm using edge flip for highly tangled meshes. The proposed algorithm iteratively performs edge flip and optimization-based mesh untangling until all inverted elements are eliminated. Finally, mesh smoothing is performed for generating high-quality meshes. Numerical results show that the proposed algorithm is effective in eliminating inverted elements and maintains good element qualities, even for highly tangled meshes. Numerical results indicate that pure optimization-based mesh untangling methods by just relocating vertex positions have limitations for highly tangled meshes, since many interior vertices around inverted elements are highly constrained to move due to geometric constraints. For these cases, we propose initial edge flip prior to using mesh untangling methods, since edge flip yields large improvement in the element quality and significantly eliminates inverted elements. In the future, we plan to add other topological changes (e.g., edge collapse and edge split) for handling extremely tangled meshes.

Conflicts of Interest

The author declares that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

This work was supported by the Incheon National University Research Grant in 2014. The author wishes to thank David O. McLaurin for providing code developed in [15].

References

- [1] R. Garimella, J. Kim, and M. Berndt, "Polyhedral mesh generation and optimization for non-manifold domains," in *Proceedings of the 22nd International Meshing Roundtable*, pp. 239–250, Sandia National Laboratories, 2013.
- [2] J. Danczyk and K. Suresh, "Finite element analysis over tangled simplicial meshes: theory and implementation," *Finite Elements in Analysis and Design*, vol. 70/71, pp. 57–67, 2013.
- [3] J. Kim, T. Panitanarak, and S. M. Shontz, "A multiobjective mesh optimization framework for mesh quality improvement and mesh untangling," *International Journal for Numerical Methods in Engineering*, vol. 94, no. 1, pp. 20–42, 2013.
- [4] P. M. Knupp, "Hexahedral and tetrahedral mesh untangling," *Engineering with Computers*, vol. 17, no. 3, pp. 261–268, 2001.
- [5] J. W. Franks and P. M. Knupp, "A new strategy for untangling 2D meshes via node-movement," in *CSRI Summer Proceedings*, pp. 152–165, Sandia National Laboratories, 2010.
- [6] D. Benitez, E. Rodriguez, J. M. Escobar, and R. Montenegro, "Performance evaluation of a parallel algorithm for simultaneous untangling and smoothing of tetrahedral meshes," in *Proceedings of the 22nd International Meshing Roundtable*, 598, 579 pages, Sandia National Laboratories, 2013.
- [7] S. M. Shontz and S. A. Vavasis, "Analysis of and workarounds for element reversal for a finite element-based algorithm for warping triangular and tetrahedral meshes," *BIT Numerical Mathematics*, vol. 50, no. 4, pp. 863–884, 2010.

- [8] M. Brewer, L. Freitag Diachin, P. Knupp, T. Leurent, and D. Melander, "The mesquite mesh quality improvement toolkit," in *Proceedings of the 12th International Meshing Roundtable*, pp. 239–250, Sandia National Laboratories, 2003.
- [9] L. A. Freitag and P. Plassmann, "Local optimization-based simplicial mesh untangling and improvement," *International Journal for Numerical Methods in Engineering*, vol. 49, no. 1-2, pp. 109–125, 2000.
- [10] S. Bhowmick and S. M. Shontz, "Towards high-quality, untangled meshes via a force-directed graph embedding approach," in *Proceedings of the International Conference on Computational Science*, vol. 1 of no. 1, pp. 357–366, Procedia Computer Science, May 2010.
- [11] S. P. Sastry, S. M. Shontz, and S. A. Vavasis, "A log-barrier method for mesh quality improvement," in *Proceedings of the 20th International Meshing Roundtable (IMR '11)*, pp. 329–346, Sandia National Laboratories, October 2011.
- [12] J. Kim, B. J. Miller, and S. M. Shontz, "A hybrid mesh deformation algorithm using anisotropic PDEs and multiobjective mesh optimization," *Computers & Mathematics with Applications*, vol. 70, no. 8, pp. 1830–1851, 2015.
- [13] M. L. Staten, S. J. Owen, S. M. Shontz, A. G. Salinger, and T. S. Coffey, "A comparison of mesh morphing methods for 3D shape optimization," in *Proceedings of the 20th International Meshing Roundtable, IMR 2011*, pp. 293–312, October 2011.
- [14] L. A. Freitag and C. Ollivier-Gooch, "Tetrahedral mesh improvement using swapping and smoothing," *International Journal for Numerical Methods in Engineering*, vol. 40, no. 21, pp. 3979–4002, 1997.
- [15] J. Kim, D. McLaurin, and S. M. Shontz, "A 2D topology adaptive mesh deformation framework for mesh warping," in *Proceedings of the 4th TetrahedronWorkshop on GridGeneration for Numerical Computation*, May 2014.
- [16] T. Munson, "Mesh shape-quality optimization using the inverse mean-ratio metric," *Mathematical Programming*, vol. 110, no. 3, pp. 561–590, 2007.
- [17] D. McLaurin, D. Marcum, M. Remotigue, and E. Blades, "Algorithms and Methods for Discrete Surface Repair," ProQuest/UMI, Mississippi State University, 2010.



Hindawi

Submit your manuscripts at
<https://www.hindawi.com>

