Noname manuscript No.
(will be inserted by the editor)

A Log-Barrier Method for Mesh Quality Improvement and Untangling *

Shankar P. Sastry ·
Suzanne M. Shontz** ·
Stephen A. Vavasis

Received: date / Accepted: date

Abstract The presence of a few inverted or poor-quality mesh elements can negatively affect the stability, convergence and efficiency of a finite element solver and the accuracy of the associated partial differential equation solution. We propose a mesh quality improvement and untangling method that untangles a mesh with inverted elements and improves its quality. Worst element mesh quality improvement and untangling can be formulated as a nonsmooth unconstrained optimization problem, which can be reformulated as a smooth constrained optimization problem. Our technique solves the latter problem using a log-barrier interior point method and uses the gradient of the objective function to efficiently converge to a stationary point. The method uses a logarithmic barrier function and performs global mesh quality improvement. We have also developed a smooth quality metric that takes both signed area and the shape of an element into account. This quality metric assigns a negative value to an inverted element. It is used with our algorithm to untangle a mesh by improving the quality of an inverted element to a positive value. Our method usually yields better quality meshes than existing methods for improvement of the worst quality elements, such as the active set, pattern search, and multidirectional search mesh quality improvement methods. Our method is faster and more robust than existing methods for mesh untangling, such as the iterative stiffening method.

* A preliminary version of a portion of these results appeared in shortened form in the Proceedings of the 2011 International Meshing Roundtable

Shankar Prasad Sastry
The Pennsylvania State University
University Park, PA 16802
E-mail: sps210@cse.psu.edu

Suzanne M. Shontz
**Corresponding author
Mississippi State University
Mississippi State, MS 39762
E-mail: sshontz@math.msstate.edu

Stephen A. Vavasis
University of Waterloo
Waterloo, ON N2L 3G1
E-mail: vavasis@math.uwaterloo.ca

1 Introduction

This paper proposes a new method for displacing vertices of a finite element mesh in order to optimize the mesh quality. High-quality meshes are necessary for the stability [1] and efficiency of finite element (FE) solvers and for the accuracy [2–4] of the solution of the associated partial differential equations (PDEs). Inverted and poor quality elements affect the conditioning of the linear system that arises from the PDE and the mesh [1]. There are numerous geometric mesh quality improvement algorithms which are effective in improving the average mesh quality [5–9]. However, even a few poor quality elements can negatively affect the entire finite element analysis due to the resulting ill-conditioned matrices that hinder the efficiency and the accuracy of the finite element solvers [10]. Therefore, we focus on the development of an algorithm to improve the worst quality mesh element. Our algorithm solves a smooth constrained optimization problem to untangle a mesh with inverted elements and to improve the worst quality elements. It can be classified as an interior point method [11].

In many computational fluid dynamics (CFD) applications, for example, a consistent topology is desired for problems with evolving domains. For such applications, moving mesh methods are used to warp a mesh [12]. Note that local remeshing techniques may be used if consistent mesh topology is not required. In this paper, we assume that the mesh topology remains the same through the course of the simulation. In the event of mesh tangling after a moving mesh method is applied, mesh untangling may be used to obtain a valid mesh to continue the CFD simulation. Therefore, mesh untangling and mesh quality improvement algorithms are used to reorient inverted mesh elements after mesh deformation (e.g., mesh warping [13–15]) or to improve the quality of mesh elements when the initial qualities of the elements obtained from mesh generators are poor.

In order to efficiently solve isotropic, elliptic PDEs, the quality of a tetrahedral mesh is improved so that the shape of its elements are close to being regular [16]. In this paper, we use quality metrics whose optimal value for a quality of a tetrahedral element corresponds to a regular tetrahedral element. For other PDEs, posterior or prior error estimates may be used to improve the mesh quality [17]. Our technique can be used to improve the quality of the mesh as long as the quality metric is continuous for all possible vertex positions for a valid mesh.

Other algorithms have been developed for improvement of the quality of the worst element. Freitag and Plassmann developed an active set algorithm [18] for mesh quality improvement. However, their algorithm requires the objective function specifying the mesh quality metrics to be convex. Park and Shontz developed two derivative-free optimization algorithms, namely the pattern search and multidirectional search methods [19], for mesh quality improvement. These algorithms do not use the gradient to optimize the mesh quality; thus, we expect their rate of convergence to be slow. These algorithms are described in more detail in Section 2.

Also, inverted mesh elements result in termination of FE solvers. In order to untangle an inverted mesh, Freitag and Plassmann also developed a simplex-based mesh untangling algorithm [18], which maximizes the area (2D) or volume (3D) of a tetrahedral element. However, this technique may yield a mesh with poor quality. Escobar *et al*. developed an algorithm [20] that can simultaneously untangle and smooth tetrahedral meshes. Though this method yields a mesh with good average quality, it does not target the elements with the worst quality for improvement. Thus, a second optimization problem must be solved in order to improve the mesh quality.

Mesh untangling and worst element mesh quality improvement can be formulated as a nonsmooth unconstrained optimization problem. In Section 3, we describe the problem statement and show its reformulation into a constrained optimization problem. We solve this unconstrained optimization problem using an interior point method.

We develop a log-barrier interior point method [21] that seeks to improve the quality of its worst element. We suitably modify our algorithm and extend its application to untangle meshes with inverted elements. Our algorithm untangles the mesh first, and then improves the quality of the worst elements. As mesh quality improvement is a simpler problem, we first describe our algorithm for mesh quality improvement, and then describe the modifications necessary for our algorithm to be used for mesh untangling. Our method overcomes the disadvantages posed by the other algorithms presented above by employing a logarithmic barrier term, which is a function of the quality of the worst element. Though derived from classical optimization theory, the log-barrier method in our context has the following natural interpretation. On each iteration, the gradient of the log-barrier function points in a direction that is a weighted combination of the directions that improve each individual element. The weights are selected automatically in such a way that elements with the worst qualities have the highest weights (see (5) below). Therefore, the method globally updates vertex positions but concentrates on the improvement of the worst elements. Our interior point method solves the primal formulation of the constrained optimization problem and can be used on both convex and nonconvex objective functions. The algorithm for mesh quality improvement and untangling is presented in Section 4, and its characteristics are discussed in Section 5.

In order to improve the worst element mesh quality, any shape-based geometric mesh quality metric may be used with our method. However, for tangled meshes, the signed volume of the element is also important. We have developed a smooth quality metric that takes both signed volume and the shape of an element into factor. This quality metric assigns a negative value to an inverted element. It is used with our algorithm to untangle a mesh by improving the quality of an inverted element to a positive value. The suitable quality metrics for our algorithm for mesh quality improvement and for mesh untangling are presented in Section 6.

We perform numerical experiments to assess the efficiency of our algorithm by comparing its performance against that of existing algorithms. Mesh quality improvement techniques can be used until a desired mesh quality is reached (often specified by an analysis tool) or until the technique has converged [16]. In this paper, we execute our technique and the related techniques until convergence. Numerical experiments and their results for mesh quality improvement and mesh untangling are discussed in Sections 7 and 8, respectively. We give our concluding remarks in Section 9.

2 Related Work

2.1 Derivative-Free Algorithms for Mesh Quality Improvement

Park and Shontz developed pattern search (PS) and multidirectional search (MDS) mesh quality improvement algorithms [19] to improve the worst quality mesh element. Their derivative-free algorithms do not compute the gradient of the objective function, but instead use function evaluations to move the vertices. These algorithms also employ local mesh quality improvement, as pattern search techniques are not efficient on large problems. The following objective function is maximized in order to improve the worst element mesh quality:

$$f(x) = \min_{1 \le i \le m} q_i(x), \tag{1}$$

where m is the number of elements and $q_i(x)$ is the quality of i^{th} element.

Pattern Search (PS) Method. The PS method moves a mesh vertex in one of a pre-defined (usually orthogonal) set of directions. The direction and distance (i.e., the step length) by which each vertex is moved is determined by evaluating the objective function at the pattern points. If the vertex movement would result in a tangled mesh, the movement is backtracked by decreasing the step length along the search direction. This is known as a backtracking line search and is used to prevent the inversion of an element.

Multidirectional Search (MDS) Method. The MDS method uses search directions given by a simplex, i.e., a triangle (2D) or a tetrahedron (3D). The simplex is expanded, contracted, and/or reflected in order to determine the optimal position for a vertex. A backtracking line search is used to prevent an element from inverting, if needed, in a similar manner as for the PS method.

2.2 Active Set Method for Mesh Quality Improvement

Freitag and Plassmann developed an active set mesh quality improvement algorithm [18], which maximizes the quality of triangular or tetrahedral mesh elements. To guarantee convergence, the relevant objective function formed by quality metrics should possess convex level sets. Examples of such quality metrics include the minimum of the sine of the angles of the triangle in 2D and the aspect ratio quality metric in 3D for individual submeshes. This method employs a local mesh quality improvement technique, where individual submeshes are optimized by moving one vertex at a time.

For each submesh, the objective function is defined as the quality of the worst element. The objective function described above in (1) is maximized in order to improve quality of the worst element in the mesh. They define the *active value* as the current value of the objective function due to the vertex placement, x, which is a vector containing the vertex positions. They define *active set*, denoted by A, as a set of those functions that result in the active value. The nonsmooth optimization problem of improving the quality of the worst element is solved using the steepest descent algorithm [11]. Here, the gradient and hence the descent direction is obtained by considering all possible convex combinations of active set gradients, $\nabla_x (q_i(x)) = g_i(x)$, and choosing the one that solves

$$\min_x \bar{g}^T \bar{g}, \text{where } \bar{g} = \sum_{i \in A} \beta_i g_i(x),$$

$$\text{s.t. } \sum_{i \in A} \beta_i = 1, \beta_i \geq 0.$$

A backtracking line search technique is used to determine the points at which the active set changes, and the vertex is moved to the point that results in the best mesh quality improvement.

3 Problem Formulation for Mesh Untangling and Quality Improvement

In this section, we mathematically formulate our mesh quality improvement and untangling problem. Our problem involves the description of the quality metrics used to measure the mesh quality and the objective function that is optimized in order to untangle meshes and improve its quality. We will describe the quality metrics we use for mesh quality improvement and untangling in Section 6.

Objective Function for Mesh Quality Improvement and Untangling. The problem of improving the worst quality element (inverted or otherwise) can be expressed as

$$\min_x \left(\max_{i \in [1,m]} q_i(x) \right).$$

In this paper, the aspect ratio quality metric and the nonsmooth aspect ratio quality metric is used (described in Section 6). Their range is from 1 to ∞, where 1 is the quality of an ideal element, and ∞ is the quality of a degenerate element. Note that this formulation assumes that the quality of an element should be minimized in order to obtain the ideal element. However, our algorithm is designed for maximization of a quality metric. Thus, the objective function is modified by taking the reciprocal of the quality metric as follows:

$$\max_x \left(\min_{i \in [1,m]} \frac{1}{q_i(x)} \right).$$

This can be reformulated as a constrained optimization problem

$$\max_x \; t \text{ subject to } t \leq \frac{1}{q_i(x)}, \forall i \in [1,m]. \tag{2}$$

Note that all of these formulations improve the quality of the worst element, as minimizing a function is equivalent to maximizing its reciprocal. Any quality metric that defines a maximum value for an ideal element, zero for a degenerate element, and a negative value for an inverted element may be used with our method. Such a quality metric may also be obtained by taking the reciprocal, scaling, and/or translating other quality metrics.

4 A Log-Barrier Method for Mesh Quality Improvement and Untangling

In this section, we describe our log-barrier method for mesh quality improvement and untangling. We develop our algorithm based on interior point methods [11], which can be used to solve constrained optimization problems. Our method uses a logarithmic barrier term, which emphasizes the improvement of inverted and poor quality elements, and solves the constrained optimization problem using the gradient of the objective function.

4.1 Interior Point Methods

Interior point methods are a class of methods used to solve constrained optimization problems [11]. For a constrained optimization problem, the objective function, $f(x)$, is maximized while respecting k constraints, $c_i(x) \leq 0$, $\forall i \in [1,k]$. When interior point methods are employed, the constraint is added as a logarithmic barrier term to the objective function. The new objective function, $F(x,\mu)$, is given by:

$$F(x,\mu) = f(x) + \mu \sum_{i=1}^{k} \log\left(-c_i(x)\right),$$

where $\mu > 0$. The modified objective function is iteratively maximized using an unconstrained optimization algorithm. On every iteration, μ is reduced so that the barrier term is eventually negligible, and the original objective function, $f(x)$, is maximized subject to the constraint. Pseudocode for a typical interior point method is presented in Algorithm 1.

Algorithm 1 Interior Point Method

Input: $f(x)$, $c_i(x)$, tol.
Output: x that approximately solves $\max_x f(x)$ such that $c_i(x) \leq 0$, $\forall i \in [1,k]$.
Initialize $\mu_j > 0$ and x_0 such that $c_i(x_0) < 0$, $\forall i \in [1,k]$.
while $\mu_j \geq$ tol **do**
  Maximize $f(x) + \mu_j \sum_{i=1}^{k} \log\left(-c_i(x)\right)$ using any gradient-based optimization algorithm.
  Decrease μ_j towards 0.
end while

4.2 The Log-Barrier Method for Mesh Quality Improvement and Untangling

We seek to untangle a mesh with inverted elements and, if no element is inverted, improve the element with the worst quality. As in (2), a quantity t is defined such that

$$t \leq \min_i \frac{1}{q_i(x)},$$

where $q_i(x)$ is the quality of the i^{th} element. The x that leads to the maximal value of t corresponds to the mesh whose worst element has the best possible quality. The two goals of untangling and improving are unified if the quality metric rates inverted elements with the lowest value. In what follows, $q_i(x)$ will be negative if element i is inverted when the nodal positions are given by x.

The log-barrier method replaces the constraint $t \leq q_i(x)$ with a barrier term

$$\log\left(\frac{1}{q_i(x)} - t\right)$$

in the objective function. In other words, we replace the constrained problem (2) by the unconstrained problem of maximizing

$$t + \mu_k \sum_{i=1}^{m} \log\left(\frac{1}{q_i(x)} - t\right). \tag{3}$$

The barrier term automatically enforces the constraint since the logarithm function tends to $-\infty$ at the boundary of the feasible region. In fact, it enforces strict inequality $t < q_i(x)$. After every iteration, μ is brought closer to 0 by multiplying it by some factor γ, where $0 < \gamma < 1$. Note that other techniques may be used to reduce μ towards 0 after every iteration. We modify t such that $\frac{d}{dt}(F(x,t,\mu)) \approx 0$. For a fixed μ and x, the objective function is strictly concave in t. Therefore, setting its derivative to 0 corresponds to globally maximizing the objective with respect to t. The log-barrier method for mesh quality improvement and mesh untangling is shown in Algorithm 2.

Algorithm 2 Interior Point Method for Mesh Quality Improvement

Initialize μ_k and $t < 1/q_i(x)$ for all $i \in [1,m]$ where $q_i(\cdot)$ is the quality metric function.
while not converged **do**
  Maximize $F(x,t,\mu_j) = t + \mu_j \sum_{i=1}^{m} \log\left(\frac{1}{q_i(x)} - t\right)$, where t and μ are held constant, using the nonlinear conjugate gradient method.
  Decrease μ_j towards 0 by setting $\mu_{j+1} = \gamma\mu_j$.
  Update t to a new value such that $\frac{d}{dt}(F(x,t,\mu_{j+1})) \approx 0$.
end while

5 Characteristics of the Log-Barrier Method for Mesh Quality Improvement and Untangling

In this section, we show that the set of first-order necessary conditions, i.e., the Karush-Kuhn-Tucker (KKT) conditions [11], are satisfied for a solution of our constrained optimization problem. This fact is well known in the optimization literature; we include it here for the sake of completeness and also to provide more detail about the properties of the objective function and the algorithm. In addition, we examine the monotonicity of our algorithm.

5.1 Satisfaction of the KKT Conditions

Consider a constrained optimization problem of maximizing $f(x)$, while respecting the k constraints $c_i(x) \leq 0$, $\forall i \in [1,k]$. The Lagrangian is given by

$$L(x,\lambda) = f(x) + \lambda c(x),$$

where λ is a vector of Langrange multipliers. The active set is given by

$$A(x) = \{i | c_i(x) = 0\}.$$

For a given point x, linear independence constraint qualification (LICQ) holds if the active set gradients $\{\nabla c_i(x) | i \in A(x)\}$ are linearly independent.

Suppose that x^* is a solution to our constrained optimization problem and that LICQ holds at x^*. Then there is a Lagrange multiplier vector λ^* such that the following conditions (i.e., KKT conditions) are satisfied at (x^*, λ^*):

- stationarity condition:
$$\nabla_x L(x^*, \lambda^*) = 0$$
- primal feasibility:
$$c_i(x^*) \leq 0,\ \forall i \in [1,k]$$
- dual feasibility:
$$\lambda_i^* \geq 0,\ \forall i \in [1,k]$$
- complementarity condition:
$$\lambda_i^* c_i(x^*) = 0,\ \forall i \in [1,k].$$

Now, consider our constrained optimization formulation of the mesh quality improvement problem. For a solution, x^*, of a constrained optimization problem, the gradient of the Lagrangian vanishes at the solution, i.e., $\nabla_x L(x^*, t^*, \lambda^*) = 0$. For (2), the Lagrangian is given by

$$L(x,t,\lambda) = t + \sum_{i=1}^{m} \lambda_i \left(\frac{1}{q_i(x)} - t \right).$$

Hence, its gradient is given by

$$\nabla_x L(x,t,\lambda) = \sum_{i=1}^{m} \lambda_i \nabla_x \left(\frac{1}{q_i(x)} \right). \tag{4}$$

The nonlinear conjugate gradient (CG) step in the log barrier method computes x such that the gradient of the objective function given by (3), i.e., $\nabla_x F(x,t,\mu_k)$, vanishes. For this choice of F,

$$\begin{aligned} \nabla_x F(x,t,\mu_k) &= \mu_k \nabla_x \sum_{i=1}^{m} \log \left(\frac{1}{q_i(x)} - t \right) \\ &= \mu_k \sum_{i=1}^{m} \frac{1}{\left(\frac{1}{q_i(x)} - t \right)} \nabla_x \left(\frac{1}{q_i(x)} \right). \end{aligned} \tag{5}$$

From (4) and (5), we see that, if λ_i is defined by

$$\lambda_i^* = \frac{\mu_k^*}{\frac{1}{q_i(x^*)} - t^*}, \tag{6}$$

then the solution obtained by our method satisfies the stationarity requirement of the KKT conditions. The stationarity conditions are satisfied, as

$$\nabla_x L(x^*, t^*, \lambda^*) = \sum_{i=1}^{m} \lambda_i^* \nabla_x \left(\frac{1}{q_i(x^*)} \right) = 0.$$

Primal feasibility is also satisfied, since

$$\frac{1}{q_i(x^*)} - t^* \geq 0.$$

Dual feasibility is satisfied if

$$\lambda_i^* \geq 0.$$

From (6), and since $\mu_k > 0$ and $\frac{1}{q_i(x^*)} - t^* > 0$ at the solution, we have

$$\lambda_i^* \geq 0.$$

The complementarity condition requires that

$$\lambda_i^* \left(\frac{1}{q_i(x^*)} - t^* \right) = 0.$$

Substituting for λ_i^*, we see that

$$\lambda_i^* \left(\frac{1}{q_i(x^*)} - t^* \right) = \mu_k.$$

The log-barrier method drives μ_k to 0 as $k \to \infty$. Thus, the complementarity condition is also satisfied. Therefore, our log-barrier method converges to stationary points. Our implementation explicitly checks that the line search exploration moves the vertices in an ascent direction.

5.2 Monotonicity

In our algorithm, the optimization method maximizes the objective function given by (3),

$$F(x,t,\mu_k) = t + \mu_k \sum_{i=1}^{m} \log \left(\frac{1}{q_i(x)} - t \right),$$

on every iteration. Because t and $\mu_k > 0$ are constants for a given iteration, the maximization of the objective function is equivalent to maximization of the sum of the logarithmic terms. This is equivalent to maximizing the product of the terms (without taking their logarithms).

For simplicity of the analysis, let us now examine the monotonicity of our algorithm when employed on a patch having only two elements. If we plot the qualities of the two elements on the X and Y axes, we obtain hyperbolic contours representing the objective function as shown in Fig. 1. In Fig. 1, P represents a patch with nearly equal qualities of the elements, and Q represents a patch with unequal element qualities. The symbols a, b, c, and d represent the paths the patches can take in order to maximize the objective function. Ideally, P should take path b, and Q should take path d, so that the qualities of both the elements improve. In many cases, this is not possible, as improving the quality of one of the elements decreases the quality of the other. When the qualities of the two elements are nearly equal, if P takes path a, the quality of the worst element decreases. Thus, we see

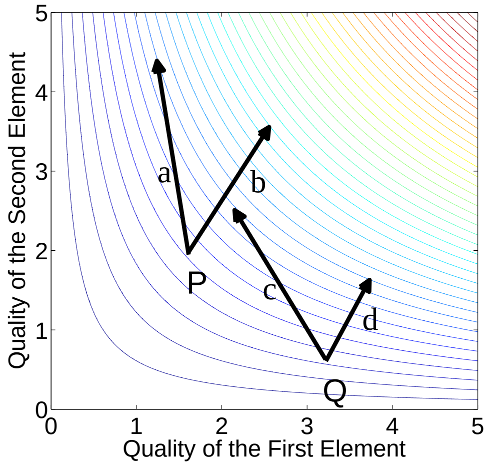
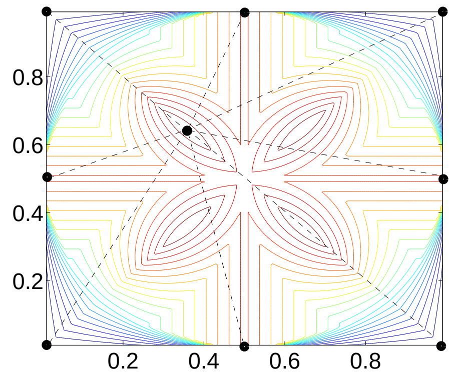


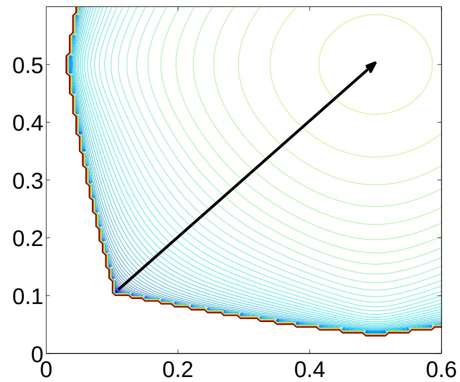
Fig. 1 Illustration of possible nonmonotonicity in the convergence of our algorithm. The X and Y axes represent the qualities of two elements in a patch, and the hyperbolic contours represent the sum of the two qualities on a logarithmic scale (which is maximized in an iteration). P and Q are possible locations of qualities of the patch. The symbols a, b, c, and d are the possible paths our algorithm can take. Although the objective function is maximized, notice that the quality of the worst element may not improve in all cases.

that our algorithm may not monotonically increase the quality of the worst element in the mesh. For the unequal case, path c also improves the quality of the worst element.

Fig. 2 shows how the nonsmooth objective function for maximizing the worst quality element in Fig. 2(a) is converted to a smooth objective function in Fig. 2(b). In Fig. 2(a), the nonsmooth aspect ratio is plotted for a patch with a free vertex in the square formed by the diagonal from $(0,0)$ to $(1,1)$. Other vertices are on the perimeter of the square at $(0,0)$, $(0,0.5)$, $(0,1)$, $(0.5,1)$, $(1,1)$, $(1,0.5)$, $(1,0)$, and $(0.5,0)$. Note that the contours are nonsmooth when plotting the worst quality element in the patch. For illustration purposes, we chose the point $(0.1,0.1)$, where the function is nonsmooth and set t in the log-barrier objective function as some quantity less than the worst element quality in the patch with the free vertex at $(0.1,0.1)$. The smooth contours of the objective function are plotted in Fig. 2(b). Our algorithm moves the free vertex at $(0.1,0.1)$ to a point close to $(0.5,0.5)$.



(a) Contours of the mesh quality



(b) Contours of the objective function

Fig. 2 Contours of the worst element quality and log-barrier objective function for a patch with eight vertices on the perimeter. (a) Contours of the nonsmooth aspect ratio quality metric. They are nonsmooth at points where two worst quality elements are present. The dotted lines indicate a patch with one possible vertex position. (b) Contours of the log-barrier objective function with the free vertex at $(0.1,0.1)$. Notice how smooth the contours are.

6 Quality Metrics

In this section, we define quality metrics that may be used with our method to improve the quality of mesh elements. However, the naïve use of the same metrics do not work for mesh untangling. Thus, we suitably modify the metrics to successfully untangle meshes.

6.1 Quality Metrics for Mesh Quality Improvement

We use two quality metrics, i.e., one smooth and one nonsmooth, to demonstrate the effectiveness of our algorithm.

Aspect Ratio Quality Metric: We define the quality of tetrahedral mesh element i as

$$q_i(x) = \left(\frac{l_1^2 + l_2^2 + \cdots + l_6^2}{6}\right)^{\frac{3}{2}} \Big/ \left(\mathrm{vol} \times \frac{12}{\sqrt{2}}\right),$$

where x are the vertex positions, vol is the unsigned volume of the i^{th} tetrahedron, and l_j is the length of side j of the tetrahedron [22]. The range of this quality metric is from 1 to ∞, where 1 is the quality of an regular tetrahedron. The quality tends to infinity as the tetrahedron becomes degenerate. Note the reciprocal of the metric is used in our implementation as described in Section 4.4.

Nonsmooth Aspect Ratio Quality Metric: We define the quality of the i^{th} tetrahedral mesh element as

$$q_i(x) = \frac{\sqrt{2}}{12} \frac{l_{\max}^3}{\text{vol}},$$

where $l_{\max}$ is the length of the longest edge of the tetrahedron. The range of this quality metric is from 1 to ∞, where 1 is the quality of an equilateral tetrahedron. The quality tends to infinity as the tetrahedron becomes degenerate. Note the reciprocal of the metric is used in our implementation as described in Section 4.4.

Log Barrier Term for Nonsmooth Quality Metrics: As our log-barrier method uses the gradient of the objective function, the objective function needs to be differentiable. However, the nonsmooth aspect ratio metric, which is used in defining the objective function, is not differentiable. Therefore, our method handles this metric by using each of the edges in the tetrahedron to compute six qualities for the tetrahedron and uses them as additive terms in the log barrier function. Since each individual term (for each edge in the tetrahedron) is smooth, the resulting log barrier function is also smooth. Note that the nonsmooth aspect ratio quality metric is defined using only the longest edge of a tetrahedron.

6.2 Quality Metrics for Mesh Untangling

Below, we describe how our log-barrier method can also be used to untangle meshes. Inverted elements are assigned a negative value, and by improving the quality of the inverted element to a positive value, a mesh can be untangled. We first describe an unsuccessful attempt to untangle a mesh by maximizing the minimum signed mesh quality and then describe a successful attempt in which the minimum signed volume of the mesh elements was maximized. We improve upon this successful attempt by combining the two techniques and compare the resulting algorithm against the element quality measuring technique of Escobar *et al.* [20].

Maximizing the Minimum Signed Mesh Quality: The reciprocal of the smooth aspect ratio quality metric that is used to determine the element quality is a function of the signed volume of the element. It is negative when the element is inverted. By maximizing the element quality to a positive value, it is possible to untangle the mesh. In Fig. 3, the contours of the element quality (for a triangular element) associated with a free vertex when the other two vertices are held stationary are plotted. It is clear that a gradient-based optimization algorithm will be able to improve an inverted element only when the gradient points towards the line connecting the stationary vertices. If the free vertex is sufficiently far away from this line, then the gradient points away from the line; hence, the element will remain inverted. We tested our hypothesis by perturbing a mesh vertex by some distance. When the distance of the perturbation was increased beyond a certain value, our method was no longer able to untangle the mesh.

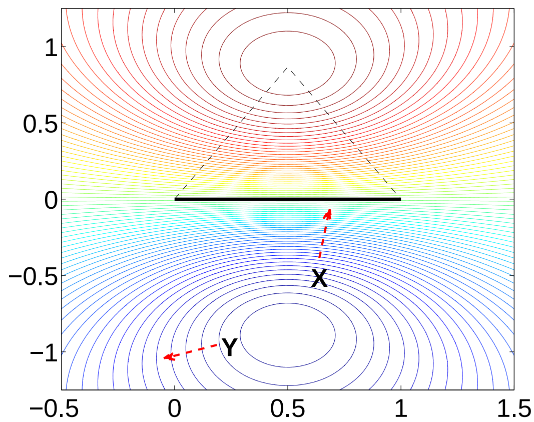


Fig. 3 Contours of the reciprocal of the smooth aspect ratio quality of a triangular element as a function of a free vertex. The other two vertices are held stationary at (0,0) and (1,0). The solid line indicates the line segment joining the stationary vertices. The dotted lines indicate the other two sides of the triangular element when the free vertex is at an optimal position. The triangle is inverted when the free vertex is below the line joining the stationary vertices. When the free vertex is at X, the gradient points towards the line. However, when the free vertex is at Y, the gradient points away from the line.

Maximizing the Minimum Volume: We replaced the reciprocal of the smooth aspect ratio quality metric with the signed volume of the element. The contours of the signed volume (for a triangular element) are shown in Fig. 4. Limited success was achieved with this modification. The maximization of the minimum volume moves a vertex to a location that may yield suboptimal element quality. This results in constrained movement of the vertex that is responsible for element inversion because other vertices in the neighborhood are not in an optimal location.

Maximizing the Hybrid Mesh Quality: We found that, by combining the signed element volume and the reciprocal of the aspect ratio quality metric, simultaneous mesh quality improvement and untangling may be achieved. We define our hybrid mesh quality as follows: if the element is inverted, the signed element volume is defined to be its quality, and if the element is not inverted, the reciprocal of the smooth aspect ratio is defined to be its quality. Note that this hybrid mesh quality metric is continuous, as is shown

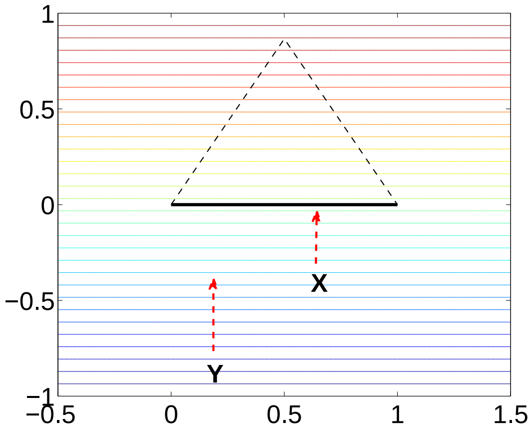


Fig. 4 Contours of the signed volume of a triangular element as a function of a free vertex. The other two vertices are held stationary at (0,0) and (1,0). The solid line indicates the line segment joining the stationary vertices. The dotted lines indicate the other two sides of the triangular element when the free vertex is at an optimal position. The triangle is inverted when the free vertex is below the line joining the stationary vertices. When the free vertex is at X or Y, the gradient points towards this line.

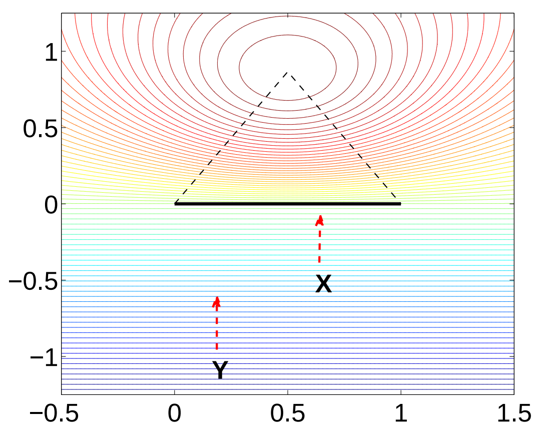


Fig. 5 Contours of the hybrid quality of a triangular element as a function of a free vertex. The other two vertices are held stationary at (0,0) and (1,0). The solid line indicates the line segment joining the stationary vertices. The dotted lines indicate the other two sides of the triangular element when the free vertex is at an optimal position. The triangle is inverted when it is below the line joining the stationary vertices. When the free vertex is at X or Y, the gradient points towards the line. Note that the metric is continuous (but not smooth) as the volume and the quality vanish for a degenerate element.

in Fig. 5, but is not smooth. Since we use a line search technique to move the vertices, and since the direction of the gradients of the functions defining the signed element volume and the reciprocal of the smooth aspect ratio quality metric are the same for a free vertex forming a degenerate element, it is possible to untangle the mesh and to improve the quality of the mesh elements. Our numerical experiments using the hybrid mesh quality metric show successful mesh quality improvement and untangling.

Hybrid Mesh Quality with Sigmoid Functions: As differentiable objective functions are desirable for convergence properties of gradient-based optimization algorithms, we proceed by constructing a differentiable function that is similar to the objective function defined by the hybrid mesh quality metric. The hybrid mesh quality can be formulated as

$$Q_i = w_{\mathrm{vol}} \mathrm{vol}_i(x) + w_{\mathrm{ar}} q_{\mathrm{ar}}^{-1}(x),$$

where vol_i is the signed volume of element i, w_{vol} is its weight, $q_{\mathrm{ar}}(x)$ is the smooth aspect ratio quality of the element, and w_{ar} is its weight. For the hybrid mesh quality metric, $[w_{\mathrm{vol}}, w_{\mathrm{ar}}] = [1, 0]$ for an inverted element, and $[w_{\mathrm{vol}}, w_{\mathrm{ar}}] = [0, 1]$ for an element that is not inverted. In order to make the function differentiable, the weights can be modified using sigmoid functions. The sigmoid functions defining the weights are given by

$$w_{\mathrm{vol}} = \frac{1}{1+e^{\alpha \cdot \mathrm{vol}_i(x)}} \text{ and } w_{\mathrm{ar}} = \frac{1}{1+e^{-\beta \cdot q_{\mathrm{ar}}^{-1}(x)}},$$

where α and β are scaling factors. In order to scale the gradient of the hybrid quality of an inverted element at its vertex positions, we have scaled the signed volume component of the hybrid quality metric. This ensures that the magnitude of the gradient of the hybrid quality of an inverted element at its vertex locations is scaled relative to that of the other elements. This is important as a *global* implementation of mesh optimization is being used. Thus, the new objective function is given by:

$$\max_x \left(\min_{i \in [1,m]} \left(\left(\frac{1}{1+e^{\alpha \cdot \lambda \mathrm{vol}_i(x)}} \right) \lambda vol_i + \left(\frac{1}{1+e^{-\beta \cdot q_{\mathrm{ar}}^{-1}(x)}} \right) q_{\mathrm{ar}}^{-1}(x) \right) \right),$$

where λ is the volume scaling factor. Note that volume scaling constant may vary for various parts of the mesh depending on the local grading. As we have used a homogeneous mesh in our numerical experiments, the same scaling factor is used for all the elements. In our numerical experiments, we find that the new objective function is able to be used to perform simultaneous mesh untangling and quality improvement of the worst element. Note that the quality metric should be defined such that the quality of a degenerate element is 0, and the quality of an ideal element is 1. This can be achieved by scaling, taking the reciprocal, and/or translating any smooth quality metric.

Quality Metric with Mapped Volume: In [20], the authors have suggested a modification to the function defining any signed volume-based quality metric to enable simultaneous mesh untangling and quality improvement. The signed volume of a tetrahedral element is modified using the following mapping function:

$$h(\mathrm{vol}_i(x)) = \frac{1}{2} \left(\mathrm{vol}_i(x) + \sqrt{\mathrm{vol}_i^2(x) + 4\delta^2} \right),$$

where δ is some constant. The contours of the reciprocal of the mapped aspect ratio quality metric are shown is Fig. 6. Note that this mapping is possible only when the quality of an element is a function of the signed volume of the element. We maximize the mapped quality of the worst element using the log-barrier method. Note that the magnitude of the gradient of the modified quality metric reduces as we move away from the optimal vertex position. The parameter, δ, can be increased to scale the magnitude to a larger value, but as δ increases, the optimal vertex position for the free vertex moves from being an equilateral triangle to a point closer to the line joining fixed vertices. Note that the gradients of the mapped aspect ratio quality metric (and not its reciprocal) are scaled so that their magnitudes are larger at vertex positions of inverted elements. For the mapped aspect ratio quality metric, the element quality must be minimized for quality improvement. For this formulation, techniques to improve the average quality have been developed, but not for worst element mesh quality improvement. In this chapter, we consider the problem of improving the quality of the worst mesh element.

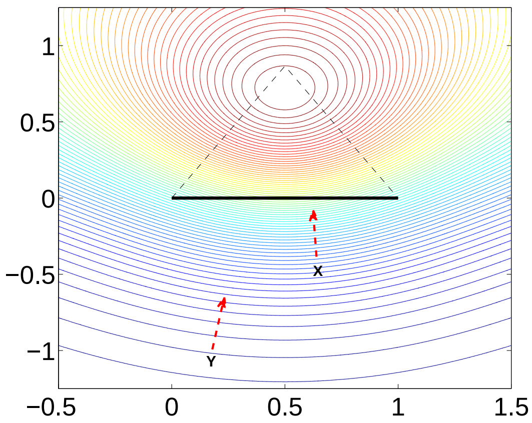


Fig. 6 Contours of the reciprocal of the aspect ratio quality with mapped volume of a triangular element as a function of a free vertex. The other two vertices are held stationary at (0,0) and (1,0). The solid line indicates the line segment joining the stationary vertices. The dotted lines indicate the other two sides of the triangular element when the free vertex is at an optimal position for the signed aspect ratio quality metric (not the modified quality metric with a mapped volume). The triangle is inverted when it is below the line joining the stationary vertices. When the free vertex is at X or Y, the gradient points towards the line. Note that the metric is continuous and smooth.

7 Numerical Experiments for Mesh Quality Improvement

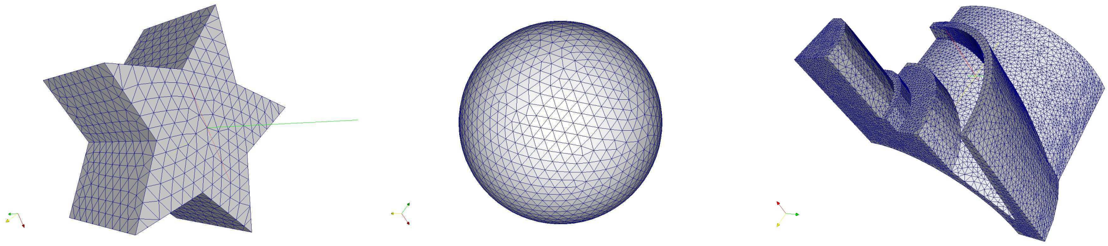
In this section, we describe the numerical experiments that we designed to evaluate the performance of our log-barrier method for mesh quality improvement. We implemented our algorithm, the PS, and the MDS methods in the Mesquite Mesh Quality Improvement Toolkit Version 2.0.0 [23]. The active set method was already implemented in Mesquite. For each of the methods, the movement of the surface vertices was enabled. For the star meshes, the gradient of the objective function for the boundary vertices was projected onto the respective planes. For the sphere meshes, if the boundary vertices moved away from the surface, they were snapped back onto the surface.

Star and sphere (Fig. 7) meshes of various sizes were constructed using CUBIT [24]. We considered these simple geometries because the movement of surface vertices for such planar and spherical surfaces is possible in Mesquite. We will be performing our experiments on more complex domains in future. Surface vertex movement can be designed in our method's implementation if the surface definitions from computer-aided design packages are imported into mesh quality improvement software. In order to test our algorithm on challenging meshes, 50% of the vertices in the original meshes were randomly perturbed. The following three experiments were performed.

7.1 Effect of Parameters on Algorithmic Performance

For our first experiment, the following set of parameters were modified to determine their effect on the performance of the mesh quality improvement. Three variants of the nonlinear conjugate gradient method, i.e., the Fletcher-Reeves, Polak-Ribière, and Hestenes-Stiefel variants were used to improve the mesh quality in the inner loop. Two, four, and eight CG iterations per outer iteration were used in each of the experiments. The parameter μ was reduced to 90%, 60%, and 30% of its value after every outer iteration. We used the largest star mesh with approximately 1.012 million elements. We maximized the reciprocal of the aspect ratio quality metric for each of the subexperiments. The subexperiments were carried out for all combinations of these parameters. The numerical experiments were run until the quality of the worst element did not improve for five successive iterations.

The detailed results of these experiments can be found in our previous paper [21]. We found that the Hestenes-Stiefel variant of the CG method with four inner iterations and 90% reduction in μ resulted in the most optimal performance. We have used these parameters in the rest of the paper for both mesh quality improvement and untangling unless specified otherwise.



(a) Star domain (b) Sphere domain (c) Stab3 domain

Fig. 7 Unperturbed coarse meshes on the three domains representative of actual meshes considered in this paper.

7.2 Scalability

For this experiment, the reciprocal of the aspect ratio metric was maximized to improve the quality of the perturbed meshes using all the methods described in Section 4.2. Two inner iterations were carried out for every outer iteration in the log-barrier, PS and MDS methods. The log-barrier method employed the Hestenes-Stiefel conjugate gradient algorithm [11] in the inner loop. After every outer iteration, μ was reduced to 90% of its value in the previous iteration. In order to accurately estimate the time per iteration, our implementation was executed for 50 iterations on each star mesh.

The results from the scalability experiment are shown in Fig. 8, where the time per outer iteration for each method scales linearly with the number of elements in the mesh, as the time required to compute the gradient and move the vertices is directly proportional to the mesh size. Table 1 provides the initial and the final worst element mesh quality for each of the meshes. We determined the order of convergence as a function of the problem size for our method. The order of convergence, α, is given by $T = k * m^{\alpha}$, where T is the time to convergence, m is the number of mesh elements, and $k > 0$ is a constant. In order to determine α, a least squares fit was computed by taking the logarithm of both sides. The value of α was found to be 0.9946.

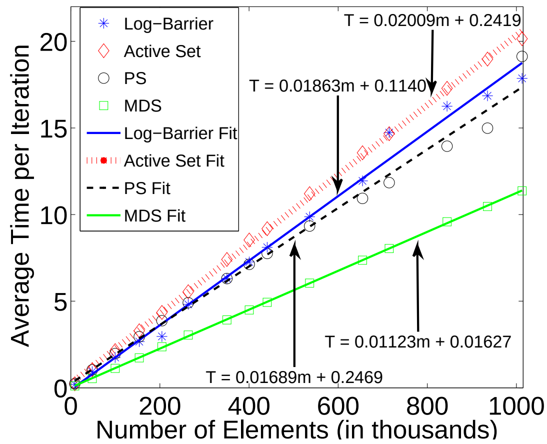


Fig. 8 Scalability experiment results for mesh quality improvement. The data and the linear regression fits are shown. In the equations, T refers to average time per iteration (in seconds), and m refers to number of elements (in thousands). The reciprocal of the aspect ratio quality metric was maximized. The average time per iteration for the first 50 outer iterations of all the methods were used to compute the least squares fit. For the log-barrier method, two Hestenes-Stiefel CG iterations were carried out per outer iteration.

7.3 Comparison with Existing Mesh Quality Improvement Methods for the Aspect Ratio Quality Metric

For this experiment, we used the largest meshes for each domain containing approximately 1.012 million and 1.014 million elements in the star and the sphere mesh, respectively. The reciprocal of the aspect ratio quality metric was maximized by the four algorithms previously discussed: the log-barrier, active set, PS, and MDS methods. Several experiments were conducted to find the values of the various parameters in each of the algorithms resulting in the best performance (measured as the worst element quality at the time by which our algorithm converged). The numerical experiments were carried out until the worst element quality remained the same for five iterations. We present results for the best performance for each of the algorithms.

The results for mesh quality improvement on the star mesh are shown in Fig. 9(a). Our method improved the mesh quality by the greatest amount when compared to the other methods. It was followed by MDS, PS, and then the active set method. A closer inspection of the plot reveals that, in the initial iterations, the active set method was the fastest method to significantly improve the worst quality. The method was followed by the MDS and PS methods. During the initial phase, our method was slower than the rest.

The slow convergence of the other three methods, despite their initial performance, may have been caused by the slow propagation of unequal patches due to their use of local mesh quality improvement. The active set method was able to quickly improve the worst quality by a significant amount within two iterations but became stagnant afterward. The active set method moves every vertex to the optimal location with respect to the patch.

The optimal vertex locations are approximately determined in each iteration for the PS and MDS methods. Thus, unequal patches are present throughout the mesh. This enables steady improvement of the worst element quality by MDS and PS. With MDS, we noticed a behavior similar to that of the active set method where the worst element quality was constant for seven iterations, and then the method converged to a mesh with a slightly better quality.

The results for the sphere mesh are shown in Fig. 9(b). As in the earlier case, our method was able to improve the mesh quality by the greatest amount. Here the PS method was very competitive and converged faster than our method, but converged to a lower optimal value. The MDS and active set methods also converged to a lower optimal value.

7.4 Comparison with Existing Mesh Quality Improvement Methods for the Nonsmooth Aspect Ratio Quality Metric

Through this experiment, we demonstrate that our algorithm is also efficient for mesh quality improvement when a nonsmooth or nonconvex quality metric is used to define the objective function. For this experiment, the objective function that was formed from the reciprocal of the nonsmooth aspect ratio quality metric was maximized by the log barrier, PS, and MDS methods. The active set method is designed to be used with a convex objective function and yields a tangled mesh when used with a nonconvex quality metric. Thus, we have shown only the results for the other three methods in Fig. 10(a). The numerical experiments were carried out until the quality of the worst element did not improve for five iterations. It can be clearly seen in Fig. 10(a) that our method yields a better quality mesh than the other methods. In Fig. 10(b), it can be seen that our log-barrier method converged to a mesh of somewhat lower quality than the PS method. We seek to determine the line search parameter values that can make our method converge to a mesh with better quality.

7.5 Domain with Sharp Features and High Curvature

In order to show that our technique also performs efficiently on domains with sharp features and high curvature, we ran our technique on a mesh on the stab3 domain (Fig. 7(c)). The mesh containing 124,907 vertices and 726,834 tetrahedral elements was generated using Tetgen [25]. The boundary vertices were held fixed in this experiment. The smooth aspect ratio quality metric was used to measure the quality of the mesh elements. The initial quality of the mesh was 0.05330. Our technique was able to improve the quality to 0.1085 in 423 seconds (13 iterations). The quality improvement is not as large as in the cases of the star and sphere mesh because this is a more challenging domain and because the boundary vertices were held fixed. Mesquite implements boundary vertex movement by moving the boundary vertices in the same way as the interior vertices and then snapping the vertices back to the boundary surface. This is not very robust for challenging geometries as snapping the vertices may lead to inversion of elements. We plan to incorporate robust boundary vertex movement in our future work.

8 Numerical Experiments for Mesh Untangling

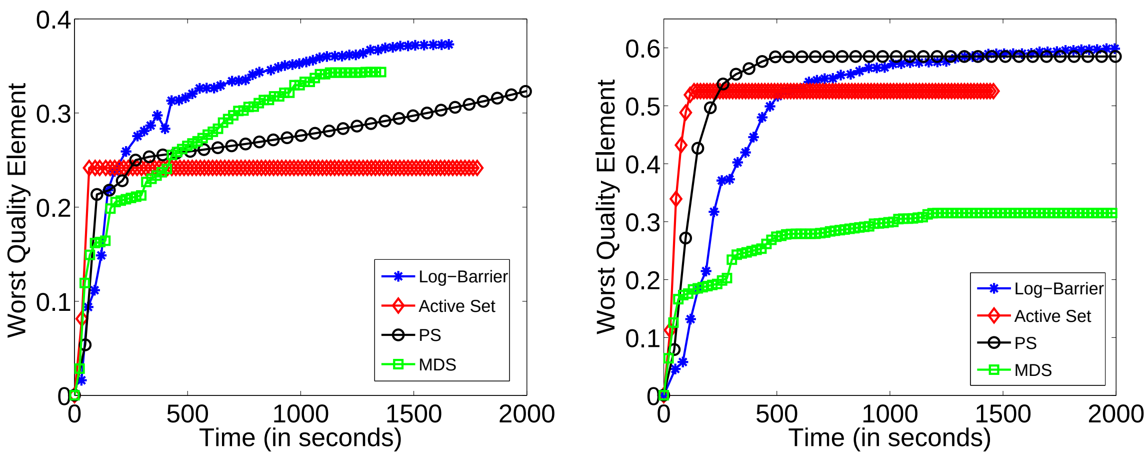
We have used experimental results from Section 4.7.1 for the input parameters to mesh untangling experiments. We have not performed scalability experiments for mesh untangling because the time taken to untangle a mesh is not only dependent on the size of the mesh, but also depends on the extent of the mesh tangling. Since there are multiple variables, we have only compared our algorithm with other existing techniques and their implementations.

In order to obtain test cases for our untangling algorithm, we perform the same numerical experiments as presented in [14] for each method. In the Untangling Before Newton (UBN) algorithm by Shontz and Vavasis [14], hyperelasticity equations for Dirichlet or Neumann boundary conditions are solved using Newton's method. Newton's method requires a starting point close to the real solution in order to successfully converge to a solution. In order to solve the hyperelasticity equations, the starting point should correspond to an untangled mesh. In their algorithm, the starting point is obtained by solving isotropic linear elasticity equations for the same boundary conditions. When the magnitudes of the vertex movements are large, obtaining vertex positions by solving the linear elasticity equations may yield tangled meshes. A procedure called iterative stiffening (IS) was developed to untangle the deformed mesh iteratively. The IS method selectively increases the stiffness of the tangled elements by a *stiffening constant*, and the linear elasticity equations are solved again to obtain the new positions of the vertices. We compare our log-barrier method against the IS method and examine their effectiveness in untangling meshes.

Since the log-barrier method is implemented in Mesquite, i.e., C++, and the IS method is implemented in MATLAB, we compare only the number of iterations required to successfully untangle the mess and the robustness of each technique. For the log-barrier method, we use four nonlinear conjugate gradient iterations for every outer iteration. In the implementation of the IS algorithm, the linear equation for the linear elasticity problem is solved using the linear conjugate gradient (CG) method with the ILU preconditioner. In order to compare the two techniques, we report the total

# Vertices	# Elements	Initial Worst Quality	Final Worst Quality
2,128	9,099	1.854e-03	5.299e-02
9,501	48,219	8.996e-04	5.617e-02
21,972	99,684	4.623e-04	2.044e-02
29,096	153,780	1.287e-05	3.490e-02
38,163	204,612	4.242e-05	4.014e-02
48,880	263,602	5.775e-04	4.598e-02
64,673	350,303	9.485e-06	3.330e-02
73,617	400,522	3.130e-05	3.458e-02
80,926	440,711	6.132e-05	2.984e-02
97,981	535,921	7.253e-05	2.358e-02
119,137	654,606	4.593e-05	4.036e-02
129,952	714,495	7.980e-06	2.334e-02
152,929	844,425	4.988e-06	3.151e-02
169,024	935,178	4.925e-06	1.433e-02
182,760	1,012,632	2.335e-04	3.050e-02

Table 1 Number of vertices and elements in the star meshes with their initial and final qualities after 50 outer iterations of quality improvement using the log-barrier method. The objective function that is formed from the reciprocal of the aspect ratio quality metric is maximized. The aspect ratio metric is a smooth, convex quality metric.

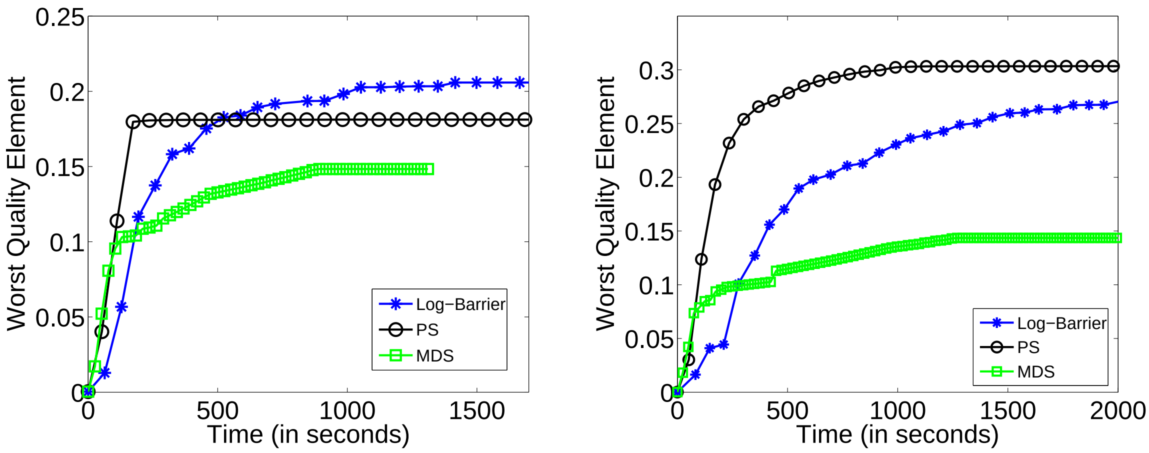


(a) Star mesh (b) Sphere mesh

Fig. 9 Results from the experiment that compares the mesh quality improvement algorithms. The aspect ratio quality metric was improved in the meshes.

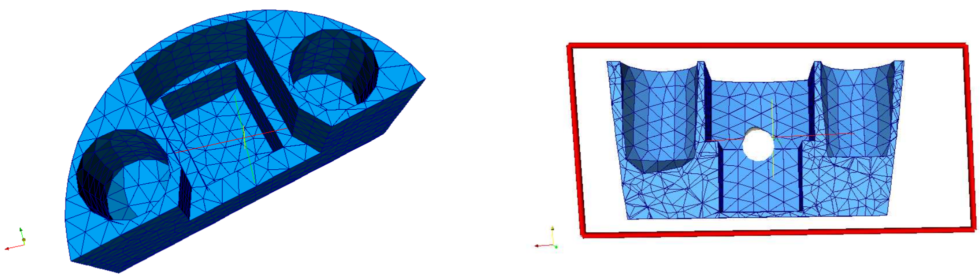
Mesh	Deformation	Mesh Untangling Methods					
		The Log Barrier Method with the Hybrid Quality Metric		The Log Barrier Method with Escobar *et. al.*'s Quality Metric [20]		Iterative Stiffening [14]	
		Volume Scaling	# of Iterations	The δ Parameter	# of Iterations	Stiffening Constant	# of Iterations
Foam5	0.50	10	1 (4)	0.002	1 (4)	1.5	1 (1)
	1.00	10	3 (12)	0.001	3 (12)	4.0	4 (4)
	1.25	10	8 (32)	0.001	7 (28)	4.0	4 (4)
	1.50	10	10 (40)	0.001	13 (52)	4.0	6 (6)
	1.60	10	16 (64)	0.001	18 (72)	2.0	7 (7)
	1.75	10	17 (68)	0.001	27 (108)	*	*
	1.90	10	29 (116)	0.0005	67 (268)	*	*
	2.00	10	44 (176)	*	*	*	*
	2.20	*	*	*	*	*	*

Table 2 Performance of various mesh untangling techniques. Input parameters to each of the algorithms are provided. The number of inner iterations are given within the parentheses. A '*' denotes that the method did not converge.



(a) Star mesh (b) Sphere mesh

Fig. 10 Results from the experiment that compares the mesh quality improvement algorithms. The nonsmooth aspect ratio quality metric was improved in the meshes. The active set method is designed to be used with a convex objective function, and it yields a tangled mesh when used with a nonconvex quality metric, and hence its performance is not shown here.



(a) Surface foam5 mesh (b) Cut-away view of the deformed mesh

Fig. 11 Foam5 mesh and its cut-away view showing the deformation. The foam5 mesh was provided to us by P. Knupp [26].

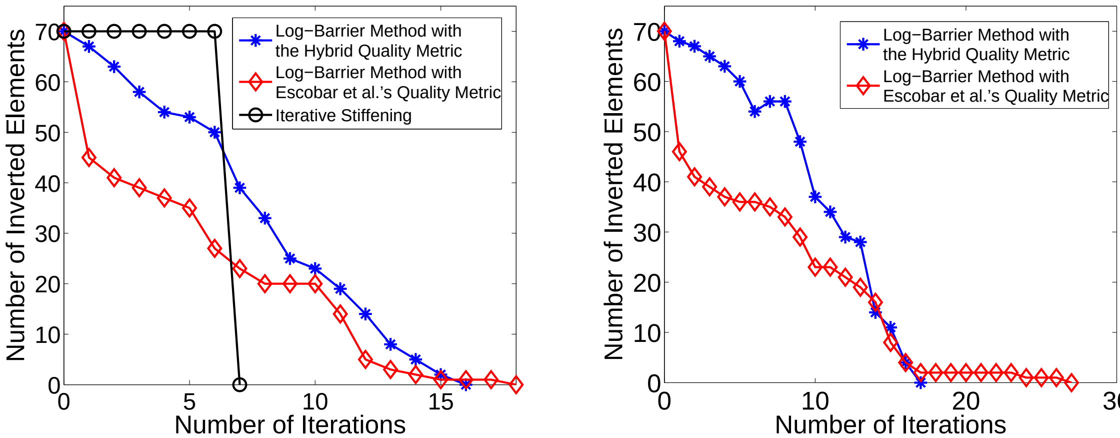
numbers of iterations required by the respective algorithms to converge to a solution.

In our numerical experiments, the foam5 mesh (Fig 11(a)), which was provided to us by P. Knupp [26], is deformed using Dirichlet boundary conditions. The mesh contains 1337 vertices and 4847 tetrahedral elements. Fig. 11(b) shows a cut-away view of the deformed mesh. The Dirichlet boundary conditions are provided for all the boundary vertices. The log-barrier method and the IS method untangle the mesh by moving only the interior vertices.

Table 2 shows the number of iterations required to untangle the mesh for various deformations (from 0.5 to 2.2 units). We ran multiple experiments to determine the parameters that resulted in the best overall performance. For the log-barrier method with the hybrid quality metric with sigmoid functions, the volume is scaled by a factor of 10. In the case of Escobar *et al.*'s quality metric, the parameter, δ, is set to 0.001 for all cases except for the deformation of 1.9 units, where δ was set to 0.0005. The factor by which the stiffness of the inverted elements is increased is given in the table. In Fig. 12, the number of inverted elements after each iteration is provided for the deformation of 1.6 and 1.75 units for all the three methods. The number of inverted elements gradually reduces for the log-barrier method with both the hybrid metric and Escobar *et al.*'s quality metric. For the IS method, the number of inverted elements falls from 70 to 0 in the last iteration in the case of 1.6 units of deformation. For the case of 1.75 units of deformation, the IS method does not yield an untangled mesh even after 100 iterations.

The deformations are restricted to 2.2 because the deformations greater than or equal to 2.2 result in a mesh in which the boundary surfaces themselves intersect. For such a deformation, it is not possible to untangle the mesh by moving only the interior vertices, since the boundary vertices should also be moved to untangle the mesh.

We see that the log-barrier method with the hybrid quality metric with sigmoid weight functions untangles the deformed meshes in the fewest iterations. The number of iterations vary when the volume is scaled by different factors. This additional parameter enabled quick mesh untangling as the gradient of the function defining the signed element



(a) Deformation of 1.6 units

(b) Deformation of 1.75 units

Fig. 12 The number of inverted elements after each iteration for all the three methods for the deformation of 1.6 and 1.75 units of the foam5 mesh. Note that only the number of iterations are provided. For 1.75 units of deformation, the iterative stiffening algorithm does not yield an untangled mesh even after 100 iterations. The time for untangling cannot be compared because the implementations of the respective algorithms are in different software environments.

volume can be appropriately scaled to move the vertices by the optimal magnitude. Although the log-barrier algorithm with Escobar *et al.*'s quality metric was initially competitive, its inability to scale the gradients resulted in a slower performance for highly deformed meshes. As shown in Fig.6, when the vertices are far away from their optimal locations, the gradients are poorly scaled. Hence, they are not moved to their optimal locations by a line search technique that moves all the vertices simultaneously (as in the case of global implementation of the mesh optimization algorithms). The IS method is not as successful as the log-barrier method. For large deformations, the IS method is unable to yield an untangled mesh.

9 Conclusions and Future Work

We have presented a log barrier interior point method for untangling and improving the worst quality elements in a finite element mesh. We reformulated the nonsmooth problem of maximizing the quality of the worst element as a smooth constrained optimization problem, which is solved using a log barrier interior point method. Our method uses a log barrier function whose gradient places a greater emphasis on poor quality elements in the mesh and performs global mesh quality improvement. We have shown that our algorithm converges to stationary points. We have also presented a suitable quality metric, which takes both the signed area and the shape of an element into account. It assigns a negative value as the quality of an inverted element, and the gradient of the function defining the metric at a vertex position points towards a direction necessary to reorient the element. This quality metric can be used with our algorithm to untangle meshes by improving the quality of an inverted element to a positive value.

Our method usually yields a better quality mesh than other existing worst quality mesh improvement methods and scales roughly linearly with the problem size. We have shown that our algorithm is more robust in untangling meshes with large numbers of inverted elements than existing algorithms.

We plan to use more complex domains and import surface geometry definitions from computer-aided design packages, and move surface vertices using optimization techniques for such domains. We have used the conjugate gradient method to perform mesh quality improvement in the inner loop. We plan to use Newton's method for the objective function maximization instead of the nonlinear CG method because it uses second-order information which may lead to faster convergence. The constrained optimization problem can also be solved using primal-dual Newton-based methods. An example of such a method includes the Mehrotra predictor-corrector method [27]. We plan to explore such methods to determine the most efficient solver for mesh quality improvement.

Acknowledgments

The work of the first two authors was supported in part by NSF grants CNS-0720749 and NSF CAREER Award OCI-1054459. The work of the third author was supported in part by a Discovery grant from NSERC (Canada) and a grant from the U.S. Air Force Office of Scientific Research. The authors wish to thank the three anonymous referees for their careful reading of the paper and for their helpful suggestions which strengthened it.

References

1. E. Fried, "Condition of finite element matrices generated from nonuniform meshes," *AIAA Journal*, vol. 10, pp. 219–221, 1972.
2. I. Babuska and M. Suri, "The p and h-p versions of the finite element method, basic principles, and properties," *SIAM Rev.*, vol. 35, pp. 579–632, 1994.
3. M. Berzins, "Solution-based mesh quality for triangular and tetrahedral meshes," in *Proc. of the* 6^{th} *International Meshing Roundtable*, pp. 427–436, 1997.
4. M. Berzins, "Mesh quality - Geometry, error estimates, or both?," in *Proc. of the* 7^{th} *International Meshing Roundtable*, pp. 229–237, 1998.
5. P. Knupp, "Matrix norms and the condition number: A general framework to improve mesh quality via node-movement," in *Proc. of the* 8^{th} *International Meshing Roundtable*, pp. 13–22, 1999.
6. P. Knupp and L. Freitag, "Tetrahedral mesh improvement via optimization of the element condition number," *Int. J. Numer. Meth. Eng.*, vol. 53, pp. 1377–1391, 2002.
7. N. Amenta, M. Bern, and D. Eppstein, "Optimal point placement for mesh smoothing," *In Proc. of the* 8^{th} *ACM-SIAM Symposium on Discrete Algorithms*, pp. 528–537, 1997.
8. T. Munson, "Mesh shape-quality optimization using the inverse mean-ratio metric," *Math. Program.*, vol. 110, pp. 561–590, 2007.
9. A. Plaza, J. Suárez, M. Padrón, S. Falcón, and D. Amieiro, "Mesh quality improvement and other properties in the four-triangles longest-edge partition," *Comput. Aided Geom. D.*, vol. 21, no. 4, pp. 353 – 369, 2004.
10. J. Shewchuk, "What is a good linear element? Interpolation, conditioning, and quality measures," in *Proc. of the* 11^{th} *International Meshing Roundtable*, pp. 115–126, 2002.
11. J. Nocedal and S. Wright, *Numerical Optimization*. New York: Springer, 2nd ed., 2006.
12. T. Tang, "Moving mesh methods for computational fluid dynamics," in *Proc. of the International Conference on Recent Advances in Adaptive Computation*, vol. 383, Contemporary Mathematics, 2004.
13. S. Shontz and S. Vavasis, "Analysis of and workarounds for element reversal for a finite element-based algorithm for warping triangular and tetrahedral meshes," *BIT Numer. Math.*, vol. 50, pp. 863–884, 2010.
14. S. Shontz and S. Vavasis, "A robust solution procedure for hyperelastic solids with large boundary deformation," *Eng. with Comput.*, vol. 28, no. 2, pp. 135–147, 2012.
15. P. Knupp, "Updating meshes on deforming domains: An application of the target-matrix paradigm," *Commun. Num. Meth. Engr.*, vol. 24, pp. 467–476, 2007.
16. J. Kim, S. Sastry, and S. Shontz, "Efficient solution of elliptic partial differential equations via effective combination of mesh quality metrics, preconditioners, and sparse linear solvers," in *Proc. of the* 19^{th} *International Meshing Roundtable*, pp. 103–120, 2010.
17. R. Bank and R. Smith, "Mesh smoothing using a posterior error estimates," *SIAM J. Numer. Anal.*, vol. 34, pp. 979–997, 1997.
18. L. Freitag and P. Plassmann, "Local optimization-based simplicial mesh untangling and improvement," *Int. J. Numer. Meth. Eng.*, vol. 49, pp. 109–125, 2000.
19. J. Park and S. Shontz, "Two derivative-free optimization algorithms for mesh quality improvement," in *Proc. of the 2010 International Conference on Computational Science*, vol. 1, pp. 387–396, 2010.
20. J. Escobar, E. Rodriguez, R. Montenegro, G. Montero, and J. Gonzalez-Yuste, "Simultaneous untangling and smoothing of tetrahedral meshes," *Comput. Meth. in Applied Mechanics and Engineering*, vol. 192, pp. 2775–2787, 2003.
21. S. Sastry, S. Shontz, and S. Vavasis, "A log-barrier method for mesh quality improvement," in *Proc. of the* 20^{th} *International Meshing Roundtable*, pp. 329–346, 2011.
22. V. Parthasarathy, C. Graichen, and A. Hathaway, "A comparison of tetrahedron quality measures," *Finite Elem. Anal. Des.*, vol. 15, pp. 255–261, 1994.
23. M. Brewer, L. Frietag-Diachin, P. Knupp, T. Laurent, and D. Melander, "The Mesquite Mesh Quality Improvement Toolkit," in *Proc. of the* 12^{th} *International Meshing Roundtable*, pp. 239–250, 2003.
24. "CUBIT generation and mesh generation toolkit." http://cubit.sandia.gov/.
25. H. Si, "TetGen: A Quality Tetrahedral Mesh Generator and Three-Dimensional Delaunay Triangulator." http://tetgen.berlios.de/.
26. P. Knupp. Sandia National Laboratories, Personal communication, 2003.
27. S. Mehrotra, "On the implementation of a primal-dual interior point method," *SIAM J. Optimiz.*, vol. 2, no. 4, pp. 575–601, 1992.