SAND99-0110J

ACHIEVING FINITE ELEMENT MESH QUALITY VIA OPTIMIZATION OF THE JACOBIAN MATRIX NORM AND ASSOCIATED QUANTITIES PART I - A FRAMEWORK FOR SURFACE MESH OPTIMIZATION

PATRICK M. KNUPP
PARALLEL COMPUTING SCIENCES DEPARTMENT
SANDIA NATIONAL LABORATORIES*
M/S 0441, P.O. BOX 5800
ALBUQUERQUE, NM 87185-0441
PKNUPP@SANDIA.GOV

Abstract. Structured mesh quality optimization methods are extended to optimization of unstructured triangular, quadrilateral, and mixed finite element meshes. New interpretations of well-known nodally-based objective functions are made possible using matrices and matrix norms. The matrix perspective also suggests several new objective functions. Particularly significant is the interpretation of the Oddy metric and the Smoothness objective functions in terms of the condition number of the metric tensor and Jacobian matrix, respectively. Objective functions are grouped according to dimensionality to form weighted combinations. A simple unconstrained local optimum is computed using a modified Newton iteration. The optimization approach was implemented in the CUBIT mesh generation code and tested on several problems. Results were compared against several standard element-based quality measures to demonstrate that good mesh quality can be achieved with nodally-based objective functions.

Key words. unstructured grid generation, finite element mesh, mesh optimization

AMS subject classification. 65M50

1. Introduction

Mesh quality remains an important issue in generating useful meshes for finite element calculations. Optimization techniques are a systematic approach to achieving mesh quality and many papers have been written on optimization of finite element meshes. References from the finite element literature include [9], [22] and [25]; an extensive bibiography of papers on this subject is found in [3]. There is also a parallel literature on mesh optimization for structured meshes that is not well-known to the unstructured mesh community, for example [1], [4], [5], [6], [7], [12], and [19]. A major purpose of this paper is to show how techniques from structured mesh optimization can be applied to create promising new objective functions for unstructured mesh optimization. The authors' work in variational grid generation [13], matrix-based optimization [16], and structured mesh optimization [14] and [15] is applied. An important part of this effort was to convert the theory outlined in this paper into a practical algorithm useful to the finite element meshing community. This was done by implementing the algorithms in detail into the CUBIT meshing code [8] and subjecting them to practical meshing problems.

This paper is the first in a planned series of papers on mesh optimization. In this paper discussion is confined to meshes in two dimensions in order to provide a firm foundation for later extensions. Results which hold in 2D do not necessarily extend to 3D but, at least, the use of matrices permits easy generalization. Another important

* THIS WORK WAS FUNDED BY THE DEPARTMENT OF ENERGY'S MATHEMATICS, INFORMATION AND COMPUTATIONAL SCIENCES PROGRAM (SC-31) AND WAS PERFORMED AT SANDIA NATIONAL LABORATORIES. SANDIA IS A MULTIPROGRAM LABORATORY OPERATED BY SANDIA CORPORATION, A LOCKHEED MARTIN COMPANY, FOR THE UNITED STATES DEPARTMENT OF ENERGY UNDER CONTRACT DE-AC04-94AL85000.

DISCLAIMER

issue that is reserved for a later paper is that of mesh tangling. The goal in the present paper is to propose and analyze reasonable objective functions and compare how they perform in benign situations to identify those which are most promising for application to more difficult problems. This paper only considers schemes in which the nodes are moved with fixed connectivity. The topic of edge-swapping in which mesh connectivity is changed as well as nodal positions will be considered within the present framework in a later paper[1]. Another important issue, control of the mesh on the boundary, will also be considered in a subsequent paper.

The outline of this paper is as follows: Section 2 describes the basic building blocks used in constructing 2D nodally-based objective functions. Section 3 discusses a number of objective functions, most of them previously known either from variational grid generation or structured grid optimization, but interpreted in a new light using matrix norms. Section 4 on implementation issues considers the gradients of the building blocks given in section 2. Gradients of complete objective functions are given in Appendix I. Section 4 also discusses various global and local approaches to mesh optimization. Section 5 gives results of several test problems for the objective functions. The paper finishes with section 6, a summary and conclusions.

2. Building Blocks for Nodally-Based Objective Functions

In finite element meshing, the goal of optimization is often stated to be that of improving *element* quality. As with finite element analysis, the focus is on the mesh elements. Such a perspective leads to formulation of mesh optimization functions in terms of element quality metrics such as skew, taper, aspect ratio, and warpage for quadrilateral elements [23] or aspect ratio and maximum/minimum angle for triangular elements. Although optimization of element quality is the main goal, the direct approach of formulating objective functions in terms of element quality has a number of drawbacks. First, geometric descriptions of element quality can lead to non-convex objective functions. Lack of convexity generally means that the existence of a minimum is not guaranteed. Another potential problem with element-based objective functions is that they are cumbersome to implement in a local fashion and difficult to define generally enough to include both triangular and quadrilateral elements. In spite of these difficulties, element-based mesh optimization schemes are feasible. This paper takes an indirect approach which focuses not on the element quality but on the quality of the geometric quantities centered about each node of the mesh. This approach is motivated by the way in which the partial differential equations of structured grid generation are solved using finite difference methods, in which a node-centered stencil plays the key role. Capitalizing on work done in [16], it is shown here that discrete objective functions for finite element meshes can be formulated in terms of quantities that are analogous to, but not the same as, the Jacobian matrix and metric tensor of structured meshing. Convexity is naturally obtained using nodal-based objective functions because neighbor nodes create conflicting requirements which can only be satisfied in a least-squares sense. Another significant advantage of the node-centered approach is that it applies equally well to quadrilateral, triangular, and mixed element meshes. It is an open question to determine the conditions under which a mesh

[1] In unstructured mesh optimization techniques, the term "smoothing" is widely used to refer to moving the nodes of the mesh with fixed connectivity in order to obtain high quality meshes. In structured meshing the term smoothing usually refers to some notion of ellipticity of an underlying PDE. The proper term for general node movement schemes should be *optimization*, with *smoothing* reserved for optimization methods having a connection with ellipticity.

whose quality has been optimized in terms of nodal objective functions necessarily results in a mesh having optimal element quality. However, evidence given in this paper shows that near-optimal element quality can be obtained with nodally-based objective functions.

Let $M \geq 3$ be the valence (i.e., number of edges) of a given interior node of the mesh that is to be "smoothed" and let $\mathcal{M}$ be the set of integers $m = 0, 1, 2, \ldots, M-1$. Assume for the rest of this paper that $m \in \mathcal{M}$. Let the given node be associated with the vector $x \in R^2$ and the neighbor nodes with $x_m \in R^2$. Define $x_M = x_0$ to achieve periodicity. To achieve control over mesh quality, the local objective function $F(x)$ needs to be based, not directly on x, but rather on the important geometric entities associated with the node. The critical quantities are:

1. The M edge vectors, e_m, in the plane:

$$e_m = x_m - x$$

2. The M Jacobian matrices:

$$J_m = [e_m, e_{m+1}] = \begin{pmatrix} x_m - x & x_{m+1} - x \\ y_m - y & y_{m+1} - y \end{pmatrix},$$

and the M metric tensors:

$$G_m = J_m^T J_m .$$

3. The elements of the metric tensor G_m are:

$$\begin{aligned} \ell_m^2 &= e_m \cdot e_m , \\ \ell_{m+1}^2 &= e_{m+1} \cdot e_{m+1} , \\ \beta_m &= e_m \cdot e_{m+1} . \end{aligned}$$

4. Finally, define

$$g_m = \det G_m = \ell_m^2 \, \ell_{m+1}^2 - \beta_m^2 ,$$

and the Jacobian determinant

$$j_m = \det J_m .$$

Then $j_m^2 = g_m$ and j_m is the local area of the m-th parallelogram enclosed by the corresponding pair of edges. For shorthand, let $\mid j_m \mid = \sqrt{g_m}$. It is important to distinguish between j_m and $\sqrt{g_m}$ because the former can be negative or positive, with negative j_m signifying inverted (also called tangled or folded) elements. A mesh with all-positive j_m is generally the minimum acceptable quality mesh. The local area, j_m, and the local angle measure, β_m, depend upon the included angle between the two edges. This angle is not explcitily used in the construction of the present objective functions, but is controlled indirectly via the local area and angle measures.

Although these quantities are patterned after the continuum quantities of mappings, they do not mimic them exactly. In particular, a finite difference discretization of the elements of the metric tensor of the continuum mapping does not involve the center node whereas the discrete metric tensor above does contain the center node. Ramifications of this are most apparent in Part II where the folding problem is considered.

Recent work in continuum variational methods of mesh generation has shown that a powerful approach to the construction of objective functions involves the use of matrices and matrix norms [16]. The Jacobian matrix is the fundamental quantity that describes all the first-order mesh qualities (length, areas, and angles) of interest, therefore, it is appropriate to focus the building of objective functions on the Jacobian matrix or the associated metric tensor. Most of the well-known objective functions in structured mesh optimization can be cast in the form of norms of matrices. The main advantage of this approach, perhaps, is that generalization of a surface mesh objective function to a volume mesh objective function is straightforward. This also helps in interpreting the geometric meaning of objective functions. Another advantage is that matrices permit easy introduction of weighted forms of objective functions for anisotropic mesh quality measures. For these reasons, objective functions constructed from matrices are emphasized.

The matrices of interest must be converted to scalars to create scalar objective functions. This can be done using the trace, determinant, or matrix norms. Matrix norms are not unique but, for mesh generation, the Frobenius norm has proven the most useful and most easily implemented. Let A be a 2×2 matrix, then the Frobenius norm of A is $\mid A \mid_F = (\mathrm{tr}(A^T A))^{\frac{1}{2}}$. The norm-squared is the sum of the square of the elements in the matrix, which is why the Frobenius norm lends itself so readily to the construction of objective functions for meshing. For the rest of this paper the F subscript is omitted since all the norms will be Frobenius norms. The following matrix properties prove useful in analyzing the objective functions used in the present approach. Let $A_{2\times 2}$ be non-singular. Then

$$\mid A^{-1} \mid = \mid A \mid / \mid \alpha \mid$$

with $\alpha = \det(A)$, the determinant of A. Also,

$$\mid A/\sqrt{\mid \alpha \mid} \mid^2 = \mid A \mid \mid A^{-1} \mid = \mid \sqrt{\mid \alpha \mid} A^{-1} \mid^2 = \sqrt{2 + \mid A^T A \mid \mid (A^T A)^{-1} \mid}$$

The following matrices are quite useful:

$$Diag A = \begin{cases} A_{ij} & \text{if } i = j \\ 0 & \text{else,} \end{cases}$$

which is the matrix formed from the diagonal elements of A, and

$$Offdiag\, A = A - Diag\, A\,,$$

which is the matrix consisting of the off-diagonal elements of A.

3. A Parade of 2D Objective Functions

Although many different objective functions are introduced in this section, they overlap considerably and it is believed that not all of these objective functions will ultimately be needed. This intuition is based on results from continuum variational methods where there is one dominant objective function known as Smoothness [2]. However, for theoretical purposes, it is useful to survey objective functions previously proposed for discrete mesh optimization in a systematic framework provided by the use of matrices and their norms. Let f_m be a scalar quality metric for the m-th element attached to the center node, derived from the building blocks in the previous section. For example

$$f_m = f(A_m, B_m, ...)$$

where $A_m, B_m, \ldots$ are matrices constructed from the adjacent edges. Consider the vector $f = (f_0, f_1, \ldots, f_{M-1}) \in R^M$. The p-norm of f is

$$\mid f \mid_p = (\sum_m \mid f_m \mid^p)^{1/p}$$

with $p \geq 1$. Usually $p = 1, 2$ will be used, but the ℓ_∞ norm:

$$\mid f \mid_\infty = \max_m \{\mid f_m \mid\}$$

is also a useful norm because it enables optimization of worse-case mesh quality measures instead of average measures. The ℓ_∞ norm will be considered further in the next paper in connection with mesh untangling.

The general form of the mesh objective functions in this work is $\frac{1}{2} \mid f \mid_p^p$. For finite p, the minimizations are unconstrained, two-variable, and have smooth objective functions. The necessary condition for a minimum to exist are (i) the gradient of the objective function is zero at the minimum, and (ii) the Hessian is positive semi-definite at the minimum [10]. Sufficient conditions are similar, requiring that the Hessian be positive definite.

How, then, is f_m to be constructed? It will be constructed from Frobenius norms of matrices or combinations thereof. The matrix p-norms do not appear useful except possibly for $p = 2$ which gives the modulus of the maximum eigenvalue (see the objective function F_{MEV} below).

It is important to bear in mind that, as each of the objective functions in this section is discussed, not all of the properties given in this section generalize to three-dimensions. The objective functions can be grouped according to their dimensionality (i.e., by powers of length). This is useful because, as Brackbill has observed, it is difficult to reliably combine objective functions of mixed dimensionality.

Group I: Objective Functions whose dimensions are $(length)^4$:

p-Length

$$\begin{aligned} F_{pL} &= \frac{1}{2} \sum_m \mid J_m \mid^{2p}, \\ &= \frac{1}{2} \sum_m (\ell_m^2 + \ell_{m+1}^2)^p \end{aligned}$$

with $p \geq 1$. The "Length" objective function [4] is obtained with $p = 1$ and the so-called "Length-weighted Laplacian" [8] is approximated when $p = 2$. Dimensions of $(length)^4$ requires $p = 2$. The effect of the parameter p on mesh quality is considered in section 5.

Norm of the Metric Tensor (NMT)

An early proponent of this functional showed that the continuum version of this functional was convex [20]:

$$F_{NMT} = \frac{1}{2} \sum \mid G_m \mid^2,$$

$$= \frac{1}{2}\sum_m \ell_m^4 + \ell_{m+1}^4 + 2\beta_m^2 ;$$
$$= \frac{1}{2}\sum_m (\ell_m^2 + \ell_{m+1}^2)^2 - 2\,g_m .$$

As noted in [16], because G_m is symmetric, the square of the norm of the metric tensor is the sum of the squares of its eigenvalues. Ideally, minimization of this functional would make the eigenvalues of the metric tensor equal. The metric tensor would then be proportional to the identity matrix, giving a locally orthogonal mesh having equal aspect ratios.

Angle
The Angle objective function:

$$F_< = \frac{1}{4}\sum_m \mid Offdiag\, G_m \mid^2 ,$$
$$= \frac{1}{2}\sum_m \beta_m^2$$

attempts to equidistribute the angles about a node. For $M = 4$, the "orthogonality" objective function is obtained [4]. Although geometrically appealing, Angle suffers the great drawback that it is non-convex in general, i.e., the minimal mesh does not always exist. To make practical use of Angle requires combining it with other objective functions to get convexity. For this paper Angle is combined with 2-Length since they both have dimensions of $(length)^4$.

Group I Combinations
Because all the objective functions in Group I are dimensionally homogeneous, they are easy to combine. Let μ. ν be two real parameters and define the Group I combination

$$F_I(\mu,\nu) = \frac{1}{2}\sum_m \mu\,(1-\frac{3}{2}\nu)\mid G_m \mid^2 + (1-\mu-\nu+\frac{3}{2}\mu\,\nu)\mid J_m \mid^4 + (1-\mu)\,\nu \mid Offdiag\, G_m \mid^2$$

The combined objective function is bounded below by $\frac{1}{2}\,(1-\mu\,\nu)\sum_m \mid Offdiag\, G_m \mid^2$. The weighted interpolation gives Length-weighted Laplacian for $\mu = 0$, $\nu = 0$, the Norm of the Metric Tensor for $\mu = 1$, $\nu = 0$, and Angle for $\mu = 0$, $\nu = 1$. Grouping the objective functions in this way is an improvement over earlier proposals because the group is dimensionally homogeneous and because more known functions are included in the group. For example, three other important objective functions found in this group are given below.

Area
The Area functional [4], [1] makes locally equal areas:

$$F_A = \frac{1}{2}\sum_m g_m$$
$$= \frac{1}{4}\sum_m \mid J_m \mid^4 - \mid G_m \mid^2 .$$

This can be obtained with $\mu = 1$, $\nu = 1$ because area is 2-Length minus NMT. In the continuum it is non-elliptic, so non-smooth grids result. This property carries over to the discrete optimization function for area.

Equal EigenValues of G_m

$$\begin{aligned} F_{EE} &= \frac{1}{2}\sum_m (\ell_m^2 + \ell_{m+1}^2)^2 - 4g_m\,, \\ &= \frac{1}{2}\sum_m (\ell_m^2 - \ell_{m+1}^2)^2 + 4\beta_m^2\,, \\ &= \frac{1}{2}\sum_m 2 \mid G_m \mid^2 - \mid J_m \mid^4 \,. \end{aligned}$$

This objective function tries to make the eigenvalues of G equal by minimizing the radicand in the expression for the eigenvalues. Geometrically it means orthogonality and elements with unit aspect ratio. It can be derived from Group I with $\mu = 2$, $\nu = 0$.

One can also get the "AO" (area-orthogonality) objective function [18] with $\mu = \frac{1}{2}$ and $\nu = 1$.

Group II: Objective Functions whose dimensions are $(length)^2$:

1-Length

$$\begin{aligned} F_L &= \frac{1}{2}\sum_m \mid J_m \mid^2\,, \\ &= \frac{1}{2}\sum_m \ell_m^2 + \ell_{m+1}^2\,. \end{aligned}$$

This objective function leads to standard Laplacian smoothing, as opposed to length-weighted Laplacian.

Frobenius Norm of G_m

$$\begin{aligned} F_G &= \frac{1}{2}\sum_m \mid G_m \mid\,, \\ &= \frac{1}{2}\sum_m \sqrt{\ell_m^4 + \ell_{m+1}^4 + 2\,\beta_m^2}\,, \\ &= \frac{1}{2}\sum_m \sqrt{(\ell_m^2 + \ell_{m+1}^2)^2 - 2\,g_m}\,. \end{aligned}$$

In section 5, F_G is compared to the NMT objective function to see how the lower power affects quality.

Maximum EigenValue of G_m

$$F_{MEV} = \frac{1}{2}\sum_m \mid J_m \mid^2 + \sqrt{2 \mid G_m \mid^2 - \mid J_m \mid^4}$$

F_{MEV} has been implemented with the plus sign, giving a minimization of the maximum eigenvalue. This combination of Frobenius norms of matrices is equivalent to the non-Frobenius, 2-norm of G_m.

Group II Combinations

Although it is possible to combine the given objective functions given in Group II, there seems no theoretical nor practical advantage to doing so.

Group III: Objective Functions whose dimensions are $(length)^0$:

Smoothness (Condition Number of the Jacobian Matrix)

The "Smoothness" functional was first proposed by Brackbill [2] as a continuum variational principle. It's discrete analog is:

$$
\begin{aligned}
F_S &= \frac{1}{2}\sum_m |\, J_m^{-1} \,||\, J_m \,|, \\
&= \frac{1}{2}\sum_m \sqrt{2 + |\, G_m \,||\, G_m^{-1} \,|}, \\
&= \frac{1}{2}\sum_m (\ell_m^2 + \ell_{m+1}^2)/\sqrt{g_m}
\end{aligned}
$$

and is closely related to the Winslow smoother [24]. One sees that Smoothness can be viewed as minimizing the condition number of the Jacobian matrix, a new interpretation of this objective function. This would seem to be a good idea because minimization of the condition number would maximize the distance to the set of singular matrices ([11], p26). The second equation above shows the relation between Smoothness and the Oddy objective function to be given next.

Oddy's Metric and the Condition Number of the Metric Tensor

This objective function minimizes the condition number of the metric tensor:

$$
\begin{aligned}
F_{CNG} &= \frac{1}{2}\sum_m |\, G_m^{-1} \,||\, G_m \,|, \\
&= \frac{1}{2}\sum_m (\ell_m^4 + \ell_{m+1}^4 + 2\beta_m^2)/g_m\,, \\
&= \frac{1}{2}\sum_m (\ell_m^2 + \ell_{m+1}^2)^2/g_m - 2\,.
\end{aligned}
$$

There are a number of related objective functions: $F_{ModLiao} = F_{CNG} + M$ gives the discrete form of the "modified Liao" functional proposed in [13]. The Oddy metric [21] is closely related to the condition number objective function:

$$
F_{Oddy} = \frac{1}{2}\sum_m [(\ell_m^2 - \ell_{m+1}^2)^2 + 4\,\beta_m^2]/g_m\,,
$$

i.e., $F_{Oddy} = F_{CNG} - M$. The Oddy metric was derived for quadrilaterals using ideas from the theory of elasticity. By focusing on node-centered objective functions and matrix norms, the Oddy metric is applicable to any element-type. Because these objective functions differ by only a constant, they have the same stationary points and are thus equivalent. Thus, a new interpretation of the Oddy metric has been presented.

Aspect Ratio

A dimensionless aspect-ratio objective function can be posed in terms of the condition number of the matrix $Diag\, G_m$:

$$\begin{aligned} F_{AR} &= \frac{1}{2}\sum_m \mid Diag\, G_m \mid\mid (Diag\, G_m)^{-1} \mid , \\ &= \frac{1}{2}\sum_m (\frac{\ell_m}{\ell_{m+1}})^2 + (\frac{\ell_{m+1}}{\ell_m})^2 . \end{aligned}$$

This objective function tends to make the two edge lengths ℓ_m and ℓ_{m+1} equal. It is not suggested that this objective function actually be used since it is non-convex, but it is interesting because it allows another theoretical interpretation of the Oddy metric. Because $G_m = Diag\, G_m + Offdiag\, G_m$, one can show that the following is identically true:

$$g_m \mid G_m \mid\mid G_m^{-1} \mid = \ell_m^2\, \ell_{m+1}^2 \mid Diag\, G_m \mid\mid (Diag\, G_m)^{-1} \mid + 2\,\beta_m^2$$

Another interpretation of Oddy, then, is that it minimizes a combination of the Aspect ratio and Angle functions.

The Smoothness, Oddy, and Aspect Ratio objective functions appear to be the only non-dimensional functions of interest and no combined objective function is proposed.

Group IV: Objective Functions whose dimensions are $(length)^{-2}$:

The only interesting objective function belonging to this group is inverse 1-Length. The inverse p-Length function is discussed next.

Group V: Objective Functions whose dimensions are $(length)^{-4}$:

Inverse p-Length

$$\begin{aligned} F_{Ip} &= \frac{1}{2}\sum_m \mid J_m^{-1} \mid^{2p} , \\ &= \frac{1}{2}\sum_m (\frac{\ell_m^2 + \ell_{m+1}^2}{g_m})^p \end{aligned}$$

with $p \geq 1$. One needs $p = 2$ to get dimensions of $(length)^{-4}$. All the objective functions given in this paper are analogous to continuum variational integrals over the logical space so, for example, Inverse 1-Length is not the same as Brackbill's variational Smoothness, rather, F_S is.

Inverse NMT

$$F_{INMT} = \frac{1}{2}\sum_m \mid G_m^{-1} \mid^2 ,$$

$$= \frac{1}{2} \sum_m \frac{\ell_m^4 + \ell_{m+1}^4 + 2\beta_m^2}{g_m^2},$$
$$= \frac{1}{2} \sum_m \frac{(\ell_m^2 + \ell_{m+1}^2)^2 - 2\, g_m}{g_m^2}.$$

Group V Combinations
Let μ, ν be parameters.

$$F_I(\mu,\nu) = \frac{1}{2} \sum_m \mu\,(1 - \frac{3}{2}\nu) \mid G_m^{-1} \mid^2 + (1 - \mu - \nu + \frac{3}{2}\mu\,\nu) \mid J_m^{-1} \mid^4 + (1-\mu)\,\nu \mid Offdiag\, G_m^{-1} \mid^2$$

This weighted interpolation gives Inverse 2-Length for $\mu = 0$, $\nu = 0$, Inverse NMT for $\mu = 1$, $\nu = 0$, Inverse Equal Eigenvalue for $\mu = 2$, $\nu = 0$, and Inverse Area for $\mu = \nu = 1$. This group of discrete objective functions is similar to the continuum group proposed in [2] but is not identical because inverse NMT is used in place of Brackbill's inverse area and inverse length-weighted Laplacian is used in place of Brackbill's inverse length.

Other nodally-based mesh objective functions can and have been devised. Many of these cannot be posed in terms of matrix norms (e.g., see [14] and [12]). The objective functions given in this paper provide an adequate basis for future extensions.

4. Optimization Approach

This section considers the means by which the local objective functions introduced in the previous section are combined into a global minimization problem and how the problem of finding a minimum is solved. Before doing so, however, brief comments are made on how the optimization method is extended from the plane to surfaces embedded in R^3 using solid model geometry and the tangent plane.

4.1 Tangent Plane Algorithm

Minimization of objective functions on surfaces is greatly simplified by projecting nodal positions into the tangent plane. To define a tangent place for a position p on a surface representing the coordinates of the center node, let $\hat{n}$ be the unit surface normal at that point. Let the vectors p_m correspond to the nodes attached to the center node, and let $v_m = p_m - p$. The local tangent plane axes are $\hat{\imath}$, $\hat{\jmath}$ where

$$\hat{\jmath} = (\hat{n} \times v_0) / \mid \hat{n} \times v_0 \mid,$$
$$\hat{\imath} = \hat{\jmath} \times \hat{n}.$$

Notice that $m = 0$ is arbitarily chosen in this definition. The local tangent plane coordinates of v_m are w_m where

(1) $$w_m = [p + v_m \cdot \hat{\imath}, x + v_m \cdot \hat{\jmath}]^T$$

The solid model geometry "move-to-owner" operation then moves the new central node position w_m to the closest point on the surface.

4.2 Gradients of the Building Blocks.

Regardless of how the global objective function is minimized, the gradients of the building blocks given in section 2 will be needed. These are catalogued here.

$$\begin{aligned}
\nabla \ell_m^2 &= -2\,e_m = 2\,(x - x_m)\,, \\
\nabla \ell_{m+1}^2 &= -2\,e_{m+1} = 2\,(x - x_{m+1})\,, \\
\nabla(\ell_m^2 + \ell_{m+1}^2) &= -2\,(e_m + e_{m+1}) = 4\,(x - \bar{x}_m)\,,
\end{aligned}$$

and

$$\nabla \beta_m = -(e_m + e_{m+1}) = 2\,(x - \bar{x}_m)\,,$$

where $\bar{x}_m = \frac{1}{2}\,(x_m + x_{m+1})$. Define

$$d_m = x_{m+1} - x_m\,.$$

Then

$$j_m = x \cdot d_m^{\perp} + \Delta_m$$

where $\Delta_m = x_m y_{m+1} - x_{m+1} y_m$. The gradient is

$$\nabla j_m = d_m^{\perp}$$

and

$$\begin{aligned}
\nabla g_m &= 2\,j_m\,d_m^{\perp} \\
&= 2\,[D_m x + \Delta_m d_m^{\perp}]
\end{aligned}$$

where $D_m = d_m^{\perp} \otimes d_m^{\perp}$. Finally,

$$\nabla \sqrt{g_m} = \frac{1}{2\sqrt{g_m}}\,\nabla g_m$$

with $\sqrt{g_m} \neq 0$.

4.3 Non-linear, Multi-objective Optimization.

Local objective functions have been defined which are functions of a given "smooth-node" of the mesh. This section describes how the local objective functions are used to simultaneously optimize all nodes in the global mesh. In structured mesh optimization techniques the local objective function is summed over all the interior nodes of the mesh to form the global objective function, which is then minimized. This approach has been unfairly criticized as being too slow [3] when, in fact, with proper care it can be quite fast [4]. An alternative approach used in this paper is to separately minimize each of the local objective functions (taking only a single step toward the minimum) and iterating over all the nodes of the mesh. This approach is analogous to solving a partial differential equation using finite difference methods and point relaxation. For some objective functions such as Length, the "local" and "global" approaches are equivalent but, in general, they are not. The main advantage of the "local" approach is the ease with which it is implemented. Another advantange is that local mesh quality is often easier to achieve than global mesh quality.

Local nodally-based objective functions are functions of only two variables. For example, it is easy to see that Length and Area are bi-quadratic functions with positive semi-definite Hessians. The functions in Group I are bi-quartic forms, while the dimensionless Group III and the inverse Groups IV and V are rational polynomial functions of two variables.

The following Modified Newton Search is applied to perform the local optimization. Let the objective function be $F = F(x, y)$ and assume F is differentiable on some subregion of the plane. Let $\nabla F = G$ be the gradient of F. The goal is to find (x, y) such that $G(x, y) = 0$. In terms of the Hessian matrix of F, defined to be the matrix with i, j-th element $\partial^2 F/\partial x_i \partial x_j$, the Newton iteration is

$$x_{n+1} = x_n - H^{-1}(x_n)\, G(x_n)\,.$$

In the present context, because there is an outer iteration as well as the local minimization, only a single step is taken towards the minimum per node relaxation. The initial "guess" is just $x_n = 0$ because the local coordinate system is centered at the smooth-node. Thus, the Newton iteration takes the form

$$x_{n+1} = -H^{-1}(0)\, G(0)\,.$$

Sufficient conditions to guarantee convergence are that the initial position be "close" to the minimum and that the Hessian be symmetric, positive definite.

As an example, consider the 2-Length objective function. The gradient is

$$G = 4 \sum_m (\ell_m^2 + \ell_{m+1}^2)\,(x - \bar{x}_m)$$

and the Hessian matrix is

$$H = 4 \begin{pmatrix} \sum_m [(\ell_m^2 + \ell_{m+1}^2) + 4\,(x - \bar{x}_m)^2] & 4 \sum_m (x - \bar{x}_m)\,(y - \bar{y}_m) \\ 4 \sum_m (x - \bar{x}_m)\,(y - \bar{y}_m) & \sum_m [(\ell_m^2 + \ell_{m+1}^2) + 4\,(y - \bar{y}_m)^2] \end{pmatrix}.$$

The basic Newton iteration is modified by approximating the Hessian as follows. If $M\,x \approx \sum_m \bar{x}_m$, then $H \approx 4 \sum_m (\ell_m^2 + \ell_{m+1}^2)\, I$ and the iteration can be written

$$x = \sum_m (\ell_m^2 + \ell_{m+1}^2)\, \bar{x}_m / \sum_m (\ell_m^2 + \ell_{m+1}^2)\,,$$

which is simply a re-arrangement of the equation $G = 0$. A similar approach is taken for the other objective functions in this paper as well (see Appendix I for details). For some of the objective functions it is necessary to under-relax the step towards the minimum in order to obtain convergence.

The combination function for Group I was experimented with briefly. Good mesh quality could be obtained for μ, ν inside the unit square but not too close to pure Angle or pure Area and also for combinations below but not too far from the unit square. Convexity appears to be lost for combinations above the unit square.

5. Results and Discussion.

To evaluate the various objective functions proposed in this paper, selected element quality measures are used for comparison. For quadrilateral meshes (1) maximum skew (defined in [23]), (2) maximum angle in a quadrilateral (in degrees), (3) minimum angle in a quadrilateral (in degrees), (4) the maximum Oddy metric (defined in [21]), and (5) the minimum scaled Jacobian ($j_m/\ell_m/\ell_{m+1}$) were used, while for triangular element meshes the metrics (1) aspect ratio, (2) maximum angle, (3) minimum angle, and (4) minimum scaled Jacobian were selected.

Objective functions were tested on relatively benign test problems because, if a method could not do well on these, there would be no point in considering it further. More difficult test problems in subsequent papers will further shorten the list of worthwhile objective functions. Figures 1 - 2 show initial non-optimized, paved-quadrilateral and DeLaunay-triangulated meshes on two test geometries. The initial meshes are already of good quality, so it is a challenge for the optimizers to improve quality. To compare results for a given quality metric, objective functions were divided into three cases according to how they performed. Case I: optimized mesh quality better than the Winslow smoother[2], Case II: mesh quality worse than Winslow but better than the intial paved or triangulated mesh, Case III: mesh quality worse than the intial mesh. Angle was weighted with the 2-Length objective function ($\mu = 0$, $\nu = 0.5$). Since the objective functions are based on 2-norms of f, one expects them to improve the *average* quality metrics, not their extremes. However, it was found that, for the benign test problems, if a method did well on quality metric averages, it also did well on the extremes. Table 1 gives results for the paved mesh in figure 1.

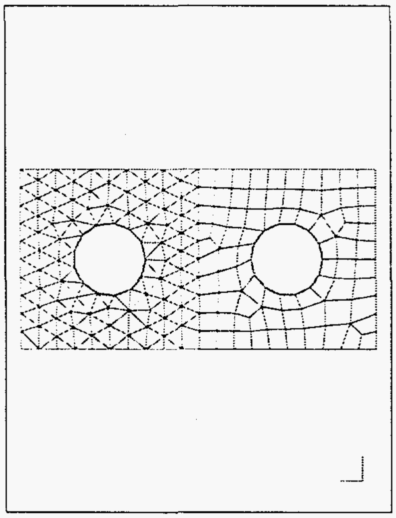

FIG. 1. Initial Triangle and Paved Meshes on Hole Geometry

All of the optimizers converged to a solution on most of the tests, indicating convexity of the objective functions. The quality metrics computed for the optimized mesh included element quality on the boundary of the mesh. This is somewhat unfair to the present approach since the smoothing takes place only on the interior nodes. Improving quality on the boundary is the subject of a future paper.

The only consistent winner was Oddy[3], which beat Winslow about half the time and fell in Case II the rest of the time. Length and Smoothness were also effective. The next best objective functions were NMT, MEV, and NormG, which performed similarly but were somewhat weak on the minimum angle criterion. Length-weighted Laplacian was mediocre, while F_{EE} (equal eigenvalue) was rejected because it exhibited rather poor behavior (i.e., lack of convergence, folded meshes) on the brick

[2] Winslow [17] is a highly effective variational-based smoother in two-dimensions against which all the optimizers should be compared.

[3] The maximum Oddy metric in the table was not obtained with the Oddy objective function because the latter used an ℓ_2 norm while the former is an ℓ_∞ norm.

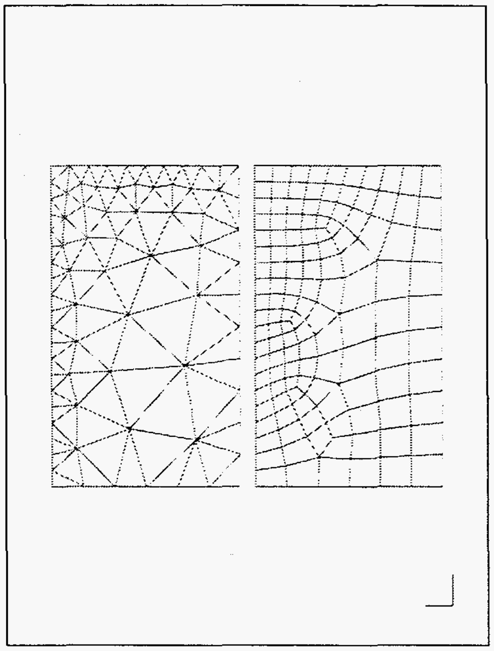

FIG. 2. Initial Triangle and Paved Meshes on Brick Geometry

TABLE 1
Extreme Quality Metrics for Optimized Quadrilateral Mesh (Hole Geometry)

Method	Max Skew	Max Angle	Min Angle	Max Oddy	Min Scaled J
Pave	0.4838	134.7	55.04	1.997	0.7103
Length	0.4385	131.1	56.88	1.704	0.7540
LWLength	0.4793	134.3	55.20	2.041	0.7156
NMT	0.4595	132.6	56.51	1.841	0.7366
Area	0.8148	164.5	31.47	3.137	0.2665
EigenValue	0.4342	132.4	57.03	1.673	0.7379
Inverse Length	0.5531	141.7	50.18	3.412	0.6201
Inverse LWLength	0.5909	144.5	46.94	4.249	0.5803
Inverse NMT	0.5540	142.6	49.44	3.670	0.6079
Oddy	0.4429	132.7	55.99	1.858	0.7353
Angle	0.4011	136.6	51.39	9.511	0.6867
Smoothness	0.4440	132.8	56.49	1.873	0.7338
Norm G	0.4399	131.1	57.41	1.715	0.7531
MEV	0.4393	131.1	57.17	1.710	0.7535
Winslow	0.4395	133.0	57.19	1.871	0.7313

geometry and others. The inverse objective functions of Groups IV and V did rather poorly, often doing worse than the paving algorithm itself. Area and Angle fared very poorly in general, both lacking smoothness and the latter, convexity. These two optimizers should be avoided in general but may prove useful in certain special instances. For production meshing, Length, Oddy (see Figure 3), and Smoothness appear to be the most effective.

Performance relative to the taper element quality metric [23] was briefly considered. None of the objective functions did well on this metric except area, probably because taper is not easily described by a nodally-based quality measure. This result indicates the need for developing element-based objective functions for these kinds of

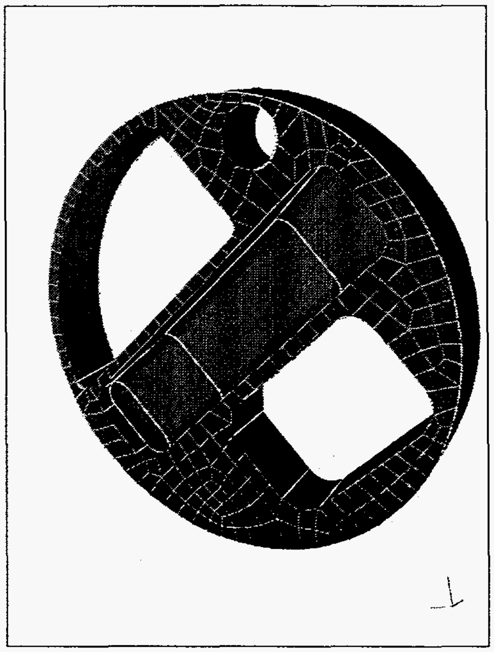

FIG. 3. Paved Mesh Optimized Using Oddy Objective Function

element quality metrics.

The performance of the objective functions was also examined for the triangle meshes in figures 1 and 2. It was found that a similar pattern held as it did for quadrilateral meshes, with best results obtained for the Oddy and Smoothness objective functions and acceptable results with Length, 2-Length, Eigenvalue, Inverse 2-Length, NormG, and MEV. Particularly bad results were obtained with Angle. An important result was that the same nodally-based objective functions that work well for quadrilateral meshes also work well for triangular meshes.

A study of mesh quality vs. the parameter p in the p-Length objective function was performed to determine if an optimal value of p exists. The test cases varied p from 0.5 to 20 (see Figure 4). It was found that larger values of p tended to give poorer mesh quality (greater skew, worse angles) than smaller values of p. Best quality was obtained with $p = 1$ or $p = \frac{1}{2}$. The results showed $p < 1$ tended to straighten out the mesh "lines" while large p made the grid less smooth. This is expected since the optimizer is getting close to the ℓ_∞ norm. Larger p appear to have a somewhat greater resistance to mesh folding (thus explaining why length-weighted Laplacian is sometimes preferred over Length), but folding is not prevented.

6. Summary and Conclusions.
This paper has focused on the use of matrix norms to define and interpret discrete objective functions for finite element mesh quality. Especially significant was the interpretation of the "Smoothness" objective function in terms of the Jacobian matrix condition number and the Oddy metric as the condition number of the metric tensor. By using matrix norms, the objective functions will be easy to generalize to three dimensions. Objective functions were stated in terms of nodally-based geometric entities. The objective functions were implemented in a production meshing code and shown to be effective in achieving good element quality on several benign problems.

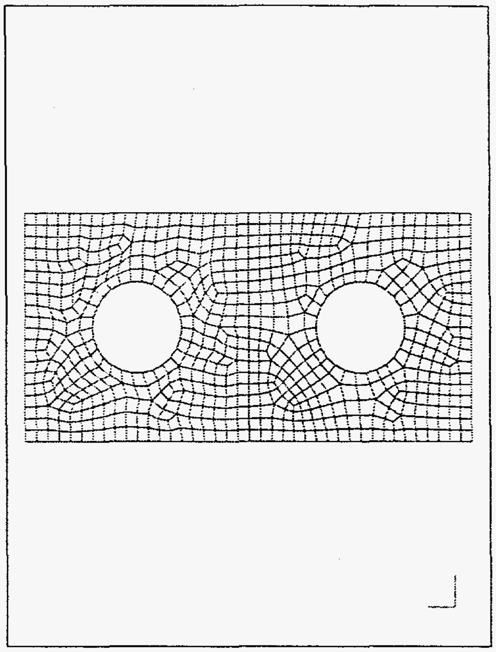

FIG. 4. Length Optimization: $p = 20$ (left) vs. $p = 1$ (right)

The same objective functions that were effective on quadrilateral meshes were also effective on triangular meshes. Speed and efficiency issues were not considered although it has been observed in practice that no significant time penalty applies to optimization with one objective function as compared to another.

In closing, recall that a folded mesh is one which contains a node for which $j_m < 0$ for some m. These meshes are, in general, useless for finite element analysis and are strongly avoided by analysts. The objective functions given here in Groups I and II have no resistance to folding and can quite easily do so. That is, given an initially unfolded mesh, these optimizers can produce an optimal mesh that is folded. In addition, the objective functions in Groups III, IV, and V do not work on initially folded meshes because the optimal mesh is separated from the initial mesh by an infinite barrier. Methods to overcome these problems are discussed in Part II of this series of papers.

Acknowledgements

The author dedicates this paper to the memory of Steve Askew and John Wormeck.

REFERENCES

[1] P. Barerra, A. Perez, L. Castellanos, *Curvilinear Coordinate System Generation over Plane Irregular Regions*, Facultad de Cienas UNAM, Mexico, 1992.

[2] J. Brackbill, J. Saltzman, *Adaptive zoning for singular problems in two dimensions*, J. Comp. Phys., **46**, pp342-368, 1982.

[3] S. Canann, J. Tristano, M. Staten, *An Approach to Combined Laplacian and Optimization-Based Smoothing for Triangular, Quadrilateral, and Quad-Dominant Meshes*, Procedings of the 7th International Meshing RoundTable, pp479-494, Detroit, 1998.

[4] J. Castillo, *On Variational Grid Generation*, Ph.D. thesis, Univ. New Mexico, 1987.

[5] J. Castillo, *A direct variational grid generation method: Orthogonality control*, in Numerical

Grid Generation in Computational Fluid Mechanics '88, S. Sengupta, J. Hausser, P. Eiseman, and J. Thompson, eds., pp247-256, Pineridge Press, Swansea UK, 1988.

[6] J. Castillo, *An adaptive direct variational grid generation method*, Computers Math. Applic., Vol. 4, No. 1, pp1-9, 1990.

[7] J. Castillo, *A discrete variational method*, SIAM J. Sci. Stat. Comput., Vol. 12, No. 2, pp454-468, 1991.

[8] *CUBIT Mesh Generation Environment*, SAND94-1100, Sandia National Laboraties, Albuquerque, NM 87185.

[9] L. Freitag, M. Jones, and P.Plassman, *An efficient parallel algorithm for mesh smoothing*, In Proceedings of the Fourth International Meshing RoundTable, Sandia National Laboratories, pp47-58, 1995.

[10] P. Gill, W. Murray, M. Wright, *Practical Optimization*, Academic Press, 1981.

[11] G. Golub and C. Van Loan, *Matrix Computations*, Johns Hopkins University Press, 1983.

[12] S. Kennon, G. Dulikravitch, *Generation of Computational grids using optimization*, AIAA Journal, Vol. 24, No. 7, pp1069-1073, 1986.

[13] P. Knupp and S. Steinberg, *The Fundamentals of Grid Generation*, CRC Press, Boca Raton FL, 1993.

[14] P. Knupp, *Robust Grid Generation on Curves and Surfaces*, Ph.D. thesis, Univ. New Mexico, 1989.

[15] P. Knupp, *The direct variational grid generation method extended to curves*, Appl. Math. Comp., **43**, pp65-78, 1991.

[16] P. Knupp, N. Robidoux, *A Framework for Variational Grid Generation: Conditioning the Jacobian Matrix with Matrix Norms*, submitted for publication.

[17] P. Knupp, *Winslow Smoothing on Two-Dimensional Unstructured Meshes*, Procedings of the 7th International Meshing RoundTable, pp449-457, Detroit MI, October 1998.

[18] P. Knupp, *A robust ellptic grid generator*, J. Comp. Phys., **100**, pp409-418, 1992.

[19] A. Kumar, N. Kumar, *A new approach to grid generation based on a local optimisation*, in Numerical Grid Generation in Computational Fluid Mechanics '88, S. Sengupta, J. Hausser, P. Eiseman, and J. Thompson, eds., pp177-184, Pineridge Press, Swansea UK, 1988.

[20] G. Liao, *Variational Approach to Grid Generation*, Num. PDE's, **8**, pp143-147, 1992.

[21] A. Oddy, J. Goldak, M. McDill, M.Bibby, *A distortion metric for isoparametric finite elements*, Trans. CSME, No. 38-CSME-32, Accession No. 2161, 1988.

[22] A. Pardhanani, G. Carey, *Optimization of Computational Grids*, Num. Meth. PDE's, Vol. 4, No. 2, pp95-117, 1988.

[23] J. Robinson, *CRE method of element testing and the Jacobian shape parameters*, Eng. Comput., Vol 4, pp113-118, June 1987.

[24] A.M. Winslow, *Numerical solution of the quasilinear Poisson equation in a nonuniform triangle mesh*, J. Comp. Phys., **2**, pp149-172, 1967.

[25] P. Zavattieri, *Optimization Strategies in Unstructured Mesh Generation*, Intl. J. Num. Meth. Engr., Vol. 39, p2055-2071, 1996.

Appendix I: Gradients and Iteration Matrices of the Functionals

p-Length

$$\nabla F_{pL} = 2p \sum_m (\ell_m^2 + \ell_{m+1}^2)^{p-1} (x - \bar{x}_m)$$

Iteration:

$$x = \frac{\sum_m (\ell_m^2 + \ell_{m+1}^2)^{p-1} \bar{x}_m}{\sum_m (\ell_m^2 + \ell_{m+1}^2)^{p-1}}$$

For $p = 1$, **Length**

$$\begin{aligned} \nabla F_L &= -\sum_m e_m + e_{m+1} \\ &= 2 \sum_m (x - \bar{x}_m) \end{aligned}$$

Iterative scheme:

$$x = \frac{1}{2M} \sum_m \bar{x}_m$$

Hence it is seen that Length Optimization gives Laplacian smoothing.
For $p = 2$, **Length-weighted Laplacian**

$$\nabla F_{L2} = 4 \sum_m (\ell_m^2 + \ell_{m+1}^2) (x - \bar{x}_m)$$

Iteration:

$$x = \frac{\sum_m (\ell_m^2 + \ell_{m+1}^2) \bar{x}_m}{\sum_m (\ell_m^2 + \ell_{m+1}^2)}$$

NMT

$$\nabla F_{NMT} = 2 \sum_m 2 (\ell_m^2 + \ell_{m+1}^2)(x - \bar{x}_m) - j_m d_m^{\perp}$$

The iteration is $T_{NMT}\, x = r_{NMT}$ with

$$\begin{aligned} T_{NMT} &= \sum_m 2 (\ell_m^2 + \ell_{m+1}^2) I - D_m \\ r_{NMT} &= \sum_m 2 (\ell_m^2 + \ell_{m+1}^2) \bar{x}_m + \Delta_m\, d_m^{\perp} \end{aligned}$$

Area

$$\nabla F_A = \sum j_m\, d_m^{\perp}$$

Iterative scheme is $T_A\, x = r_A$ where

$$T_A = \sum_m D_m$$

$$r_A = -\sum_m \Delta_m\, d_m^{\perp}$$

Note that T_A is a 2×2 constant matrix and thus this is a linear system of equations. T_A may be singluar but only under perverse circumstances.

Inverse p-Length

$$\nabla F_{Ip} = p \sum_m \left(\frac{\ell_m^2 + \ell_{m+1}^2}{g_m}\right)^{p-1} \left[\frac{2\, g_m\, (x - \bar{x}_m) - (\ell_m^2 + \ell_{m+1}^2)\, j_m\, d_m^{\perp}}{g_m^2}\right]$$

If $j_m > 0$ for all m, this can be expressed as

$$\nabla F_{Ip} = 2\, p \sum_m \frac{1}{g_m} \left(\frac{f_m}{\sqrt{g_m}}\right)^{p-1} \left[x - \bar{x}_m - \frac{1}{2}\, f_m d_m^{\perp}\right]$$

where $f_m = \frac{\ell_m^2 + \ell_{m+1}^2}{\sqrt{g_m}}$. The latter gradient gives rise to the iteration $T_{Ip}\, x = r_{Ip}$ with

$$T_{Ip} = 2\, p \sum_m \frac{1}{g_m} \left(\frac{f_m}{\sqrt{g_m}}\right)^{p-1} I$$

$$r_{Ip} = 2\, p \sum_m \frac{1}{g_m} \left(\frac{f_m}{\sqrt{g_m}}\right)^{p-1} \left[\bar{x}_m + \frac{1}{2}\, f_m d_m^{\perp}\right]$$

Inverse NMT

Gradient:

$$\nabla F_{INMT} = 2 \sum_m \left[\frac{2\, g_m^2\, (\ell_m^2 + \ell_{m+1}^2)\, (x - \bar{x}_m) - \{g_m^2 + g_m\, [\, (\ell_m^2 + \ell_{m+1}^2)^2 - 2\, g_m\,]\, \}\, j_m\, d_m^{\perp}}{g_m^4}\right]$$

If $j_m > 0$ for all m, this can be expressed as

$$\nabla F_{INMT} = 4 \sum_m \frac{f_m}{g_m^{3/2}} \left[x - \bar{x}_m - \frac{f_m - 1}{2 f_m}\, d_m^{\perp}\right]$$

The latter gradient gives rise to the iteration $T_{INMT}\, x = r_{INMT}$ with

$$T_{INMT} = \sum_m \frac{f_m}{g_m^{3/2}}\, I$$

$$r_{INMT} = \sum_m \frac{f_m}{g_m^{3/2}} \left[\bar{x}_m + \frac{f_m - 1}{2 f_m}\, d_m^{\perp}\right]$$

Note: iterating on expressions with g_m works better than with β_m.

Condition Number of G_m

Note: even though this functional is dimensionless, its gradient is not. Gradient:

$$\nabla F_{CNG} = \sum_m \frac{4\, g_m\, (\ell_m^2 + \ell_{m+1}^2)\, (x - \bar{x}_m) - (\ell_m^2 + \ell_{m+1}^2)^2\, (\Delta_m\, d_m^{\perp} + D_m\, x)}{g_m^2}$$

If $j_m > 0$ for all m, this can be expressed as

$$\nabla F_{CNG} = 4 \sum_m \frac{f_m}{\sqrt{g_m}} [x - \bar{x}_m - f_m \frac{d_m^\perp}{4}]$$

The latter gradient gives rise to the iteration $T_{CNG}\, x = r_{CNG}$ with

$$T_{CNG} = \sum_m 4 \frac{f_m}{\sqrt{g_m}} I$$

$$r_{CNG} = \sum_m 4 \frac{f_m}{\sqrt{g_m}} [\bar{x}_m + f_m \frac{d_m^\perp}{4}]$$

Note the similarity to the Smoothness gradient. These iterations can only be performed on untangled meshes. Iterating on tangled meshes causes explosions. The actual algorithm freezes the node if it detects a negative j_m.

Angle

$$\nabla F_< = 2 \sum_m \beta_m (x - \bar{x}_m)$$

Iterative scheme is $T_<\, x = r_<$ where

$$T_< = \sum_m (e_m + e_{m+1}) \otimes \bar{x}_m$$

$$r_< = \sum_m [(x_m \cdot e_{m+1}) + (x_{m+1} \cdot e_m)]\bar{x}_m - 2\, \beta_m\, x$$

Because the goal is to make β_m "small", a non-straightforward non-linear interation is needed.

Smoothness
Let $f_m = \frac{\ell_m^2 + \ell_{m+1}^2}{\sqrt{g_m}}$. The gradient for the case $j_m > 0$ for all m is:

$$\nabla F_S = 2 \sum_m \frac{1}{\sqrt{g_m}} [x - \bar{x}_m - \frac{1}{4} f_m\, d_m^\perp]$$

This gives rise to the iteration $T_S\, x = r_S$ with

$$T_S = \sum_m \frac{1}{\sqrt{g_m}} I$$

$$r_S = \sum_m \frac{1}{\sqrt{g_m}} [\bar{x}_m + \frac{1}{4} f_m\, d_m^\perp]$$

Again, this can only be used on untangled meshes - hence the need for Untangle.

Norm of G_m
Iteration $T_G\, x = r_G$ with

$$T_G = \sum_m \frac{2}{|\, G_m \,|} [2\, (\ell_m^2 + \ell_{m+1}^2)\, I - D_m]$$

$$r_G = \sum_m \frac{2}{|\, G_m \,|} [2\, (\ell_m^2 + \ell_{m+1}^2)\, \bar{x}_m + \Delta_m\, d_m^\perp]$$

Notice that the gradient for this functional is dimensionless.

Minimize the Maximum Eigenvalue of G_m

The iteration matrices are:

$$T_{MEV} = \sum_m 4\,(1 + \frac{\ell_m^2 + \ell_{m+1}^2}{\sqrt{(\ell_m^2 + \ell_{m+1}^2)^2 - 2\,g_m}})\,I - \frac{2\,D_m}{\sqrt{(\ell_m^2 + \ell_{m+1}^2)^2 - 2\,g_m}}$$

$$r_{MEV} = \sum_m 4\,(1 + \frac{\ell_m^2 + \ell_{m+1}^2}{\sqrt{(\ell_m^2 + \ell_{m+1}^2)^2 - 2\,g_m}})\,\bar{x}_m - \frac{2\,\Delta_m\,d_m^{\perp}}{\sqrt{(\ell_m^2 + \ell_{m+1}^2)^2 - 2\,g_m}}$$