Hexahedral Mesh by Area Functional

Sanjay Kumar Khattri[*1] and **Gunnar Fladmark** [1]

[1] Department of Mathematics
University of Bergen, Norway

Received 15 November 2004, revised 30 November 2004, accepted 2 December 2004

Key words Hexahedral Mesh, Area Functional, Hessian, Jacobian, Newton.
Subject classification 04A25

The purpose of this article is to present the area functional for the hexahedral mesh generation and optimization, and the Newton's optimization algorithm for finding the critical point of the functional.

1 Introduction

Functional optimization is extensively used for generating quality meshes in 2D and 3D [3, 4, 5], and references their in. In 3D hexahedral meshes are preferred over other meshes like tetrahedral , prism, pyramid, etc. But unfortunately generating quality hexahedral meshes is a difficult topic or at least not very well addressed. There is no algorithm which guareentee generating convex hexahedral mesh [6]. In this paper we will use area functional whose optimization will result in an improved hexahedral mesh. Though the functional presented can be applied to unstructured hexahedral meshes as well. We will only present the structured hexahedral mesh. In algebraic mesh optimization methods, a functional of the inner nodes is designed and it is expected that its minimum be attained in a mesh with the desirable geometrical properties (convexity,linearity,orthogonality,$\cdots$). This kind of functionals are called discrete or algebraic functionals. The first studies of this kind were done by Castillo and Steinberg[1]. Castillo and Steinberg introduced Length, Orthogonality and Area functionals. Area functional are extensively used for generating convex quadrilateral meshes in 2D [2, 7]. In 2D area functional are well known to produce superior 2D meshes compared to other functionals like length, orthogonality etc [2] and references their-in.

The article is arranged as follows in the next section area functional is presented, and we will explain its formulation for structured hexahedral meshes and in section three Newton's optimization algorithm for finding the critical point of the functional is explained, and section four presents numerical examples.

2 Area Functional for Hexahedral Meshes

First we will present the formulation of area functional for a $2 \times 2 \times 2$ structured hexahedral mesh as shown in the Figure 1. This mesh consists only one internal node, and it is shared by eight cells. The area functional for the node 0 (nodal area functional) shown in the Figure 1 can be defined by the equation (1). In the equation (1) the summation is over the 8 surrounding cells. The area functional (1) is the function of the physical coordinate $(\mathbf{x_0}, \mathbf{y_0}, \mathbf{z_0})$ of the node 0. In the equation (1) J_j^0 is the Jacobian (determinant of the Jacobian matrix) at the node 0 for the cell j. Table 1 list all the eight Jacobian matrix (matrix of covariant vectors) for the node 0. Critical point of the functional ($\nabla F_0 = 0$) will provide the proper nodal position for the node 0. Paper [2] mention important properties of the $2D$ area functional.

$$F_0(\mathbf{x_0}, \mathbf{y_0}, \mathbf{z_0}) = \sum_{j=1}^{8} (J_j^0)^2. \qquad (1)$$

* Corresponding author: e-mail: sanjay@mi.uib.no, Phone: +00 47 555 848 71, Fax: +00 47 555 896 72

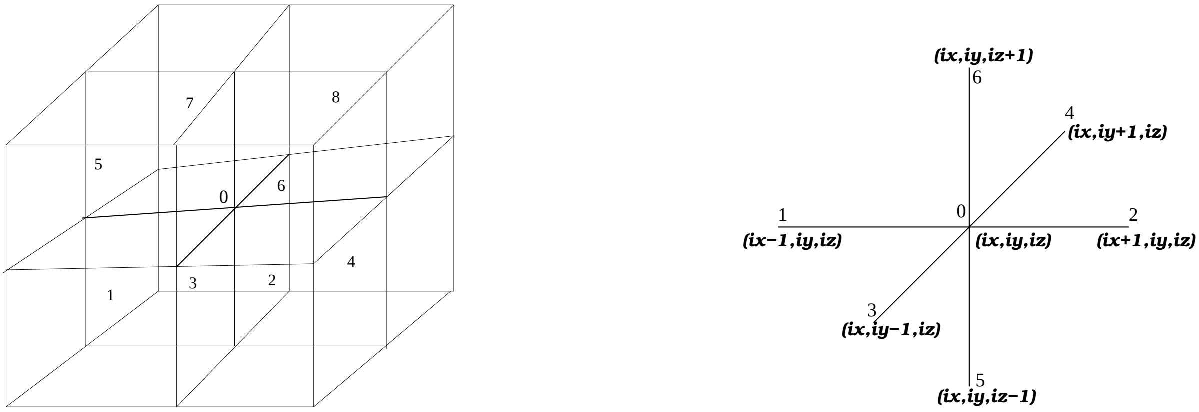

Fig. 1 Left figure shows the 8 cells sharing the node 0 and right figure shows the 8 nodes connected to the node 0.

Table 1 Jacobian matrix at the node 0 for all the 8 surrounding cells.

$$J_0^1 = \begin{bmatrix} (x_0 - x_1) & (x_0 - x_3) & (x_0 - x_5) \\ (y_0 - y_1) & (y_0 - y_3) & (y_0 - y_5) \\ (z_0 - z_1) & (z_0 - z_3) & (z_0 - z_5) \end{bmatrix} \quad J_0^2 = \begin{bmatrix} (x_0 - x_2) & (x_0 - x_3) & (x_0 - x_5) \\ (y_0 - y_2) & (y_0 - y_3) & (y_0 - y_5) \\ (z_0 - z_2) & (z_0 - z_3) & (z_0 - z_5) \end{bmatrix}$$

$$J_0^3 = \begin{bmatrix} (x_0 - x_1) & (x_0 - x_4) & (x_0 - x_5) \\ (y_0 - y_1) & (y_0 - y_4) & (y_0 - y_5) \\ (z_0 - z_1) & (z_0 - z_4) & (z_0 - z_5) \end{bmatrix} \quad J_0^4 = \begin{bmatrix} (x_0 - x_2) & (x_0 - x_4) & (x_0 - x_5) \\ (y_0 - y_2) & (y_0 - y_4) & (y_0 - y_5) \\ (z_0 - z_2) & (z_0 - z_4) & (z_0 - z_5) \end{bmatrix}$$

$$J_0^5 = \begin{bmatrix} (x_0 - x_1) & (x_0 - x_3) & (x_0 - x_6) \\ (y_0 - y_1) & (y_0 - y_3) & (y_0 - y_6) \\ (z_0 - z_1) & (z_0 - z_3) & (z_0 - z_6) \end{bmatrix} \quad J_0^6 = \begin{bmatrix} (x_0 - x_2) & (x_0 - x_3) & (x_0 - x_6) \\ (y_0 - y_2) & (y_0 - y_3) & (y_0 - y_6) \\ (z_0 - z_2) & (z_0 - z_3) & (z_0 - z_6) \end{bmatrix}$$

$$J_0^7 = \begin{bmatrix} (x_0 - x_1) & (x_0 - x_4) & (x_0 - x_6) \\ (y_0 - y_1) & (y_0 - y_4) & (y_0 - y_6) \\ (z_0 - z_1) & (z_0 - z_4) & (z_0 - z_6) \end{bmatrix} \quad J_0^8 = \begin{bmatrix} (x_0 - x_2) & (x_0 - x_4) & (x_0 - x_6) \\ (y_0 - y_2) & (y_0 - y_4) & (y_0 - y_6) \\ (z_0 - z_2) & (z_0 - z_4) & (z_0 - z_6) \end{bmatrix}$$

Let a hexahedral mesh consists of n internal nodes and let the node i is surrounder by 8 hexahedras (structured mesh is assumed, each internal node is surrounded by 8 hexahedras). The area functional for the whole mesh (global area functional) can be defined by the equation (2). The functional (2) is a function of the physical coordinates $(\mathbf{x}, \mathbf{y}, \mathbf{z})$ of the internal nodes. In the equation (2) J_j^i is the Jacobian at the node i for the cell j. It should be noted that the global area functional is the sum of the nodal area functionals. Thus instead of performing the global optimization, we can perform the nodal optimization. In this way we can utilise the most recently updated nodal position, we do not have to form and store global Hessian, implementation of Newton's minimization algorithm is simple, one very important benefit for mega meshes is that we can locally (part of the domain where solution is highly sensitive to mesh distortion) optimize the hexahedral mesh instead of optimizing the whole mesh which can be very costly or undesirable.

$$F(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \sum_{i=1}^{n} \left[\sum_{j=1}^{8} (J_j^i)^2 \right]. \tag{2}$$

3 Newton's Optimization Algorithm

In this section first we will develop the Newton's optimization procedure for the nodal area functional (1), and then provide an algorithm for applying it to the whole mesh. The nodal area functional (1) is function of its own physical coordinate ($\mathbf{r} = (x, y, z)$). Let the initial position for a node is $\mathbf{r}_0(x_0, y_0, z_0)$. Expanding the area functional for this node by Taylors series.

$$f(\mathbf{r}) = f(\mathbf{r_0}) + \nabla f(\mathbf{r_0}) \cdot (\mathbf{r} - \mathbf{r_0}) + \frac{1}{2}(\mathbf{r} - \mathbf{r_0}) \cdot \mathbf{H}(\mathbf{r_0}) \cdot (\mathbf{r} - \mathbf{r_0}) + HOT, \tag{3}$$

In the equation (3) $\nabla f(\mathbf{r_0})$ is the gradient row vector $(dimension = 3 \times 1)$ of the functional at the initial position $\mathbf{r}_0$, and $\mathbf{H}(\mathbf{r_0})$ is the Hessian matrix $(dimension = 3 \times 3)$ at the initial position $\mathbf{r}_0$. Since the gradient of the nodal area functional is also the function of the physical coordinate the node ($\mathbf{r} = (x, y, z)$). Expanding the gradient of the nodal area functional by Taylors series.

$$\nabla f(\mathbf{r}) = \nabla f(\mathbf{r_0}) + \mathbf{H}(\mathbf{r_0}) \cdot (\mathbf{r} - \mathbf{r_0}) + HOT, \tag{4}$$

We are interested in such a nodal position $\mathbf{r}$ where the gradient $\nabla f(\mathbf{r})$ of the nodal area functional vanish (critical point). Setting equation (4) equal to zero and neglecting higher order terms will result in a linear system read as follows.

$$\mathbf{H}(\mathbf{r_0}) \cdot \triangle\mathbf{r} = -\nabla f(\mathbf{r}). \tag{5}$$

Equation (5) is the basis for the Newton's minimization algorithm. The system (5) resulting from nodal area functional is 3×3. The system (5) resulting from the global area functional of a mesh consisting n internal nodes will be $3n \times 3n$. Which can be a huge system for mega meshes, and optimization consists of solving many such linear systems.

Algorithm 1 presents the pseudo code for the overall algorithm. In the Algorithm 1 the sub-script i denotes the node number, and super-script k denotes the Newton iteration. The algorithm will take advantage of the most recently calculated nodal positions.

Algorithm 1: Newton's Minimization Algorithm

```
while (iter ≤ max_iter1 || resid ≤ tol ) do
    forall (Internal Nodes i in the Mesh) do
        k = 0
        while (k ≤ max_iter2 || (||△r_i||_L2 ≤ tol & ||∇f(r^k)||_L2 ≤ tol)) do
            H(r^k) △r= -∇f(r^k)
            r_i^(k+1) = r_i^k + △r
            k^++
        end
    end
    resid = ||new_mesh − old_mesh||_L2
    iter^++
end
```

In the algorithm (1) $\|\cdot\|_{L2}$ denotes the discrete L_2 norm, max_{iter1} denotes the maximum iteration for the global mesh, $resid = \|new_{mesh} - old_{mesh}\|_{L_2}$ denotes the L_2 difference between the two global mesh, max_{iter2} denotes the maximum number of Newton's iteration, tol is the error tolerance which can have different value at different places. It it interesting to note the stopping criteria for the Newtons's iteration. We are not only using the gradient of the functional ($\nabla f(\mathbf{r})$) but also the difference nodal position vector ($\triangle\mathbf{r}$). We have implemented the above algorithm in C^{++}.

4 Numerical Examples

We performed two numerical experiments, and results are reported in the Figure 3.

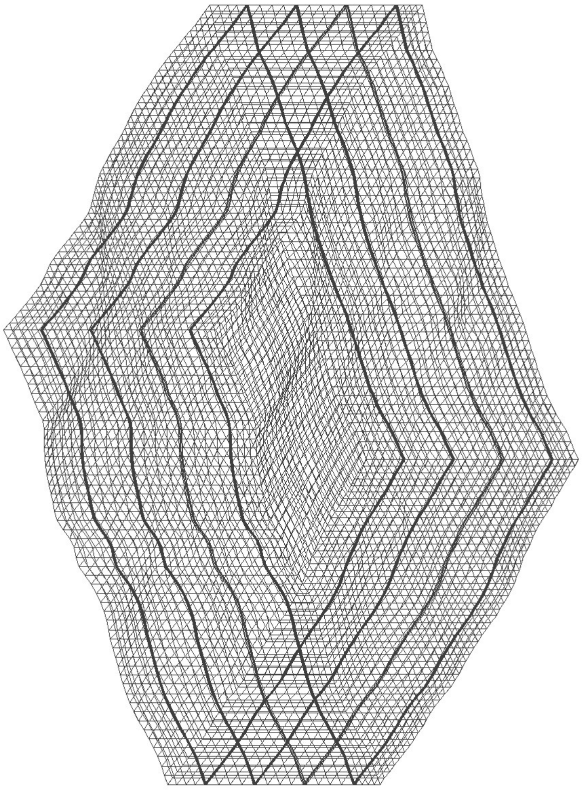

Fig. 2 Hexahedral Mesh Consisting more than 50000 nodes

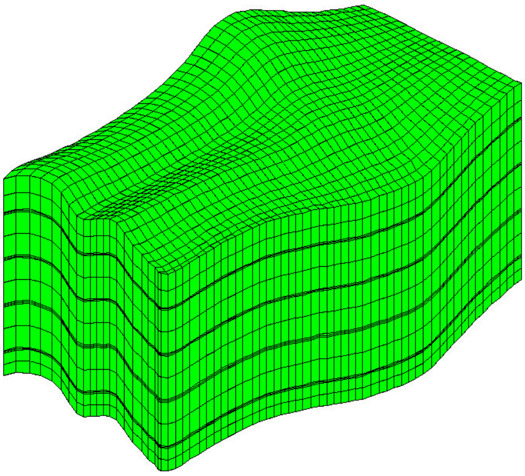

Fig. 3 Hexahedral Mesh Consisting more than 20000 nodes

References

[1] J. E. Castillo. On Variational Grid Generation, *Ph. D. Thesis, The University of New Mexico, Albuquerque, New Mexico*, 1987.

[2] T. Ruiz, J. Gerardo, P. B. Sanchez and A. C. Medina. Some Properties of Area Functionals in Numerical Grid Generation, *Proceedings, 10th International Meshing Roundtable, Sandia National Laboratories*, 43–54, 2001.

[3] P. M. Knupp and Nicolas Robidoux. A Framework for Variational Grid Generation: Conditioning the Jacobian Matrix with Matrix Norms, *Society for Industrial and Applied Mathematics*, 2029–2047, 2000.

[4] P. M. Knupp. Achieving Finite Element Mesh Quality via Optimization of the Jacobian Matrix Norm and Associated Quantities. Part II " A Framework for Volume Mesh Optimization", *International Journal for Numerical Methods in Engineering, John Wiley & Sons, Ltd*, Vol. 48, 1165–1185, 2000.

[5] P. M. Knupp. "Achieving Finite Element Mesh Quality via Optimization of the Jacobian Matrix Norm and Associated Quantities. Part I " A Framework for Surface Mesh Optimization,", *International Journal for Numerical Methods in Engineering, John Wiley & Sons, Ltd*, Vol. 48, 401-420, 2000.

[6] P. M. Knupp. Hexahedral Mesh Untangling and Algebraic Mesh Quality Metrics, *Proceedings, 9th International Meshing Roundtable, Sandia National Laboratories.* Vol. 9, 173-183, 2002.

[7] S. K. Khattri. Quadrilateral Mesh Adaptation by Area Functional, *Third MIT Conference*, 2005.