Grid generation and adaptation by functionals

SANJAY KUMAR KHATTRI

Stord Haugesund University College, Norway
E-mail: sanjay.khattri@hsh.no
http://ans.hsh.no/home/skk/

Abstract. Accuracy of a simulation is strongly depend on the grid quality. Here, quality means orthogonality at the boundaries and quasi-orthogonality within the critical regions, smoothness, bounded aspect ratios, solution adaptive behavior, etc. We review various functionals for generating high quality structured quadrilateral meshes in two dimensional domains. Analysis of Winslow and Modified Liao functionals are presented. Numerical examples are also presented to support our theoretical analysis. We will demonstrate the use of the Area functional for generating adaptive quadrilateral meshes.

1 Introduction

Accuracy of numerical solutions of partial differential equations on a grid is very much depend on the quality of the underlying grid. There are various parameters for measuring grid quality. For example, orthogonality of grid lines and grid density in the regions of large solution gradients. A desired grid may be an orthogonal grid with high grid density in the areas of sharp solution gradients. Variational methods has been used for improving quality of a given grid [1]. In the variational methods, a grid functional is defined. Grid functional is an algebraic expression of the position vectors of the internal nodes of a mesh. Optimization of the grid functional may result in a grid with desired properties

such as orthogonal grid lines, equal cell areas, linear or parallelogram cells [see 2] and untangled mesh [4, 5, 6]. There are many algebraic functionals for grid generation and optimization [cf. 2, 4, 5, 6, 7, 8, 9]. The first study of grid generation by algebraic functionals were done in [10]. Castillo and Steinberg introduced Length, Orthogonality and Area functionals [10]. Area functional are well known for producing robust quadrilateral meshes. For a detailed description of properties of area functionals, the interested readers are referred to [9]. Recently the area functional has been used for generating adaptive quadrilateral meshes [12].

Let $x(\xi, \eta)$ and $y(\xi, \eta)$ be the coordinates of a node in a mesh. Let us further assume that x and y are twice differentiable functions of the independent variables ξ and η. An integral functional $\mathcal{I}$ can be defined as follows

$$\mathcal{I}(x, y) \stackrel{\text{def}}{=} \int_{[0,1]\times[0,1]} \mathcal{F}(\xi, \eta, x, y, x_\xi, x_\eta, y_\xi, y_\eta) \, d\xi \, d\eta \,. \tag{1}$$

We are interested in finding the functions $x(\xi, \eta)$ and $y(\xi, \eta)$ for which the integral functional $\mathcal{I}$ attains an extremal value. Such coordinates x and y define a mesh with desirable properties. The integral functional $\mathcal{I}$ is also referred to as control function for adaptive grid generation [1]. The conditions for the extremal value of the integral functional $\mathcal{I}$ are expressed by the Euler-Lagrange equations. The two Euler-Lagrange equations are

$$\frac{\partial \mathcal{F}}{\partial x} - \frac{\partial}{\partial \xi}\left(\frac{\partial \mathcal{F}}{\partial x_\xi}\right) - \frac{\partial}{\partial \eta}\left(\frac{\partial \mathcal{F}}{\partial x_\eta}\right) = 0 \,, \tag{2}$$

$$\frac{\partial \mathcal{F}}{\partial y} - \frac{\partial}{\partial \xi}\left(\frac{\partial \mathcal{F}}{\partial y_\xi}\right) - \frac{\partial}{\partial \eta}\left(\frac{\partial \mathcal{F}}{\partial y_\eta}\right) = 0 \,. \tag{3}$$

The functions x and y, which satisfy the above Euler-Lagrangian equations, are called the extremal of the integral functional $\mathcal{I}$.

Let us define some quantities of interest. Figure 1 shows a quadrilateral cell, and this cell belongs to a mesh. Let the co-variant vector at the node o and in the direction oa is $\mathbf{g}_1$, and another co-variant vector at the node o but in the direction ob is $\mathbf{g}_2$. These vectors are given as

$$\mathbf{g}_1 = (x_a - x_o, y_a - y_o)^t \quad \text{and} \quad \mathbf{g}_2 = (x_b - x_o, y_b - y_o)^t \,. \tag{4}$$
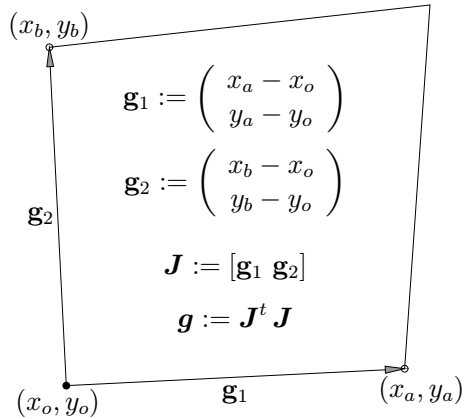
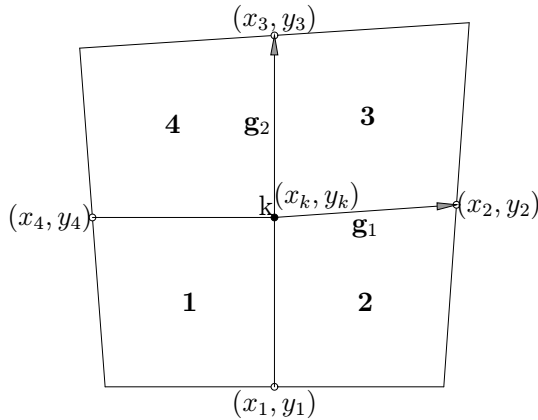Figure 1 – Quantities of interest for a quadrilateral cell.

Figure 2 – 2D Structured Mesh. Node k is surrounded by four quadrilaterals.

Other interesting quantities such as the Jacobian and g-tensor matrix can be defined from the co-variant vectors. The columns of the Jacobian matrix are the co-variant vectors. The g-tensor matrix is the product of the Jacobian matrix with it's transpose. Thus, the Jacobian matrix and the g-tensor at the node o and for the cell shown in the Figure 1 are given as

$$J = \begin{bmatrix} \mathbf{g}_1 & \mathbf{g}_2 \end{bmatrix} \quad \text{and} \quad g = J^t \, J \,. \tag{5}$$

The layout of the paper is as follows. In the Section 2, several functionals are presented. Continuous and discrete versions of the functionals are presented.

Section 3 presents several numerical experiments, and finally Section 4 concludes the paper.

2   Discrete Functionals

Let us first introduce some quantities of interest. These will be used later in formulating algebraic functionals. Figure 2 is a 2×2 structured mesh. We use this figure for defining these quantities.

$J(k_i)$ refers to the Jacobian (determinant of the Jacobian matrix $\boldsymbol{J}(k_i)$) at the node k and for the cell i. Table 1 lists the Jacobian matrix for the four cells surrounding the node k. $\mathbf{g}_1(k_i)$ refers to the co-variant base vector at the node k and for the cell i. The base vector $\mathbf{g}_1$ points along horizontal grid lines. Similarly, $\mathbf{g}_2(k_i)$ refers to the co-variant base vector at the node k and for the cell i, and it points along the vertical grid lines. Table 2 lists the co-variant vectors for the Figure 2. It should be noted that column vectors of the Jacobian matrix are the co-variant base vectors. For example, the column vectors of $\boldsymbol{J}(k_1)$ are $\mathbf{g_1}(k_1)$ and $\mathbf{g_2}(k_1)$. That is $\boldsymbol{J}(k_1) = \begin{bmatrix} \mathbf{g}_1(k_1) & \mathbf{g}_2(k_1) \end{bmatrix}$. $\boldsymbol{g}(k_i)$ refers to the co-variant metric tensor at the node k and for the cell i. It is defined as $\boldsymbol{g}(k_i) = \boldsymbol{J}(k_i)^t \, \boldsymbol{J}(k_i)$. $g_{mn}(k_i)$ refers to the (m, n) coefficient of the co-variant metric tensor $\boldsymbol{g}(k_i)$ for the node k and for the cell i. It can be seen that $g_{11}(k_i) = \mathbf{g}_1(k_i)^t \cdot \mathbf{g}_1(k_i)$ and $g_{12}(k_i) = \mathbf{g}_1(k_i)^t \cdot \mathbf{g}_2(k_i)$. Similarly, other coefficients can be defined. The coefficient g_{12} is a measure of the angle between the co-variant base vectors $\mathbf{g}_1$ and $\mathbf{g}_2$. While, the coefficient $\mathbf{g}_{11}$ is a measure of the discrete L_2 length of the co-variant vector $\mathbf{g}_1$.

$$
\boldsymbol{J}(k_1) = \begin{bmatrix} (x_4 - x_k) & (x_1 - x_k) \\ (y_4 - y_k) & (y_1 - y_k) \end{bmatrix} \quad \boldsymbol{J}(k_2) = \begin{bmatrix} (x_2 - x_k) & (x_1 - x_k) \\ (y_2 - y_k) & (y_1 - y_k) \end{bmatrix}
$$

$$
\boldsymbol{J}(k_3) = \begin{bmatrix} (x_2 - x_k) & (x_3 - x_k) \\ (y_2 - y_k) & (y_3 - y_k) \end{bmatrix} \quad \boldsymbol{J}(k_4) = \begin{bmatrix} (x_4 - x_k) & (x_3 - x_k) \\ (y_4 - y_k) & (y_3 - y_k) \end{bmatrix}
$$

Table 1 – Jacobian matrix at the node k for the surrounding cells for the Figure 2.

$$\mathbf{g}_1(k_1) = \begin{pmatrix} x_4 - x_k \\ y_4 - y_k \end{pmatrix} \quad \mathbf{g}_2(k_1) = \begin{pmatrix} x_1 - x_k \\ y_1 - y_k \end{pmatrix}$$

$$\mathbf{g}_1(k_2) = \begin{pmatrix} x_2 - x_k \\ y_2 - y_k \end{pmatrix} \quad \mathbf{g}_2(k_2) = \begin{pmatrix} x_1 - x_k \\ y_1 - y_k \end{pmatrix}$$

$$\mathbf{g}_1(k_3) = \begin{pmatrix} x_2 - x_k \\ y_2 - y_k \end{pmatrix} \quad \mathbf{g}_2(k_3) = \begin{pmatrix} x_3 - x_k \\ y_3 - y_k \end{pmatrix}$$

$$\mathbf{g}_1(k_4) = \begin{pmatrix} x_4 - x_k \\ y_4 - y_k \end{pmatrix} \quad \mathbf{g}_2(k_4) = \begin{pmatrix} x_3 - x_k \\ y_3 - y_k \end{pmatrix}$$

Table 2 – Co-variant vectors at the node k for the surrounding cells for the Figure 2.

Let us consider a structured quadrilateral mesh (each internal node is surrounded by four quadrilaterals) consisting of n internal nodes. The following functionals can be defined

2.1 Area Functional

The integral form of the standard Area functional is given as

$$\mathcal{I}_A \stackrel{\text{def}}{=} \frac{1}{2} \int_{[0,1]\times[0,1]} |\boldsymbol{J}|^2 \, d\xi \, d\eta \,, \tag{6}$$

$$= \int_{[0,1]\times[0,1]} (x_\xi \, y_\eta - x_\eta \, y_\xi) \, d\xi \, d\eta \,. \tag{7}$$

The Euler-Lagrangian equations for the Area functional are

$$\frac{\partial}{\partial \xi} \left(|\boldsymbol{J}| \, x_\eta \right) - \frac{\partial}{\partial \eta} \left(|\boldsymbol{J}| \, x_\xi \right) = 0 \,, \tag{8}$$

$$\frac{\partial}{\partial \xi} \left(|\boldsymbol{J}| \, y_\eta \right) - \frac{\partial}{\partial \eta} \left(|\boldsymbol{J}| \, y_\xi \right) = 0 \,. \tag{9}$$

In the simplified form the above equations can be written as

$$y_\eta{}^2 x_{\xi\xi} - x_\eta y_\eta y_{\xi\xi} - 2.0 y_\xi y_\eta x_{\xi\eta}$$
$$+ (x_\xi y_\eta + x_\eta y_\xi) y_{\xi\eta} + y_\xi{}^2 x_{\eta\eta} - x_\xi y_\xi y_{\eta\eta} = 0, \tag{10}$$

$$x_\eta{}^2 y_{\xi\xi} - x_\eta y_\eta x_{\xi\xi} - 2.0 x_\xi x_\eta y_{\xi\eta}$$
$$+ (x_\xi y_\eta + x_\eta y_\xi) x_{\xi\eta} + x_\xi{}^2 y_{\eta\eta} - y_\xi y_\xi x_{\eta\eta} = 0, \tag{11}$$

[see 9]. The above Euler-Lagrangian equations are non-elliptic, coupled and quasi-linear [cf. 9]. For generating adaptive mesh, the author proposed the following variation in the Area functional

$$\mathcal{F}_A(\mathbf{x}, \mathbf{y}) = \sum_{k=1}^{n} \left[\sum_{i=1}^{4} s(k_i) \, [J(k_i)]^2 \right], \tag{12}$$

[12]. In the above equation, $s(k)$ is called the adaptive function, and $s(k_i)$ is the value of the adaptive function at the node k and for cell i.

2.2   Length Functional

The integral form of the Length functional is given as

$$\mathcal{I}_{\mathrm{L}} \stackrel{\text{def}}{=} \frac{1}{2} \int_{[0,1]\times[0,1]} [\, g_{11} + g_{22} \,] \, d\xi \, d\eta, \tag{13}$$

$$= \frac{1}{2} \int_{[0,1]\times[0,1]} \left[(x_\xi)^2 + (x_\eta)^2 + (y_\xi)^2 + (y_\eta)^2 \right] d\xi \, d\eta, \tag{14}$$

[7, 8, 9, and references therein]. The conditions of extremality of the above length functional are given by the following Euler-Lagrangian equations

$$\frac{\partial^2 x}{\partial \xi^2} + \frac{\partial^2 x}{\partial \eta^2} = 0, \tag{15}$$

$$\frac{\partial^2 y}{\partial \xi^2} + \frac{\partial^2 y}{\partial \eta^2} = 0. \tag{16}$$

The above Laplace's equations can be solved in the computational domain $[0, 1] \times [0, 1]$ with a specified value of x and y on the boundary. The Euler-Lagrangian equations associated with the Length functional are linear and uncoupled.

The discrete Length functional [1, 10] is give as follows

$$\mathcal{F}_{\text{L}}(\mathbf{x}, \mathbf{y}) = \sum_{k=1}^{n} \left[\sum_{i=1}^{4} (g_{11}(k_i) + g_{22}(k_i)) \right] \,. \tag{17}$$

2.3   Orthogonality Functional

The integral form of the Orthogonality functional [7, 8, 9, and references therein] is given as follows

$$\mathcal{I}_{\text{O}} \overset{\text{def}}{=} \frac{1}{2} \int_{[0,1]\times[0,1]} (g_{12})^2 \, d\xi \, d\eta \,, \tag{18}$$

$$= \frac{1}{2} \int_{[0,1]\times[0,1]} (\mathbf{g}_1 \cdot \mathbf{g}_2)^2 \, d\xi \, d\eta, \tag{19}$$

$$= \frac{1}{2} \int_{[0,1]\times[0,1]} (x_\xi \, x_\eta + y_\xi \, y_\eta)^2 \, d\xi \, d\eta \,. \tag{20}$$

The Euler-Lagrangian equations corresponding to the minimization of the above integral are

$$\frac{\partial}{\partial \xi} \left(g_{12} \frac{\partial x}{\partial \eta} \right) + \frac{\partial}{\partial \eta} \left(g_{12} \frac{\partial x}{\partial \xi} \right) = 0 \,, \tag{21}$$

$$\frac{\partial}{\partial \xi} \left(g_{12} \frac{\partial y}{\partial \eta} \right) + \frac{\partial}{\partial \eta} \left(g_{12} \frac{\partial y}{\partial \xi} \right) = 0 \,. \tag{22}$$

These Euler-Lagrangian equations are quasilinear, coupled and non-elliptic in nature [9]. A simplified form the above Euler-Lagrangian equations is

$$x_\eta^{\,2} \, x_{\xi\xi} + x_\eta \, y_\eta \, y_{\xi\xi} + (4 \, x_\xi \, x_\eta + 2 \, y_\xi \, y_\eta) \, x_{\xi\eta}$$
$$+ (x_\xi \, y_\eta + x_\eta \, y_\xi) \, y_{\xi\eta} + x_\xi^{\,2} \, x_{\eta\eta} + x_\xi \, y_\xi \, y_{\eta\eta} = 0 \,, \tag{23}$$

$$y_\eta^{\,2} \, y_{\xi\xi} + y_\eta \, x_\eta \, x_{\xi\xi} + (4 \, y_\xi \, y_\eta + 2 \, x_\xi \, x_\eta) \, y_{\xi\eta}$$
$$+ (y_\xi \, x_\eta + y_\eta \, x_\xi) \, x_{\xi\eta} + y_\xi^{\,2} \, y_{\eta\eta} + y_\xi \, x_\xi \, x_{\eta\eta} = 0 \,, \tag{24}$$

[see 9]. This functional takes only non-negative values, and it would attain a minimum value of zero for a completely orthogonal grid. The discrete version

of the above Orthogonality functional [1, 10] is given as follows

$$\mathcal{F}_O(\mathbf{x}, \mathbf{y}) = \sum_{k=1}^{n} \left[\sum_{i=1}^{4} (\mathbf{g_1}(\mathbf{k_i}) \cdot \mathbf{g_2}(\mathbf{k_i}))^2 \right]. \tag{25}$$

It is found [cf. 1, 7, 8, 10, 13, 14] that a linear combination of Area, Length and Orthogonality functionals can produce robust grids in complicated 2D domains.

2.4   Combination of Length, Area and Orthogonality Functionals

A combined functional is given as

$$\mathcal{F}(\mathbf{x}, \mathbf{y}) = \Bbbk_A\, \mathcal{F}_A(\mathbf{x}, \mathbf{y}) + \Bbbk_L\, \mathcal{F}_L(\mathbf{x}, \mathbf{y}) + \Bbbk_O\, \mathcal{F}_O(\mathbf{x}, \mathbf{y}), \tag{26}$$

[1, 7, 8, 10, 13, 14]. Here, the parameters $\Bbbk_A, \Bbbk_L$ and $\Bbbk_O$ satisfy: $\Bbbk_A + \Bbbk_L + \Bbbk_O = 1.0$ and $\Bbbk_A \geq 0$, $\Bbbk_L \geq 0$, $\Bbbk_O \geq 0$. A serious drawback of the above combined functional is a suitable choice of the parameters. It requires an experience in coming up with a good set of parameters [10]. It was found [13, 14] that the following choice of parameters

$$\Bbbk_A = 0.50, \quad \Bbbk_L = 0.0, \quad \text{and} \quad \Bbbk_O = 0.50, \tag{27}$$

produces robust grid in many practical domains. The corresponding functional is referred as the Knupp's functional. Presented numerical work shows that this functional can produce good grids. The Euler-Lagrangian [10] equations for the minimization of the Knupp's functional are

$$(x_\eta{}^2 + y_\eta{}^2)\, x_{\xi\xi} + 4\, x_\xi\, x_\eta\, x_{\xi\eta}$$
$$+ 2(x_\xi\, y_\eta + x_\eta\, y_\xi)\, y_{\xi\eta} + (x_\xi{}^2 + y_\xi{}^2)\, x_{\eta\eta} = 0, \tag{28}$$

$$(x_\eta{}^2 + y_\eta{}^2)\, y_{\xi\xi} + 4\, y_\xi\, y_\eta\, y_{\xi\eta}$$
$$+ 2(x_\xi\, y_\eta + x_\eta\, y_\xi)\, x_{\xi\eta} + (x_\xi{}^2 + y_\xi{}^2)\, y_{\eta\eta} = 0. \tag{29}$$

2.5   Winslow Functional

The Winslow functional is given as follows

$$\mathcal{F}(\mathbf{x}, \mathbf{y}) = \sum_{k=1}^{n} \left[\sum_{i=1}^{4} \left(\frac{g_{11}(k_i) + g_{22}(k_i)}{|\boldsymbol{J}(k_i)|} \right) \right], \tag{30}$$

[1, 3, 16]. Here, $|\boldsymbol{J}(k_i)|$ is the determinant of the Jacobian matrix. One very important feature of the above functional is that it has barrier. It means the value of the functional approaches infinity when the cells degenerate. That is $|\boldsymbol{J}| \rightarrow 0$. Thus, this functional produces unfolded grids. Numerical experiments also prove this feature of the Winlow functional. Since $g_{11} = \mathbf{g}_1 \cdot \mathbf{g}_1$, and $g_{22} = \mathbf{g}_2 \cdot \mathbf{g}_2$. It can be shown that the numerator $(g_{11} + g_{22})$ in the Winslow functional (30) is the Frobenius norm of the Jacobian matrix. That is

$$g_{11}(k_i) + g_{22}(k_i) = \sum_{n=1}^{2} \sum_{m=1}^{2} (J_{mn}(k_i))^2 = (\|\mathbf{J}(\mathbf{k_i})\|)^2 ,$$

Here, J_{mn} are the components of the Jacobian matrix $\boldsymbol{J}$. Thus, the Winslow functional 30 can be written as follows

$$\mathcal{F}(\mathbf{x}, \mathbf{y}) = \sum_{k=1}^{n} \left[\sum_{i=1}^{4} \frac{\|\boldsymbol{J}(k_i)\|^2}{|\boldsymbol{J}(k_i)|} \right]. \tag{31}$$

It can be seen easily that the Frobenius norm a 2×2 matrix $\boldsymbol{A}$, and its inverse are related as $\|A^{-1}\| = \dfrac{\|A\|}{|A|}$. Here, $|A|$ is the determinant of the matrix $\boldsymbol{A}$. The condition number $\mathsf{K}(A)$ of a matrix $\boldsymbol{A}$ can be written as $\mathsf{K}(A) = \|A\| \, \|A^{-1}\|$. Here, the norm is the Frobenius norm. Thus, the Winslow functional can be written as follows

$$\mathcal{F}(\mathbf{x}, \mathbf{y}) = \sum_{k=1}^{n} \left[\sum_{i=1}^{4} \mathsf{K}(\boldsymbol{J}(k_i)) \right]. \tag{32}$$

Thus, the minimization of the functional (30) is equivalent to the minimization of the condition number of the Jacobian matrix. A detailed description of the above analysis can also be found in [4, 5, 6, 16]. The condition number $\mathsf{K}(\boldsymbol{J}(k_i))$ can also be expressed as

$$\mathsf{K}(\boldsymbol{J}(k_i)) = \frac{\mathbf{g_1}(k_i)^2 + \mathbf{g_2}(k_i)^2}{|\mathbf{g_1}(k_i) \times \mathbf{g_2}(k_i)|} .$$

The $\boldsymbol{g}(k_i)$ tensor matrix is give as

$$\boldsymbol{g}(k_i) = \begin{bmatrix} g_{11}(k_i) & g_{12}(k_i) \\ g_{21}(k_i) & g_{22}(k_i) \end{bmatrix}.$$

Let λ_1 and λ_2 be the eigenvalues of the matrix $\boldsymbol{g}(k_i)$. Then

$$\frac{g_{11}(k_i) + g_{22}(k_i)}{|\boldsymbol{J}(k_i)|} = \frac{\lambda_1 + \lambda_2}{\sqrt{\lambda_1 \lambda_2}} \geq 2.0 \,.$$

Here, we have used the relation $|\boldsymbol{J}|^2 = |\boldsymbol{g}|$. Thus, the Winslow functional is bounded from below.

2.6 Liao Functional

The Liao functional for grid generation was proposed in [11], and is give as follows

$$\mathcal{F}(\mathbf{x}, \mathbf{y}) = \sum_{k=1}^{n} \left[\sum_{i=1}^{4} (g_{11}{}^2 + g_{22}{}^2 + 2\,g_{12}{}^2) \right] \,. \tag{33}$$

2.7 Modified Liao Functional

The Liao functional can produce folded grids. We will explore it through numerical experiments. In the literature, following modification [1] of the Liao functional is given

$$\mathcal{F}(\mathbf{x}, \mathbf{y}) = \sum_{k=1}^{n} \left[\sum_{i=1}^{4} \left(\frac{g_{11}(k_i) + g_{22}(k_i)}{\sqrt{g(k_i)}} \right)^2 \right] \,. \tag{34}$$

In the above equation, $g(k_i)$ is the determinant of the covariant metric tensor $\boldsymbol{g}(k_i)$. It can be shown that $g = J^2$, where J is the Jacobian (determinant of the Jacobian matrix), and g is the determinant of the co-variant metric tensor. Thus, this functional, similar to the Winslow functional (30), has a barrier. The value of the functional approaches infinity when the cells degenerate. That is $|\boldsymbol{J}| \to 0$. Thus, this functional produces unfolded grids. Numerical experiments also prove this feature of the functional. The above functional can remove the folded grids produced by the Liao functional. The Modified Liao functional can also be written as follows

$$\mathcal{F}(\mathbf{x}, \mathbf{y}) = \sum_{k=1}^{n} \left[\sum_{i=1}^{4} (\mathsf{K}(\boldsymbol{J}(k_i)))^2 \right] \,. \tag{35}$$

The Modified Liao functional minimizes the square of the condition number where as the Winslow functional minimizes the condition number.

3   Numerical examples

We are interested in finding such a mesh for which the gradient of the functionals vanish. The minimization of functionals can be performed by a line search algorithms such as Newton's iteration. For the numerical experiments instead of performing the global optimization we solved the local minimization problems for a single node at a time [15]. In all numerical examples, initial grid was generated by linear transfinite interpolation [1].

3.1   Adaptive Grid by Area Functional

It is generally not recommended to uniformly refine the whole mesh in the hope of capturing the underlying physics. It is desired to adapt a given grid to the requirement of the underlying problem. A grid generating algorithm should be able to allocate more grid nodes in the part of the domain where large solution gradients occur, and fewer grid nodes in the part of the domain where solution is flat. Such grids are called solution-adaptive. Behavior of the underlying solution can be obtained by posteriori indicators [17]. These indicators can be computed on a coarse mesh, and they can be used to formulate adaptive function $s(x, y)$ in the equation (12). In the present work, the adaptive function $s(x, y)$ is given in the analytic form.

Figures 3 and 4 report the outcome of our numerical experiments. It should be noted that even after adaptation the quadrilateral meshes are convex. One other advantage of mesh adaptation by Area functional is that it preserves the mesh topology, and writing a solver for a structured mesh is easier compared to unstructured mesh.

3.2   Winslow Functional vs Algebraic Method

Algebraic grid generation methods such as transfinite interpolations [1] are exten-sively used for generating grids. Though, they are one of most simplest method of grid generation but algebraic methods can produce folded grids for curved domains as can be seen in the Figure 5. One other disadvantage of algebraic grid generation is that boundary discontinuity can prorogate inside the domain. It is clear from Figure 6 that Winslow functional smooth the grid, and removes the folded grid lines.
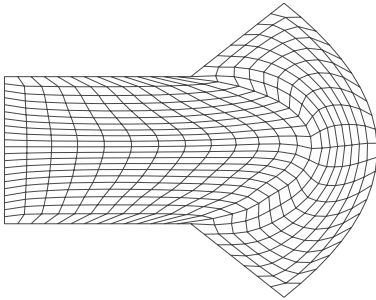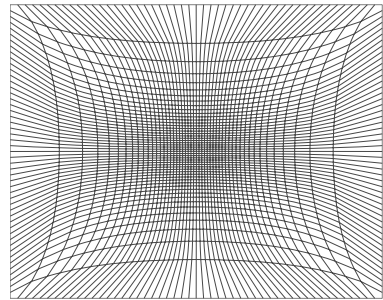
Figure 3

Figure 4

Figure 3 – Example (3.1): Adapted Grid by Area Functional. Adapted Functional is give as $s(x, y) = 5.0 + 200.0 |\sin(2\pi x) \sin(2\pi y)|$. Figure 4 – Example (3.1): Adapted Grid by Area Functional. Adapted Functional is give as $s(x, y) = 5.0 + 200.0 [\sin(2\pi x) \sin(2\pi y)]$.
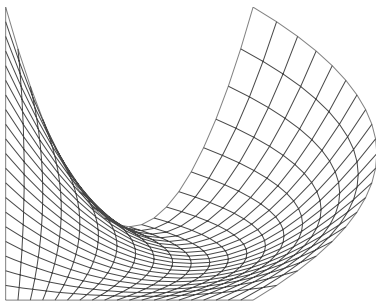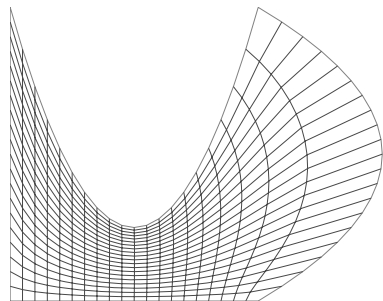

Figure 5

Figure 6

Figure 5 – Example (3.2): Folded Grid by Transfinite Interpolation. Figure 6 – Example (3.2): Smooth Grid by Winslow Functional.

3.3   Liao, Modified Liao and Area Functionals

In this example, we perform experiments for comparing Liao, Modified and Area functional on a simple domain. Outcome of our results are shown in Figures 10, 11 and 12. It can be seen from these figures that Modified Liao functional does indeed removes the inverted elements from the mesh but still the quality of the mesh generated by the area functional shown in the Figure 11 is certainly better than both Liao and Modified Liao.
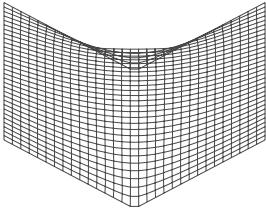
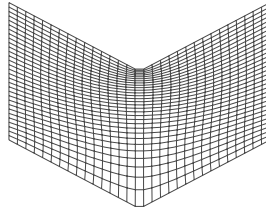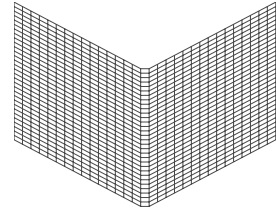Figure 7   Figure 8   Figure 9

Figure 7 – Example (3.3): Folded Grid by the Liao Functional. Figure 8 – Example
(3.3): Grid by the Modified Liao Functional. Figure 9 – Example (3.3): Grid by the
Area Functional with $s(x, y) = 1.0$.
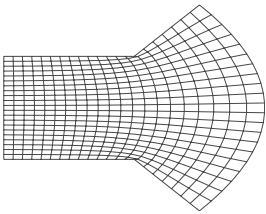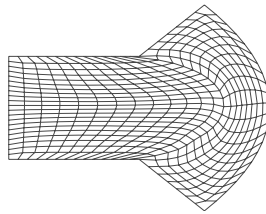

Figure 10   Figure 11   Figure 12
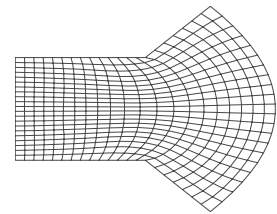
Figure 10 – Example (3.4): Grid by the Length functional. Figure 11 – Example (3.4):
Grid by the Area Functional with $s(x, y) = 1.0$. Figure 12 – Example (3.4): Grid by
the Knupp's functional.

3.4   Length, Area and Knupp's Functionals

In this example, we compare the Length, the Area and the Knupp functionals.
Figures 10, 11 and 12 are the outcome of our numerical work. The Figure 10 is
a grid by the Length functional, the Figure 11 is a grid by the Area functional,
and the Figure 12 is a grid by the Knupp's functional. It can be seen that grid by
the Area and Knupp's functional are better than the grid produced by the Length
functional.

3.5   Length and Knupp's Functionals

In this example, we are comparing Length, and the Knupp's Functional. Outcome
of our numerical work is reported in Figures 13 and 14. Figure 13 is a grid
generated by the Length functional. Figure 14 is grid generated by the Knupp's

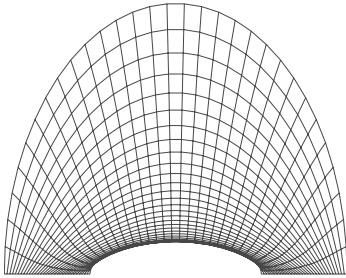functional. It can be seen that the grid generated by the Knupp's functional is of superior quality.
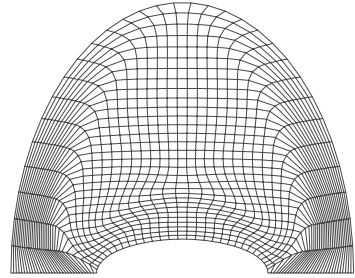

Figure 13

Figure 14

Figure 13 – Example (3.5): Grid by the Length Functional. Figure 14 – Example (3.5): Grid by the Knupp's Functional.

4   Conclusions

We have presented the formulation of various functionals for generating quadrilateral meshes, and an analysis of Winslow and Modified Liao functionals that is consistent with the numerical experiments. Numerical experiments show that Winslow and Modified Liao functionals can remove the folded grids as was expected from theoretical analysis. It has been shown that Area functionals can be used for generating robust adaptive meshes. Further research is required in formulating adaptive function from a posteriori error estimators.

REFERENCES

[1] J.F. Thompson, B.K. Soni and N.P. Weatherill, Handbook of Grid Generation. CRC Press, 1998.

[2] S.K. Khattri, A New Smoothing Algorithm for Quadrilateral and Hexahedral Meshes, Lecture Notes in Computer Science, Volume 3992, Apr 2006, Pages 239–246, DOI 10.1007/ 11758525_32, URL http://dx.doi.org/10.1007/11758525_32.

[3] A.M. Winslow, Equipotential zoning of two dimensional meshes. J. Computat. Physics, **1** (1967).

[4] P.M. Knupp, Jacobian-Weighted Elliptic Grid Generation. SIAM J. Sci. Comput., **17**(6) (1996), 1475–1490.

[5] P.M. Knupp, Algebraic mesh quality metrics. SIAM J. Sci. Comput., **23** (2001), 193–218.

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

[7] J.G. Tinoco-Ruiz and P. Barrera-Sánchez, Area control in generating smooth and convex grids over general plane regions. J. Comput. Appl. Math., **103** (1999), 19–32.

[8] J.G. Tinoco-Ruiz and P. Barrera-Sánchez, Smooth and convex grid generation over general plane regions. Math. Comput. Simulation, **46** (1998), 87–102.

[9] J.G. Tinoco, P. Barrera and A. Cortés, Some Properties of Area Functionals in Numerical Grid Generation. X Meshing RoundTable, Newport Beach, California, USA. 2001.

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

[11] G. Liao and H. Liu, Existence and $C^{(0,\alpha)}$ regularity of a minimum of a functional related to grid generation problems. Num. Math. PDEs, **9** (1993).

[12] S.K. Khattri, Analysing finite volume for single phase flow in porous media. Journal of Porous Medium, **10** (2007).

[13] P.M. Knupp and S. Steinberg, Fundamentals of Grid Generation. CRC Press. Boca Raton, Ann Arbor, London, Tokyo, 1993.

[14] P.M. Knupp, A robust elliptic grid generator. J. Comp. Phys., **100** (1992), 409–418.

[15] S.K. Khattri, An effective quadrilateral mesh adaptation. Journal of Zhejiang University Science A., **7** (2007), 2018–2021.

[16] P.M. Knupp, L. Margolin and M. Shashkov, Reference Jacobian Optimization-Based Rezone Strategies for Arbitrary Lagrangian Eulerian Methods. Report No. LA-UR-01-1194. On line avaibale report at "http://cnls.lanl.gov/ shashkov/papers/report.ps".

[17] S.K. Khattri, Analysing an Adaptive Finite Volume for Flow in Highly Heerogenous Porous Medium. Accepted for publication in the International Journal of Numerical Methods for Heat & Fluid Flow. Available at http://www.mi.uib.no/∼sanjay/publicatins.html.