Lecture 3.2
Methods for Structured Mesh Generation
• There are several methods to develop the structured meshes: Algebraic methods, Interpolation methods, and methods based on solving partial differential equations.

• In algebraic methods, the transformation is done analytically when the boundaries are rather simple and regular.

• For irregular geometries, the coordinates of the mesh (interior) nodes are obtained by numerical interpolation between the prescribed boundary data. One such method is transfinite interpolation.

• PDE methods are classified as elliptic, parabolic or hyperbolic, depending on the characteristics of the grid generation equations, which are generally transformed onto a rectangular domain and solved along with the governing equations of the problem.
Algebraic Mapping

• Consider the simply connected domain shown in Fig. 3.2.1 whose sides $AB$, $BC$, $CD$ and $DA$ are given by the equations $f_1(x, y) = 0$, $f_2(x, y) = 0$, $f_3(x, y) = 0$ and $f_4(x, y) = 0$, respectively.

• Without loss of generality, one can map the curves $AB$ and $CD$ onto the lines $\eta = 0$ and $\eta = 1.0$, using a transformation of the form

$$\eta = f_1(x, y)/\{f_1(x, y) - f_3(x, y)\}$$

• Similarly it can be assumed that

$$\xi = f_4(x, y)/\{f_4(x, y) - f_2(x, y)\}$$
Fig. 3.2.1 Algebraic mapping (a) Physical domain in x-y system with regular boundaries (b) Rectangular body in the (ξ-η) transformed plane.
In order to apply the algebraic mapping technique for a domain between the suction surface AB and pressure surface CD of a turbomachine blade cascade (Fig. 3.2.2), one may first approximate the lines AB and CD in terms of analytical functions: $f_1$ and $f_3$ and follow the other steps as suggested earlier.

![Diagram](image)

Fig. 3.2.2 Algebraic mapping of a simple blade cascade
Transfinite Interpolation

- Apply unidirectional interpolation in $\xi$ – direction (or $\eta$ – direction) between the boundary grid data given on the curves $\xi = 0$ and $\xi = 1$ (or $\eta = 0$ and $\eta = 1$) and obtain the coordinates $x'_p, y'_p$ for every interior and boundary point.

- Calculate the mismatch between the interpolated and the actual coordinates on the $\eta = 0$ and $\eta = 1$ (or $\xi = 0$ and $\xi = 1$) boundaries.

- Linearly interpolate the difference in the boundary point coordinates in $\eta$ (or $\xi$) direction and find the correction to be applied to the coordinates of every interior point.
• For applying the transfinite interpolation to a cascade geometry, consider the four–sided geometry \((ABCD)\) (Fig. 3.2.3).

• It is desired to generate \(\zeta\)–constant and \(\eta\)–constant lines within \(ABCD\) which upon transformation would become equi–spaced orthogonal grid lines inside a rectangular domain of size \(1 \times 1\).

• The first task to be completed is the placement of grid points on the boundary (Fig. 3.2.3). Here, in order to get a rectangular grid, the number of points on opposite sides should be equal.

• Also, if some idea is available regarding the nature of gradients in the problem, the boundary points can be located so as to resolve the high gradient regions, i.e. the regions where the inter row spacing domain merges with the blade surfaces.
Fig. 3.2.3 Transfinite interpolation for an aerofoil cascade identification of boundaries
Fig. 3.2.4 Transfinite interpolation: Identification of boundary nodes
Now, let us apply linear interpolation in the $\xi$ – direction between two grid points which lie on the same $\eta = \text{constant}$ line. Since the total range of $\xi$ variation is from 0 to 1, the linear interpolation formulae for the coordinates of an interior point $P$ are written as

$$x'_p = (1 - \xi) x_E + \xi x_F; \quad y'_p = (1 - \xi) y_E + \xi y_F$$

where $E$ and $F$ are the two boundary grid points.

Carrying out the same operation for all the $\eta$ – constant lines including the boundaries ($\eta = 0$ and $\eta = 1$), the interpolated points (marked by $x$) will appear as in Figure 3.2.4.

It may be noted that on the boundaries $AB$ and $CD$, the grid points obtained by unidirectional interpolation between the coordinated of the corner nodes ($A$, $B$) or ($C$, $D$), do not coincide with actual grid points shown as dots (.)

Fig. 3.2.5 Transfinite interpolation: Calculation of interior nodes
• In order to remove this anomaly, the difference between the actual points and the interpolated points should be subtracted on the $\eta = 0$ and $\eta = 1$ boundary curves.

• Moreover, some corrections need to be applied to the coordinates of point $P$ which have been obtained by unidirectional interpolation in the $\xi$ – direction, along an $\eta = $ constant line.

• Introducing such corrections brings in the influence of the boundary grid point data in the $\eta$ – direction also, for determining the coordinates of point P. Considering two grid points $G$ and $H$ corresponding to the $\xi = $ constant line on which $P$ lies, the corrections for the coordinates of point $P$ are

$$
\Delta x_P = (1 - \eta) \Delta x_G + \eta \Delta x_H; \quad \Delta y_P = (1 - \eta) \Delta y_G + \eta \Delta y_H \quad (3.2.1)
$$
• Here $\Delta x_G$, $\Delta x_H$, $\Delta y_G$, $\Delta y_H$, are the corrections for the boundary points given by

\[
\Delta x_G = x'_G - x_G; \quad \Delta y_G = y'_G - y_G; \\
\Delta x_H = x'_H - x_H; \quad \Delta y_H = y'_H - y_H; 
\]  (3.2.2)

• In Eq. 3.2.2, the coordinates indicated with prime are those obtained by unidirectional interpolation in $\xi$ – direction and those without prime are the actual boundary grid point data. The final values of the coordinates of $P$ (after interpolation in both $\xi$ and $\eta$ directions) are obtained as

\[
x_p = x'_p - \Delta x_p \quad \quad y_p = y'_p - \Delta y_p
\]

• Performing the above sequence of operations for every interior point gives the mesh in the physical domain as shown in Fig. 3.2.6. The corresponding transformed mesh is as usual a rectangular grid.
Fig. 3.2.6 Transfinite interpolation: Final mesh.
It is important to note that unidirectional interpolation can be done first in the $\eta$ direction along each $\xi = \text{constant}$ curve. In that case, corrections will have to be applied for matching actual grid points and the interpolated points along the boundaries AD ($\xi = 0$) and BC ($\xi = 1$). The final grid obtained by both the above approaches will be exactly the same.
Domain Vertex Method

- Domain vertex method is also an interpolation method, preferably used for generating multi-block structured grids. They make use of tensor products of unidirectional interpolation functions for two or three dimensions.

- Relation between physical \((x, y)\) and transformed \((\xi, \eta)\) coordinates, in two dimensions, is given by:

\[
\begin{align*}
  x_i &= \Phi_N \xi \Phi_M \eta x_{iNM}, \quad i = 1, 2, \quad N, M = 1, 2 \\
  x_i &= \Phi_N \xi,\eta x_{iN}, \quad i = 1, 2, \quad N = 1, 2, 3, 4
\end{align*}
\]

- Here, suffix \(i\) indicates the physical coordinate directions. \(N\) and \(M\) represent node numbers in the direction of the coordinates.

- \(\Phi_N \xi\) and \(\Phi_M \eta\) are the unidirectional functions and \(\Phi_N \xi,\eta\) denotes tensor product.
The following functions are known as Blending functions:

\[
\Phi_N \xi, \eta = \begin{cases} 
\Phi_1 = 1 - \xi \ 1 - \eta &= \hat{\Phi}_1 \ \xi \ \hat{\Phi}_1 \ \eta \\
\Phi_2 = \xi \ 1 - \eta &= \hat{\Phi}_2 \ \xi \ \hat{\Phi}_1 \ \eta \\
\Phi_3 = \xi \eta &= \hat{\Phi}_2 \ \xi \ \hat{\Phi}_2 \ \eta \\
\Phi_4 = 1 - \xi \ \eta &= \hat{\Phi}_1 \ \xi \ \hat{\Phi}_2 \ \eta 
\end{cases}
\]

An example of this transformation from the physical to transformed domain in two dimensions is shown in Fig. 3.2.7.

Fig. 3.2.7 Two dimensional domain
• Similarly in three dimensions, relation between physical and transformed coordinates is given by

\[ x_i = \Phi_N \, \xi \, \hat{\Phi}_M \, \eta \, \hat{\Phi}_P \, \zeta \, x_{iNMP}, \quad i = 1, 2, 3 \quad N, M, P = 1, 2 \]

or

\[ x_i = \Phi_N \, \xi, \eta, \zeta \, x_{iN}, \quad i = 1, 2, 3 \quad N = 1, \ldots, 8 \]

• Blending functions:

\[
\Phi_N \, \xi, \eta, \zeta = \begin{cases} 
\Phi_1 = 1 - \xi \quad 1 - \eta \quad 1 - \zeta \\
\Phi_2 = \xi \quad 1 - \eta \quad 1 - \zeta \\
\Phi_3 = \xi \eta \quad 1 - \zeta \\
\Phi_4 = 1 - \xi \quad \eta \quad 1 - \zeta \\
\Phi_5 = 1 - \xi \quad 1 - \eta \quad \zeta \\
\Phi_6 = \xi \quad 1 - \eta \quad \zeta \\
\Phi_7 = \xi \eta \zeta \\
\Phi_8 = 1 - \xi \quad \eta \zeta 
\end{cases}
\]
• An example of this transformation from the physical to transformed domain in three dimensions is shown in Fig. 3.2.8.

Fig. 3.2.8. Transformation of Three Dimensional Domain using Domain Vertex Method
Exercise Problems

• Using transfinite interpolation method generate grids for

The region bounded by

\[
\frac{x + 3}{4} + \frac{y^2}{9} = 1, \quad \frac{x - 3}{4} + \frac{y^2}{9} = 1
\]

\[
\frac{x^2}{9} + \frac{y + 3}{4} = 1, \quad \frac{x^2}{9} + \frac{y - 3}{4} = 1
\]

• Using transfinite interpolation method generate grid for any triangular region.
Summary of Lecture 3.2

Methods for algebraic mesh generation and with transfinite interpolation are illustrated for structured grids. A domain vertex method popular for finite element methods is also presented.

END OF LECTURE 3.2