Meshing for Numerical Simulations: A Generic Approach

Scope of the this Presentation

The compilation is to address & explain the standard practice of meshing activities which can be subsequently tailored to meet specific solver requirements. Various Topics covered are:

1. Explain types of mesh & related standard terminologies
2. Outline general approach to mesh generation starting from CAD data
3. Various categorization of mesh-generation methods
4. Describe the mesh quality parameters
5. Demonstrate the meshing methodology through a comprehensive & step-by-step approach while benchmarking various tools available in the industry

It should be noted that "Mesh Generation" and "Discretization" are not the same activity. Mesh generation is the process of dividing the computational domain into discrete (& finite) smaller domain whereas Discretization method is a method of approximation of the differential equations by a system of algebraic equations for field variables at some set of discrete locations in space (generated during Meshing operation) and time.

Mesh generation is the most time consuming and tedious part of any numerical simulation technique. The activity gets more complicated due to growing number of availability of Meshing Tools (applications) having varying Graphics User Interface, Meshing Algorithms, Geometrical Data Handling and Compatibility with Solvers. This article intends to bring out the common features available in the commercial meshing tools available in the market today.

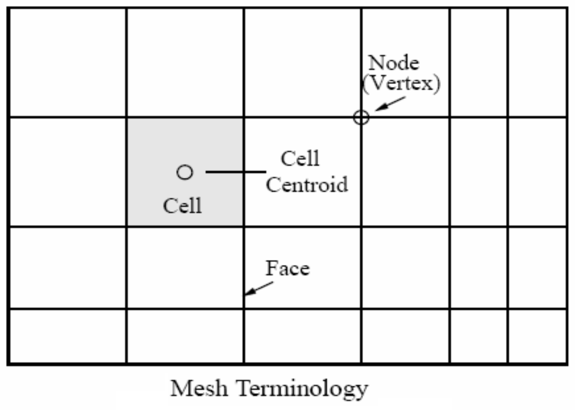
Meshing Algorithms Work on Topology of the Geometry i.e. connectivity of the points to edges, edges to surface and surfaces to volume is prime requirement of the mesh generation software. Presence of the so called "holes" in the geometry or a tee-joint may result in failure or failure to start the mesh generation process.

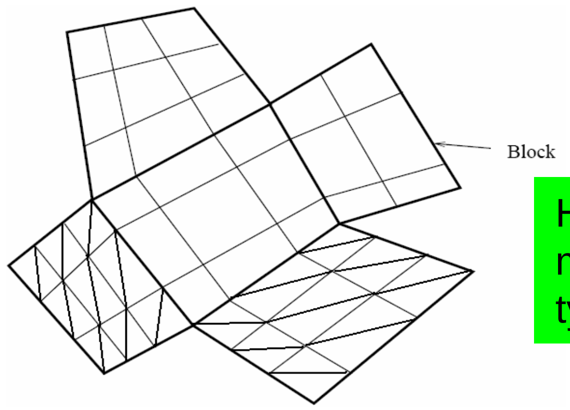
STEP-0:

Set the geometry merge tolerance in your CAD software consistent with your pre-processor. Typically CAD packages use a **merge tolerance** of the order of 0.1 ~ 0.5 mm whereas meshing software uses a merge tolerance of the order of 0.001 mm

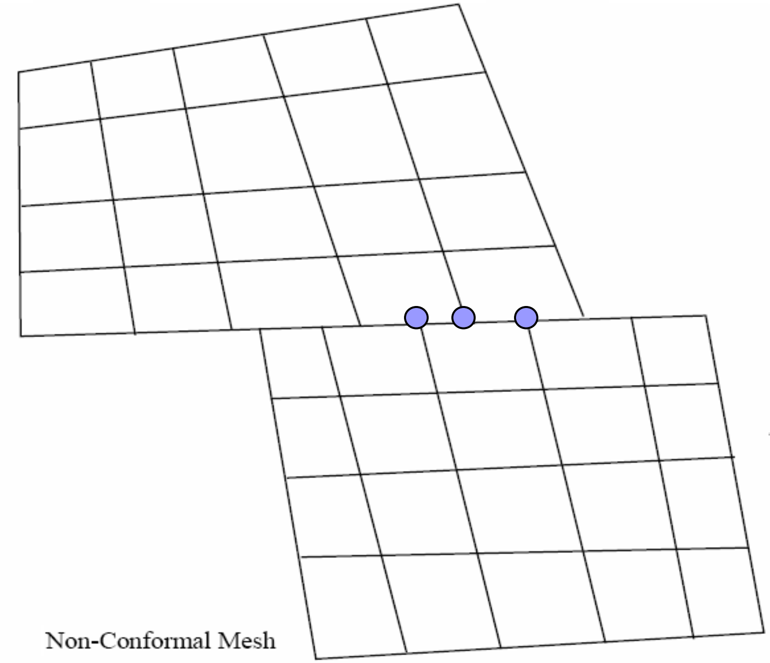
Mesh Generation – Process: Terminology

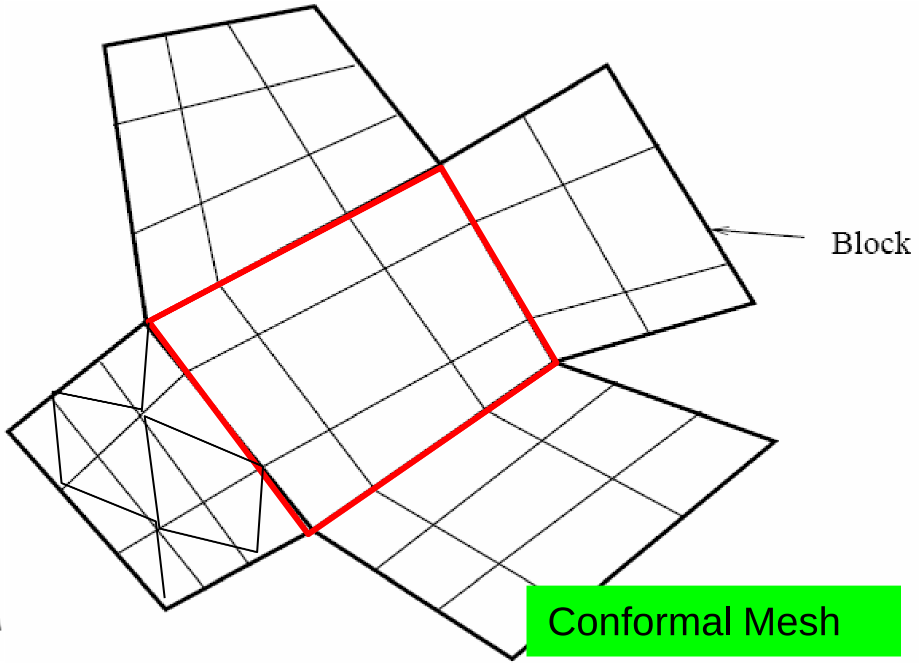
Before we proceed further, make yourselves familiar with the jargons and technical terminologies used for a mesh and its constituents namely node, elements (cells).





Hybrid Mesh: more than one type of elements





Conformal Mesh

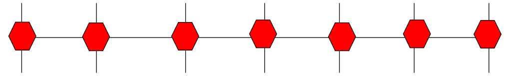
Mesh Generation – Process: Type of Elements

FEA: Lagrangian Description

Eulerian Description CFD

1D

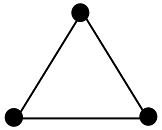
Line, Beam, Link, Spring

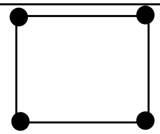


No Utility

2D

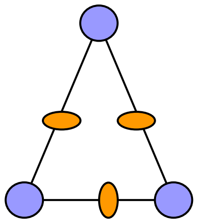
Tri(angular)

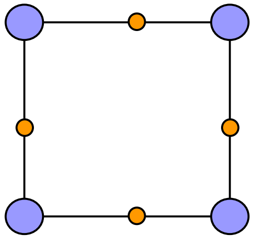


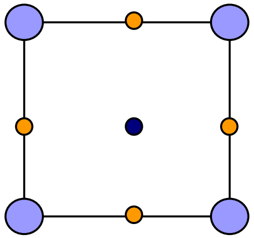


Quad(rilateral)

Limited Utility, Only 1st Order Elements are used in CFD



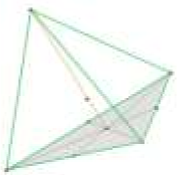


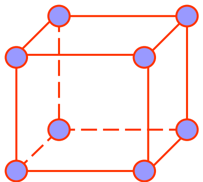


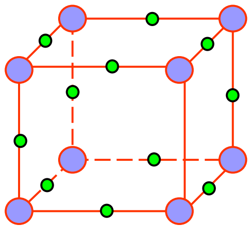
Higher Order Are More Accurate

3D

Tetra(hedron)



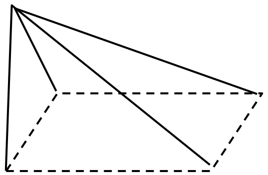


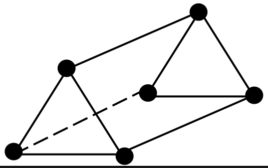


Hexa(hedral)

MAX Utility, Only 1st Order Elements

Pyramid





Prism or Wedge

Pyramids are used only as a transitional element between Tets and Hexs.

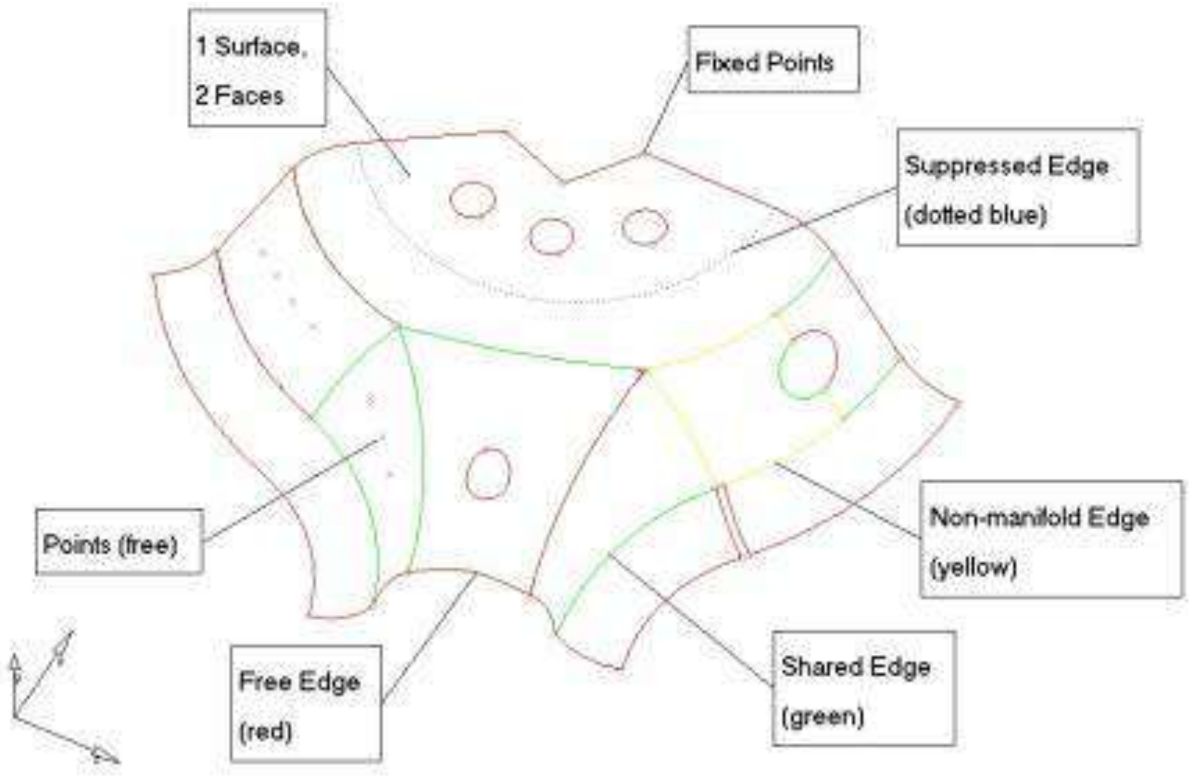
Note: Tri & Tets are called Simplex Elements (A simplex is a set of n+1 points in n-dimensions). Displacement field these elements are linear and constant. The displacement field of quad and hex are linear and tri-linear respectively. Quad and hex represent a linear displacement field & constant strain field exactly.

NODE:

•It is the most basic element of Finite Element Technology (analogous to “point” entity of CAD Technology

•Nodes represent the Computational Domain being analyzed and are referenced by higher order entity Elements to define the location and shape of the elements

•Nodes (& hence elements) are topologically associated with lines & surfaces. Hence, while executing pre-processing operations, nodes and elements can be selected “by surface”

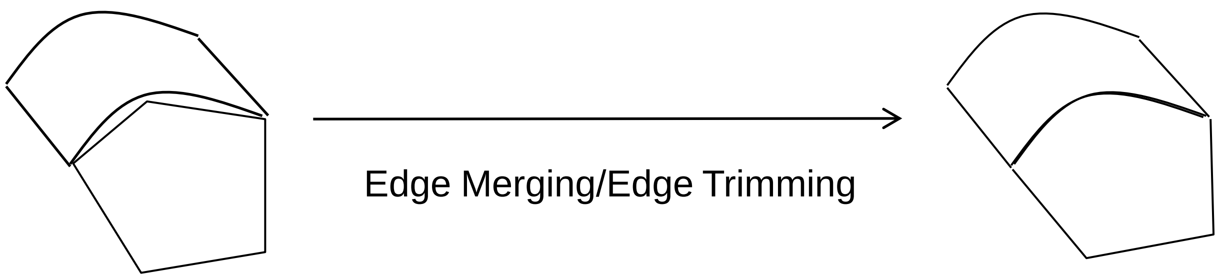


HyperMesh terminology

Mesh Generation - Process

STEP-1:

Translate the CAD data into neutral file format such as (IGES, STEP, Parasolid), import it into your preprocessor. Ensure consistency of the topology of computational domain with the help of **Geometry Cleanup / Defeaturing** options available. Go though the comprehensive list of methods and best practices available on our website. You must ensure that the geometry results in a so called **"Water Tight" topology**

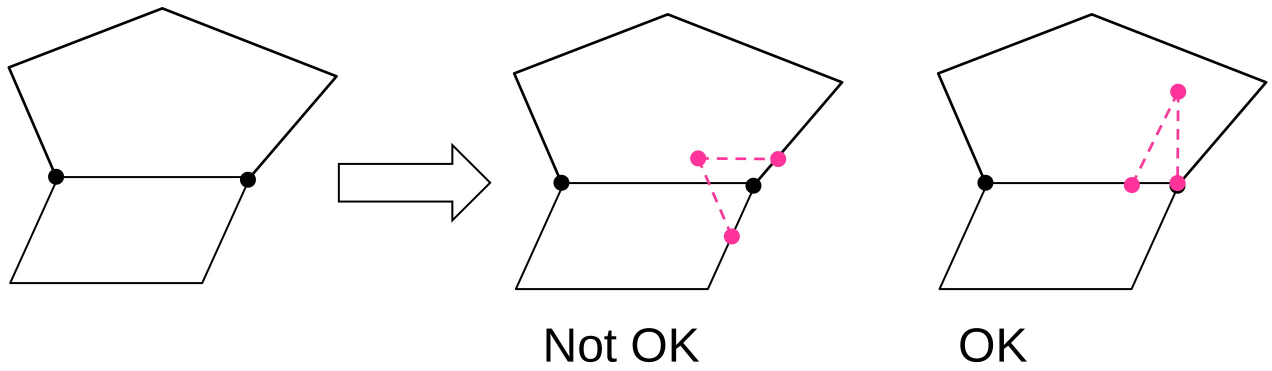


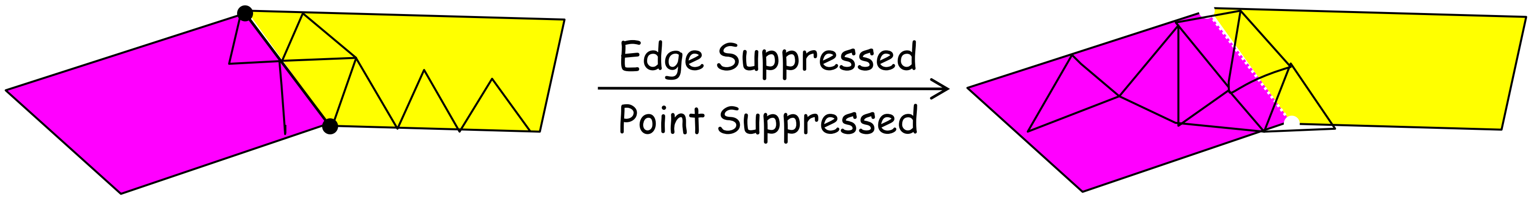
STEP-2:

Topology Simplification / Optimization: This step is important when you are using **Free Mesh** Algorithm for mesh generation. Typically, all the meshing algorithms are designed to respect the points, surface boundaries (edges) and the surface envelop. Thus, wherever there is point in your topology

STEP-2 (contd):

a node will be placed there. Similarly, the edges of the surfaces will be broken down into edges of the elements and faces of elements will add up to the nearest surface. In the example below, the element shown at the centre not only ignored (key-, hard-) point, is also ignored the edge dividing the two surfaces.





Note how the meshing pattern may get changed when points and edges are suppressed. Hence, points and edges must not be suppressed where 1. they are not tangent 2. angle between the surfaces > 10 deg

STEP-3 (contd.):

Mesh Sizing Control: Before you move on further, make yourself familiar with the way your preprocessor stores geometrical (points, curves, surface, volumes, material) and numerical entities (nodes, edges, faces, elements). For example, ICEM stores them in “Parts”, ANSYS stores them in “Components”, Hypermesh stores them in “Collectors” and GAMBIT stores them in “Zones”.

The sizing control in all preprocessors have the following precedence:

Global Size < Size at a Point < Size on a Line < Size over a Surface

Size at a Point: Equivalent to "Mesh Density" option in ICEM & Keypoint Sizing in Ansys. Points constituting geometry after importing CAD data into pre-processor act as hard points where the mesher will put a HARD that cannot be moved during mesh smoothing. Hence, such hard points should be deleted (in ICEM) & suppressed in HM for better control on mesh smoothing. <u>This sizing defines the mesh size along any curve that references that point as an end point, and doesn't have a specific mesh definition along the curve.</u> Application includes: **wake region such as flow over cylinder, car.**

STEP-3 (contd.):

Sizing on a Curve:

This is called "Line Sizing" in Ansys. It implies that the edge length of elements on this curve cannot be overridden by Global or Smart Sizing. Hence, the software will calculate Element Edge Length if no. of elements on the curve is specified or it will calculate the nearest lower integer from the ratio Curve Length / Edge Length of Element.

Sizing on Surface:

This is called "Area Sizing" in Ansys.

STEP-3: General Guidelines for Mesh Generation:

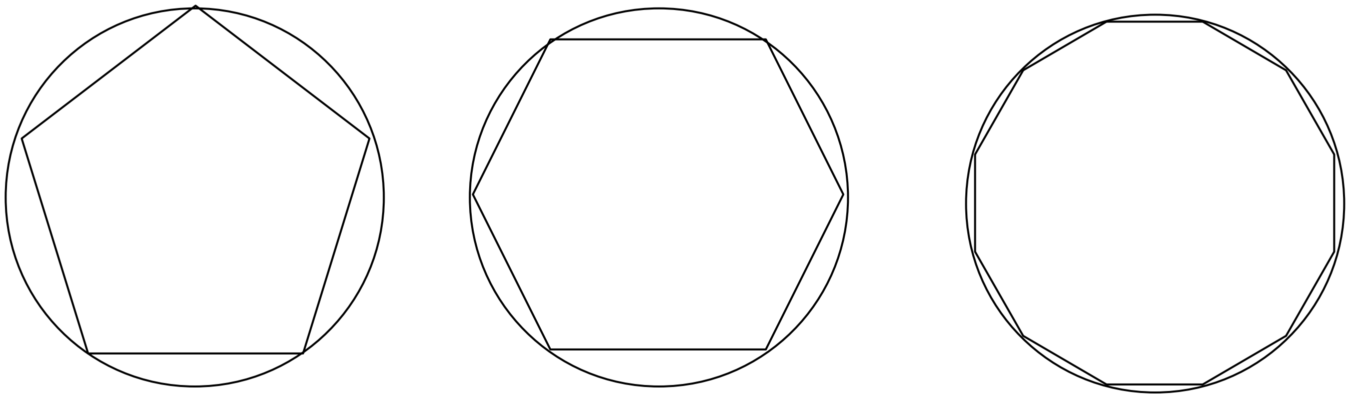
1. In a strict sense, there is no universal rule for mesh generation and the method and requirements would vary from one problem to another. However, the simulation engineer must ensure that method (software, simulation process, etc) should give physically realistic solution even for coarse & non-uniform grid.
2. An exploratory coarse-grid solution would not be useful if the method gives reasonable solution only for sufficiently fine grids.
3. Excerpts from Numerical Heat Transfer and Fluid Flow by S. V. Patankar

   The number of grid points needed for given accuracy and the way they should be distributed in the calculation domain are matters that depend on the nature of the problem to be solved. Exploratory calculations using only a few grid points provide a convenient way of learning about the solution. After all, this is precisely what is commonly done is a laboratory experiment. Preliminary experiments or trial runs are conducted, and the resulting information is used to decide the number and locations of the probes to be installed for the final experiment.

STEP-3 (contd.):

Miscellaneous Sizing Control:

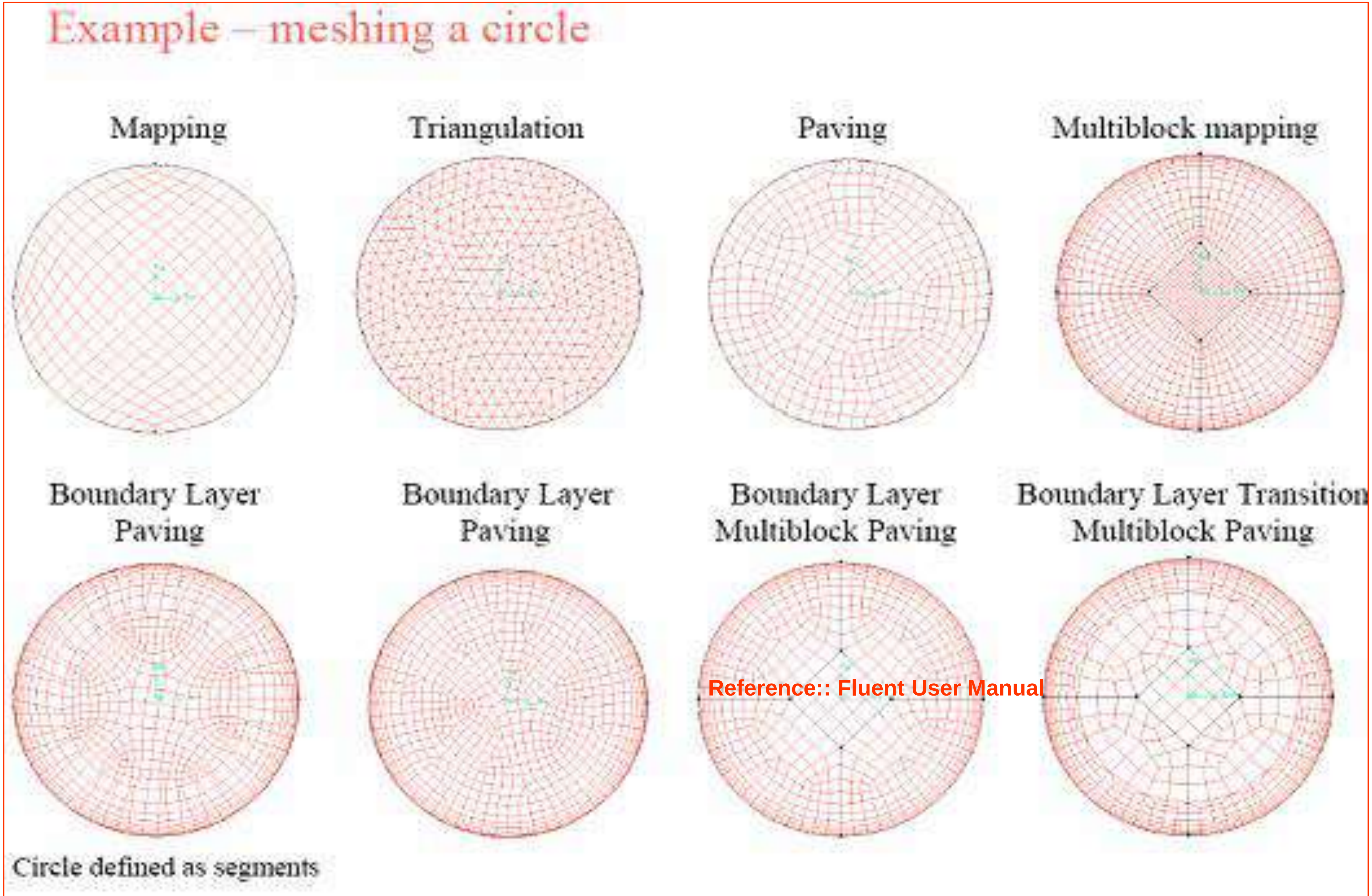
Chordal deviation: It is a meshing algorithm that automatically varies node densities and biasing along curved surface edges to gain a more accurate representation of the surface being meshed.



STEP-3 (contd.):

Miscellaneous Sizing Control:

Natural Sizing: It is a meshing algorithm that is very similar to Chordal deviation method explained on previous slide. However, the mesh sizing control is a bit different as described below.



Reference:: Fluent User Manual

CFD Vs FEA Mesh

Distinction between FEA Mesh & CFD Mesh:

1. CFD meshes consist of 1st order elements only. Emphasis is on Boundary layer Resolution
2. 2nd order elements are always desirable for FEA analysis. Emphasis is on finer elements near load and reaction points
3. CFD elements do not possess any information except Material Definition. FEA elements may possess specific definition like Spring-, Mass-, Bar-, Rod-, Rigid-, Rigid Link-, Joint-, Gap-, Weld-Elements
4. FEA software vendors develop their own library of Elements for ease of Modeling.

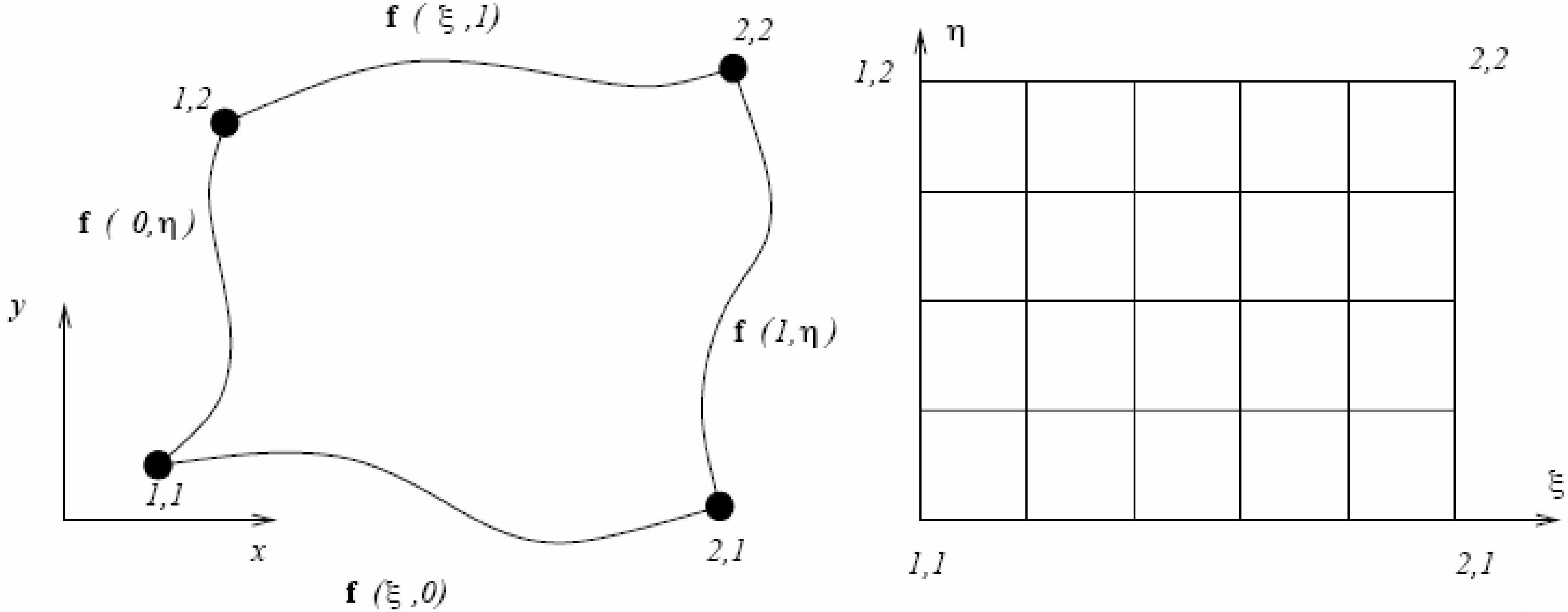
FD vs. FE vs. FV Formulation:

1. These are mathematical formulation for governing differential equations and not the type of mesh.
2. FD refers to mathematical formulation based on Taylor Series Approximation. This is applicable to Structured Meshes only.
3. FE formulation usually refers to cased where Shape Functions are used for interpolation of node values over entire element.
4. FV formulation refers to formulation where virtual elements (Finite Volume) based on geometric elements are created to calculate flux quantities.

Mapping Techniques

- Complex domain is transformed into simple one where mesh may be generated easily
- Most widely used in O-grid & C-grid Generation

Co-ordinate Mapping



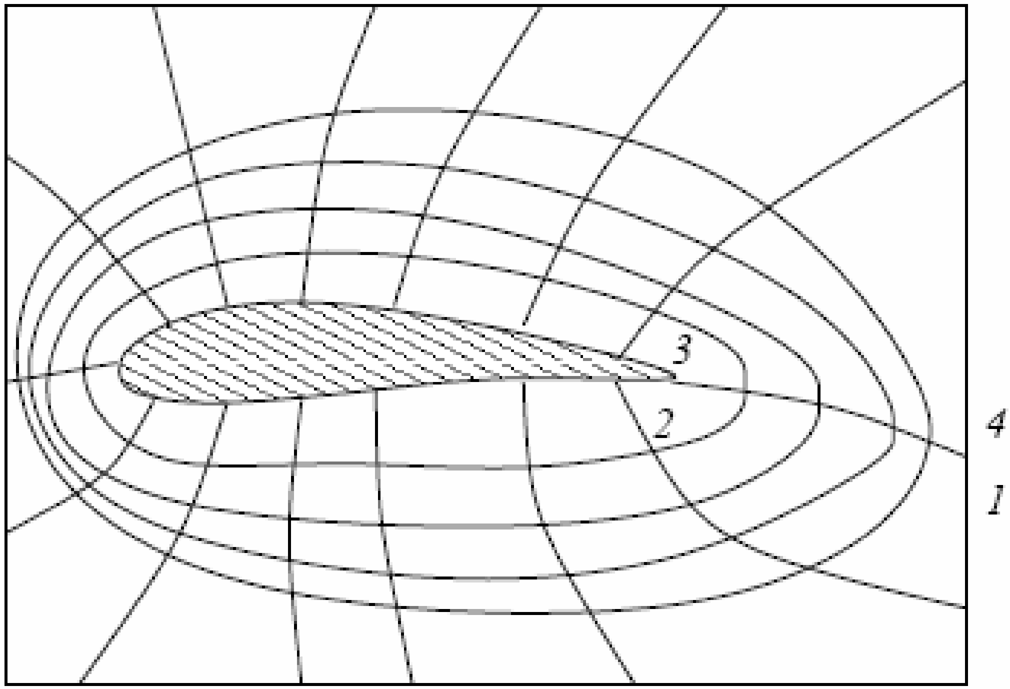
Mapping of a simply connected region (left) onto a rectangular computational domain (right)

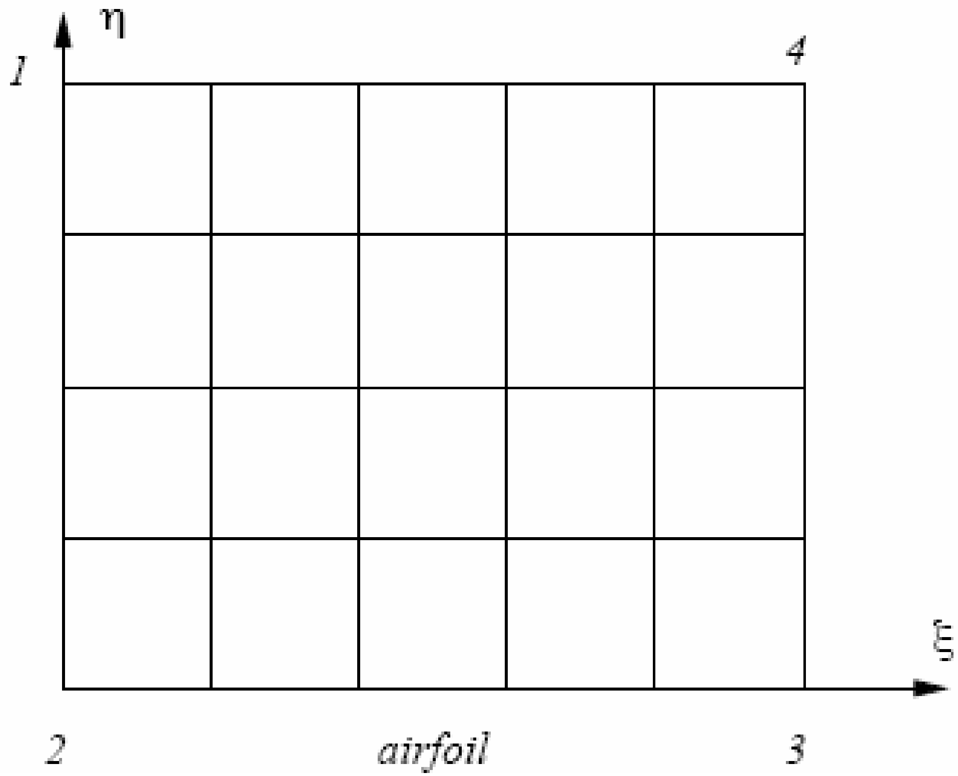
1. ***Co-ordinate transformation equation in a physical domain***
   - Lagrange Transformation
   - Hermite Transformation (Shearing Transformation)
   - Trans Finite Interpolation (TFI)
   - Multi-surface Transformation
2. ***Elliptic Mesh Generation***
   - Solution of PDE formulated by a set of Poisson's equation with forcing terms usually defined by Thomas-Middle coefficient term
   - They produce very smooth grid and mesh is typically orthogonal to the boundaries
3. ***Hyperbolic Method***
   - Solution of PDE of hyperbolic type, that are solved marching outward from the domain boundaries.
   - Very effective for external flows where the wall boundaries are well-defined, whereas the far-field boundaries are left arbitrary.

O-grid Generation Technique

•Lines of constant ξ are rays from the airfoil surface to the far field boundary in the physical plane

•Lines of constant η are closed curves encircling the airfoil

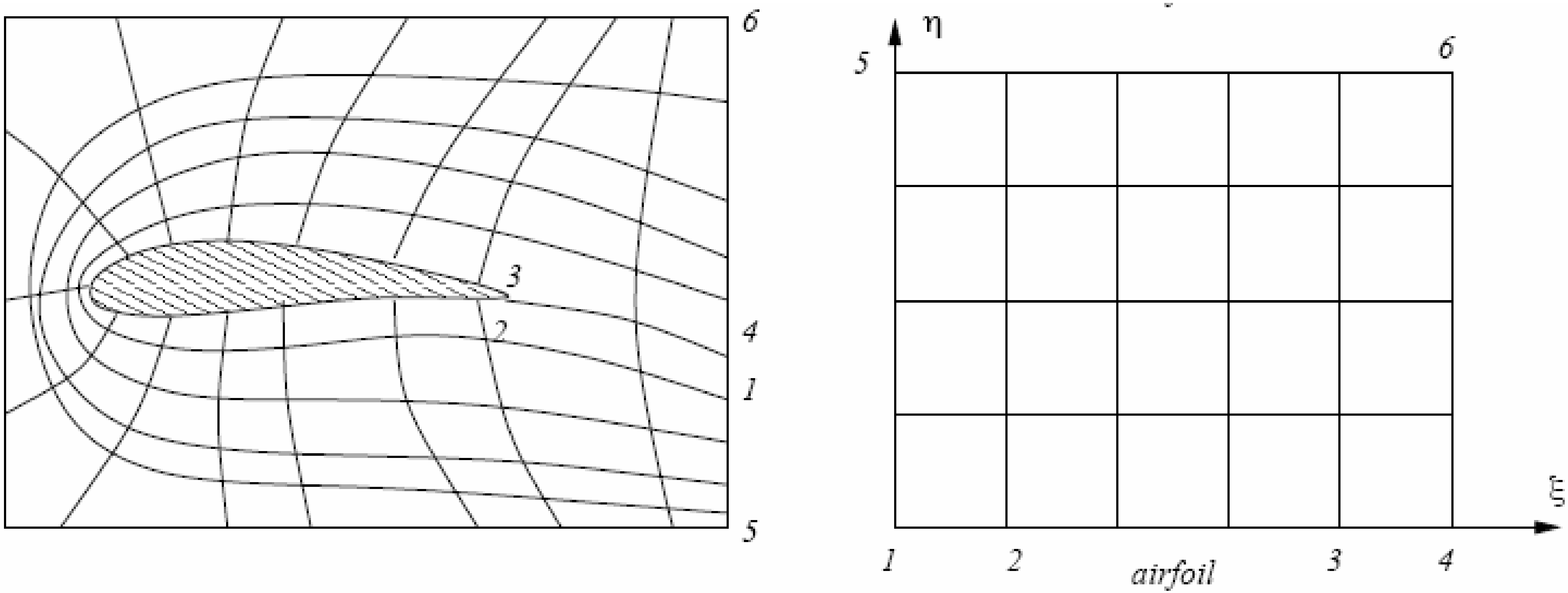




C-Grid Generation Technique:

•Lines of constant ξ become curves beginning and ending at the outflow boundary and surrounding the airfoil

•Lines of constant η are rays from the airfoil surface or the cut to the outer boundary

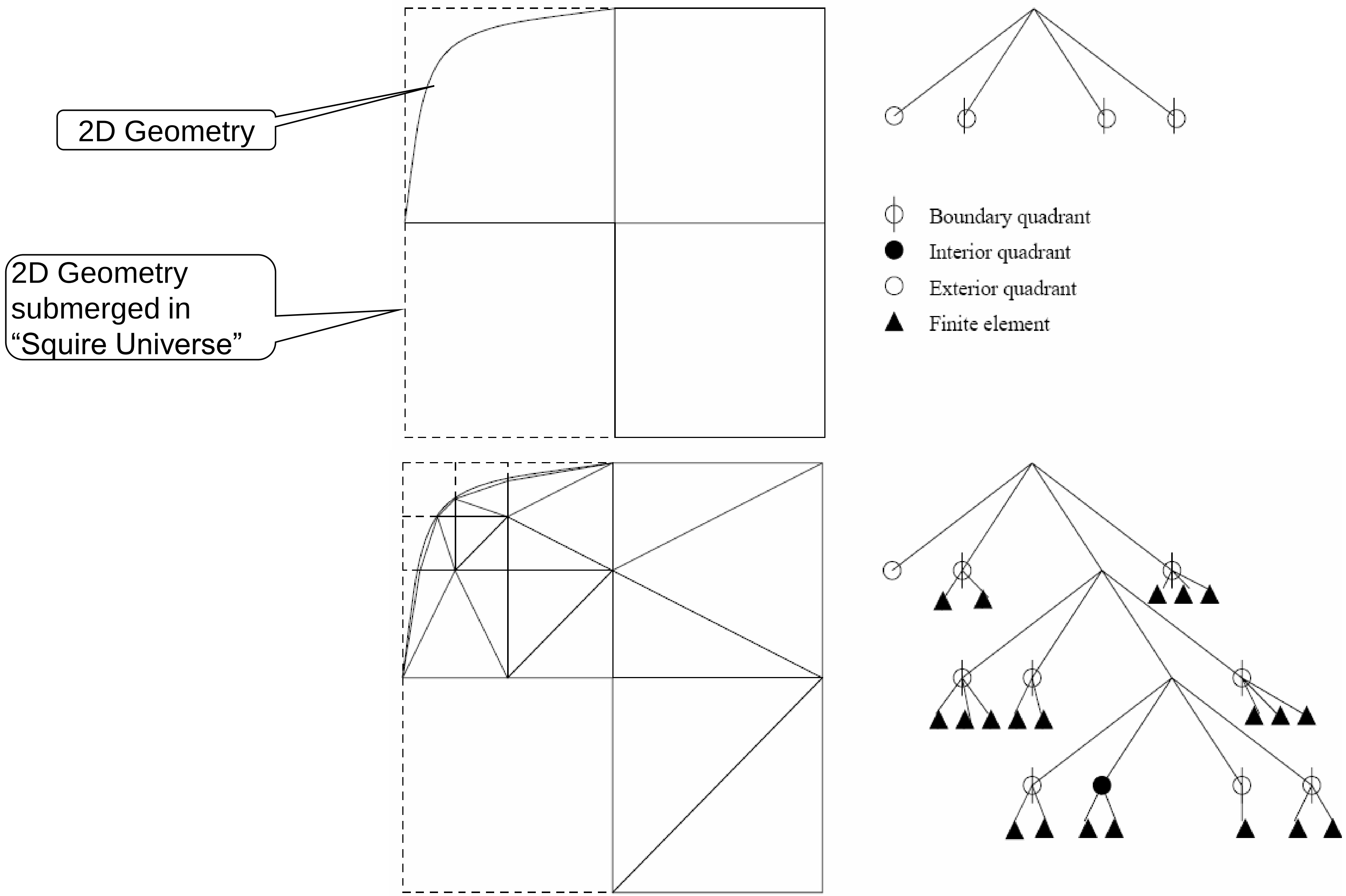


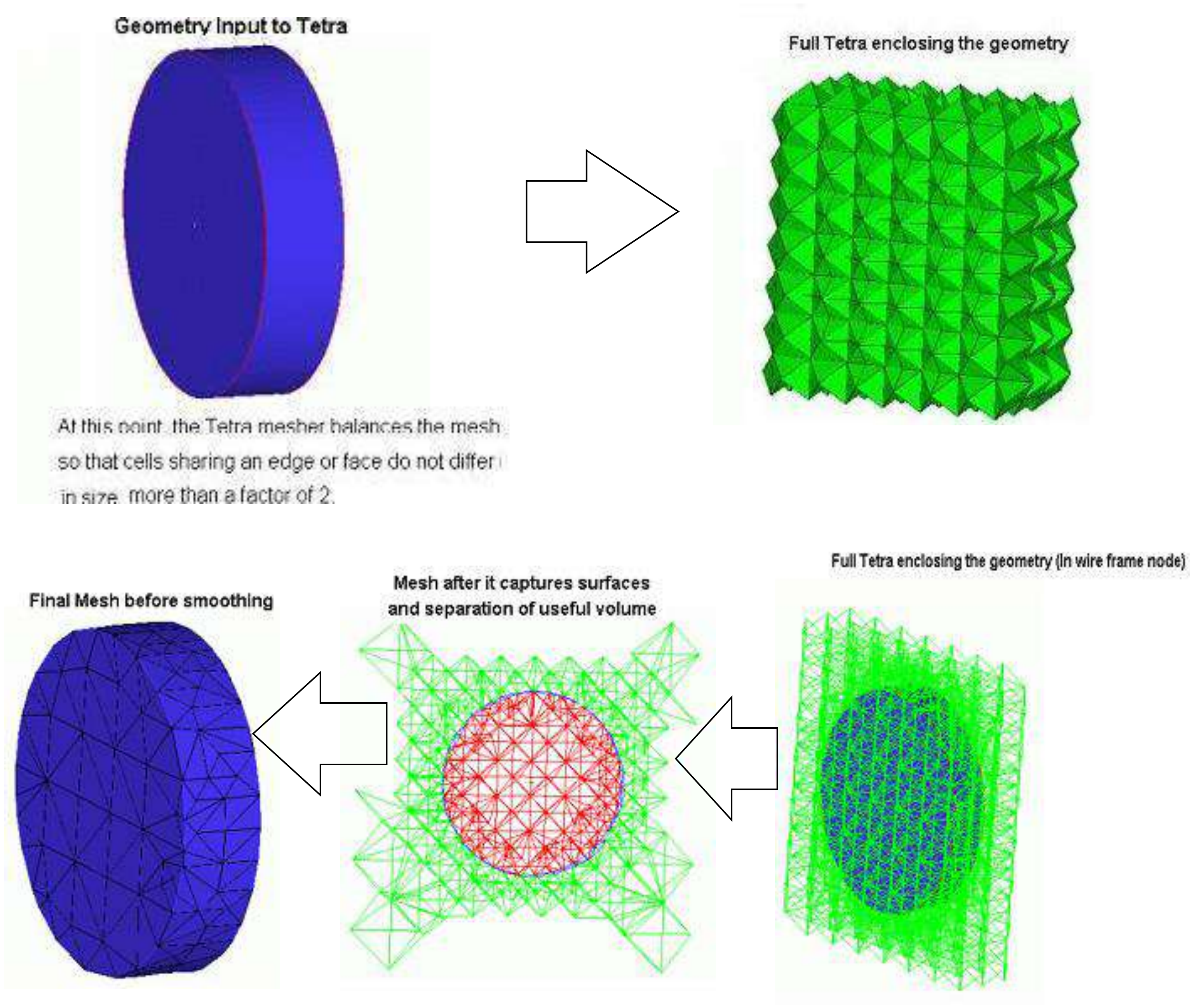
1. Delaunay Triangulation: Constrained Delaunay
2. Octree Method
3. Advancing Front Method
4. Point Insertion
5. Recursive Bisection
6. Voronoi Method
    - Some meshing software calculates "Delaunay Violation" as mesh quality control parameters

Unstructured grids are best characterized by no such repeating geometry, and structure that can be controlled only very locally. Unstructured grids are typically formed from simplexes such as tetrahedron, and the fact they have no repeating structure can make it very difficult to create and compute the necessary cell-to-cell connectivity for CFD. The random orientation of an unstructured grid can be lead to awkward interfaces within the grid, possibly reducing the final accuracy of the solution. Simplexes often require many more cells to discretize a given space. To see this, consider a single hexahedron, e.g. a cube, which requires at least five tetrahedron to describe solely with tetrahedron (and six to do so conveniently). While some of this 6:1 ratio can often be recovered for a complex geometry, not all of it can.

The Delaunay criterion, sometimes called the "empty sphere" property simply stated, says that any node must not be contained within the circumsphere of any tetrahedra within the mesh. A circumsphere can be defined as the sphere passing through all four vertices of a tetrahedron. The Delaunay criterion in itself, is not an algorithm for generating a mesh. It merely provides the criteria for which to connect a set of existing points in space. As such it is necessary to provide a method for generating node locations within the geometry. A typical approach is to first mesh the boundary of the geometry to provide an initial set of nodes. The boundary nodes are then triangulated according to the Delaunay criterion. Nodes are then inserted incrementally into the existing mesh, redefining the triangles or tetrahedra locally as each new node is inserted to maintain the Delaunay criterion. It is the method that is chosen for defining where to locate the interior nodes that distinguishes one Delaunay algorithm from another.

Quadtree (2D) / Octree (3D)





Geometry Input to Tetra
Full Tetra enclosing the geometry
At this point, the Tetra mesher balances the mesh so that cells sharing an edge or face do not differ in size more than a factor of 2.
Final Mesh before smoothing
Mesh after it captures surfaces and separation of useful volume
Full Tetra enclosing the geometry (In wire frame node)

Mesh Generation: FD Vs FE Vs FV

1. **Finite Difference Method:** The starting point is the conservation equation in differential form. The solution domain is covered by a grid. At each grid point, the differential equation is approximated by replacing the partial derivatives by Approximations in terms of the nodal values of the functions. The result is one algebraic equation per grid node, in Which the variable value at that and a certain number of neighbor nodes appear as unknowns. For all practical purpose, FD method is applied to structured grids where grid lines serve as local coordinate lines. Taylor series expansion or polynomial fitting is used to obtain approximations to the 1st and 2nd derivatives of the variables with respect to coordinates. When necessary, these methods are also used to obtain variable values at locations other than grid node (interpolation).

Salient Features:

*Easy to discretize

Drawbacks:

*Conservation is not enforced

2. **Finite Volume Method:** The starting point is conservation equation in integral form. The solution domain is subdivided into a finite number of contiguous control volumes and the conservation equations (mass, momentum, energy, etc) are applied to each CV. At the centroid of each Control Volume lies a computational node at which the variable value is calculated. Interpolation is used to express variable values at the CV surface in terms of the nodal values. Surface and volume integrals are approximated using suitable quadrature formula generating an algebraic equation for each CV, in which a number of neighbor nodal values appear. The method ensures conservation by construction, so long as surface integrals (which represent convective and diffusive fluxes) are the same for the CVs sharing the boundary. However, this method requires 3 levels of approximation namely Interpolation, Differentiation and Integration.

Salient Features:

*Conservation is always enforced

*Easy to discretize and formulate

Drawbacks:

*Difficult to implement on unstructured grid

3. **Finite Element Method:** The FE method is similar to FV method in many ways. The domain is divided into several smaller interconnected sub-domain or finite elements (it is called Finite because the number of smaller sub-domain is finite as compared to infinite possibilities in a continuum) that are generally unstructured. The distinguishing feature of FE method is that the equations are multiplied by a weight function before they are integrated over the entire domain. In the simplest FE methods, the solution is approximate by a linear shape function within each element in a way that guarantees continuity of the solution across element boundaries. Such a function can be constructed from its values at the corners of the elements. The weight function is usually of the same form. This approximation is then substituted into the weighted integral of the conservation law and the equations to be solved are derived by requiring the derivative of the integral with respect to each nodal value to zero. This corresponds to selecting the best solution within the set of non-linear algebraic equations.

Salient Features:

*Easy to discretize and formulate, easy to implement on unstructured grid

Drawbacks:

*Requires special mathematical treatment to enforce conservation

4. **Control Volume based Finite Element Method (CV-FEM):** Here, shape functions are used to describe the variation of the variables over an element. Control volumes are formed around each node by joining the centroid of the elements. The conservation equations in integral form are applied to these CVs in the same way as in the Finite Volume Method. The fluxes through CV boundaries and the source terms are calculated element-wise.

Salient Features:

*Conservation is always enforced

*Takes advantage of FE and FV approach

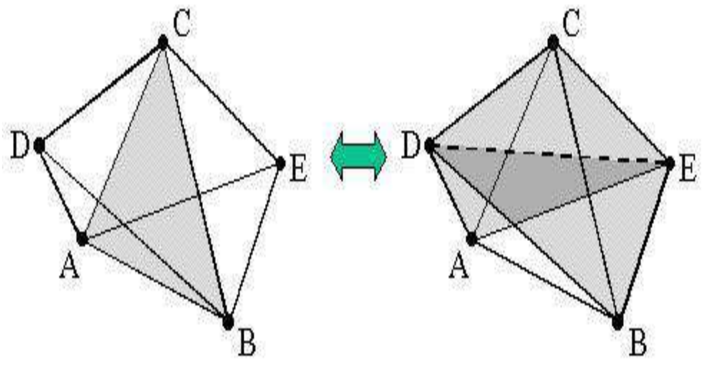
Drawbacks:

*Difficult to formulate

No mesh generation technique can be 100% manual, that is where the user has complete control of location of the nodes. However, the “blocking technique” used in ICEM can be termed as near manual where user controls the shape and size of the mesh. Due to increased complexity of problems (geometries), the auto-mesh generation is considered to be a economical compromise between cost and accuracy. However, no auto mesh generation algorithm can generate mesh without having a few elements having poor quality. These elements need to be improved without repeating the complete process. This is called topology improvements.

Node Movement Techniques: Laplacian Smoothing: This smoothing technique places each vertex at centroid of its neighbouring vertices. Elements connectivity doesn't change. This is an “auto-smoothing” process which does not yield desired improvement always.

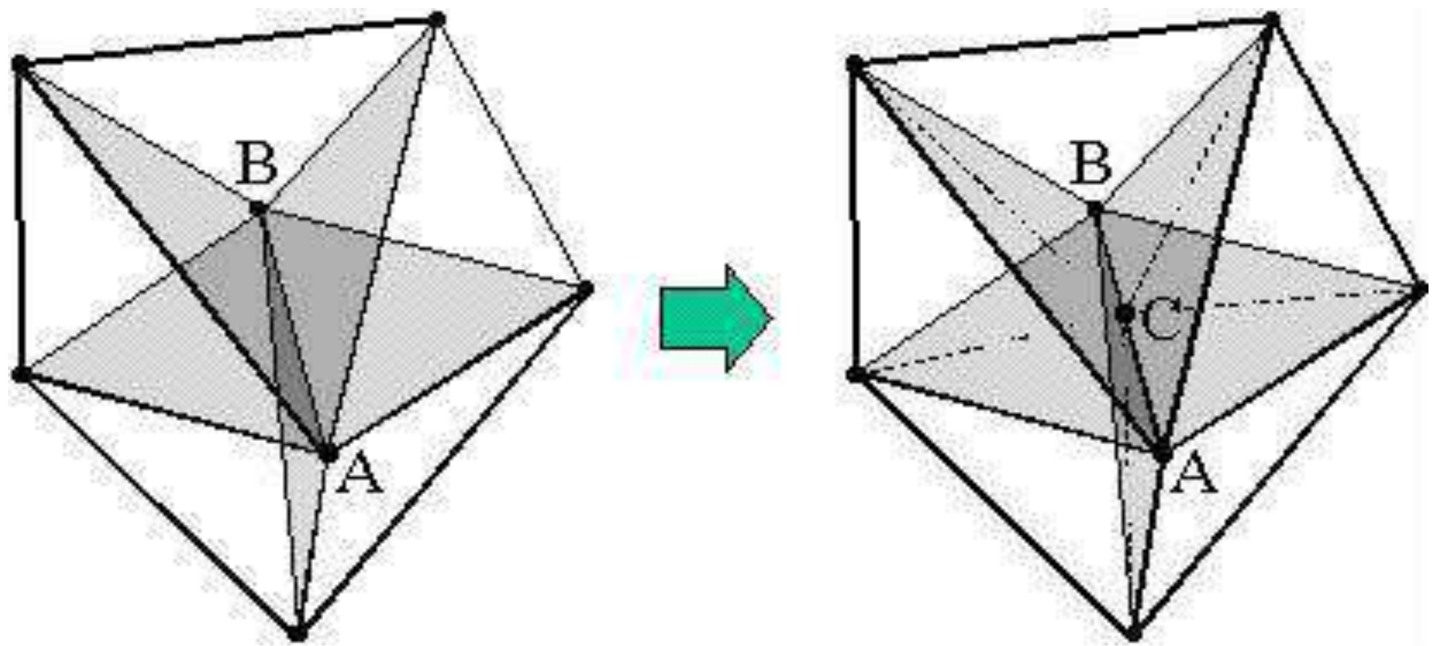
Edge Movement Techniques: The nodes are fixed, element connectivity gets changed such as Swapping two adjacent interior tetrahedrons sharing the same face for three tetrahedrons.



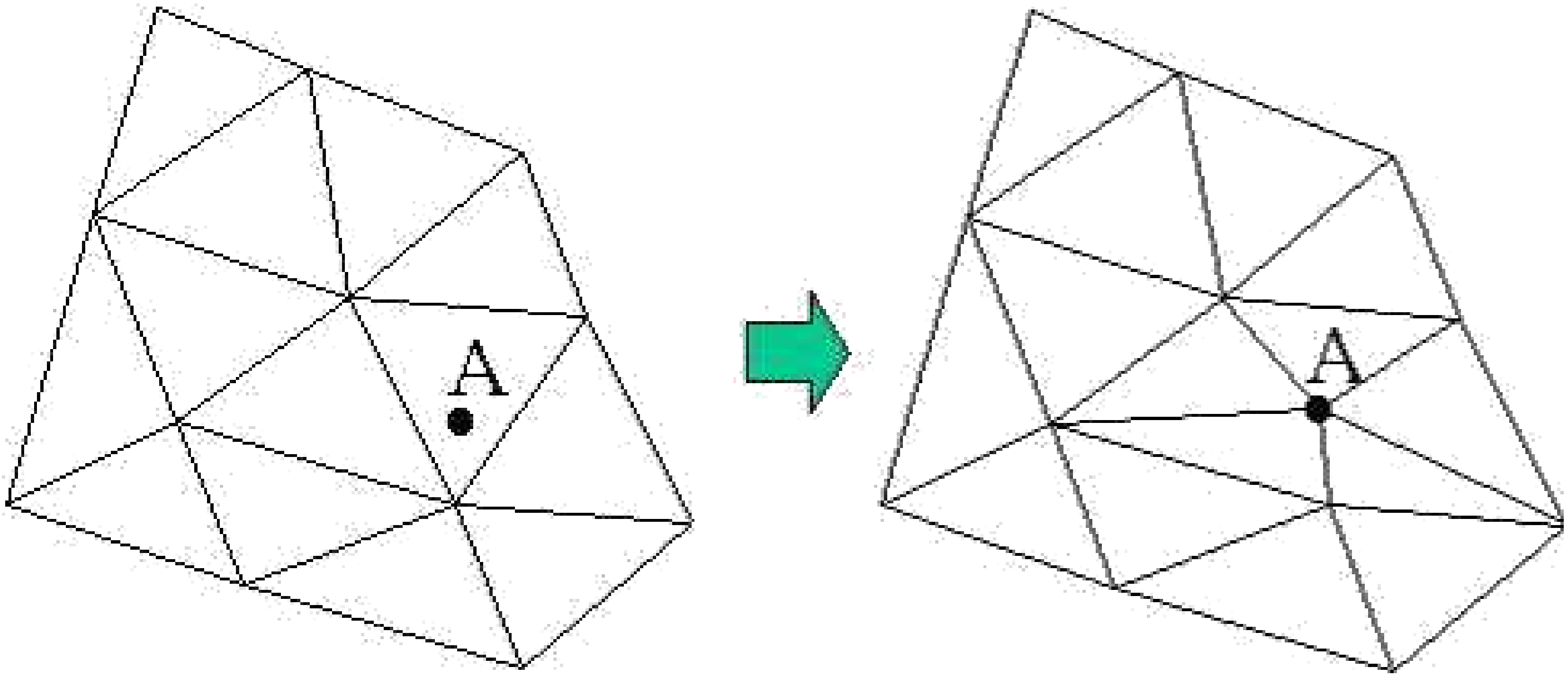
Tet transformation where 2 tets are swapped for 3

Topological improvement A common method for improving meshes is to attempt to optimize the number of edges sharing a single node. This is sometimes referred to as node valence or degree. In doing so, it is assumed that the local element shapes will improve. For a triangle mesh there should optimally be six edges at a node and 4 edges at a node surrounded by quads. Whenever there is a node that does not have an ideal valence, the quality of the elements surrounding it will also be less than optimal. Performing local transformations to the elements can improve topology and hence element quality. For volumetric meshes, valence optimization becomes more complex. In addition to optimizing the number of edges at a node, the number of faces at an edge can also be considered. For tetrahedral meshes this can involve a complex series of local transformations. For hexahedral elements, valence optimization is generally not considered tractable. The reason for this is that local modifications to a hex mesh will typically propagate themselves to more than the immediate vicinity.

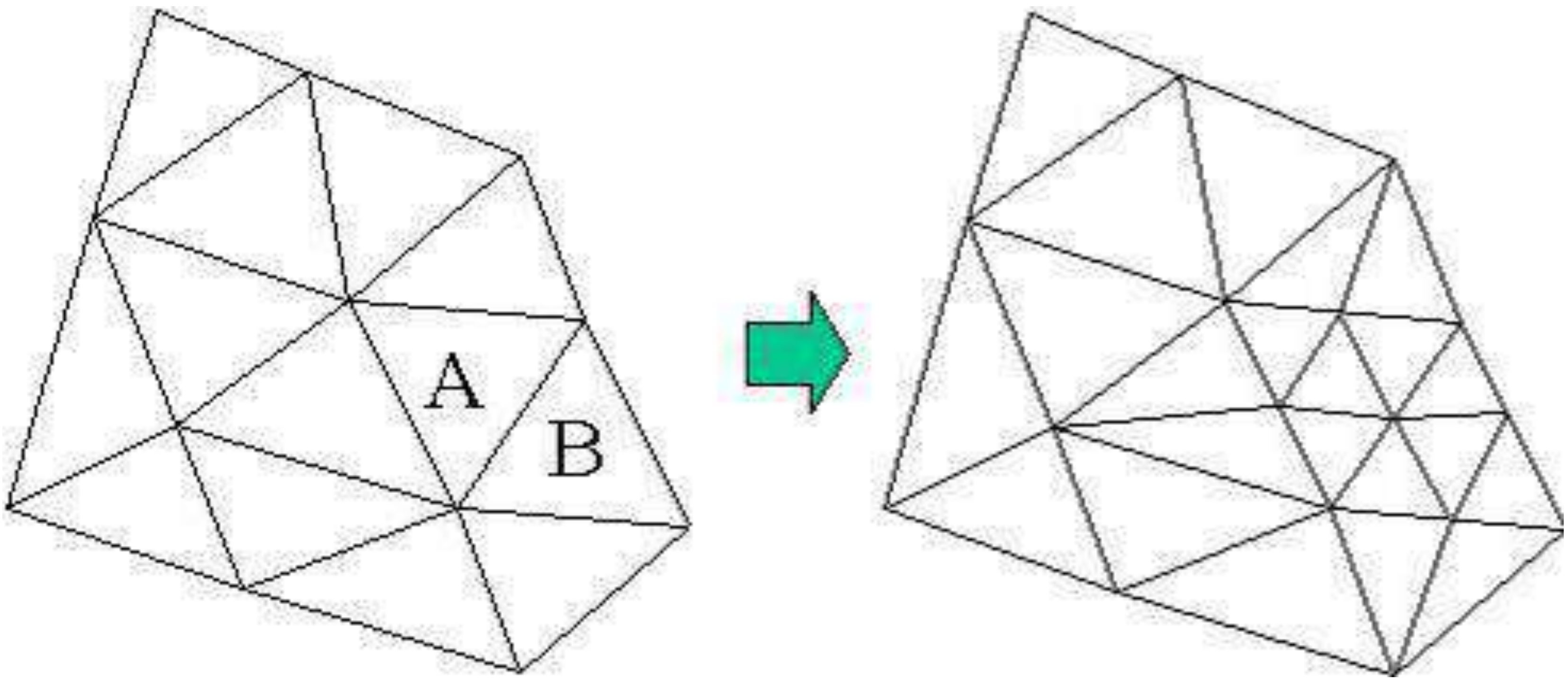
Method-1: Edge Bisection - Edge bisection involves splitting individual edges in the triangulation. As a result, the two triangles adjacent the edge are split into two. Extended to volumetric meshing, any tetrahedron sharing the edge to be split must also be split as illustrated in Figure.



Method-2: Point Insertion - A simple approach to refinement is to insert a single node at the centroid of an existing element, dividing the triangle into three or tetrahedron into four. This method does not generally provide good quality elements, particularly after several iterations of the scheme. To improve upon the scheme, a Delaunay approach can be used that will delete the local triangles or tetrahedra and connect the node to the triangulation maintaining the Delaunay criterion. Any of the Delaunay point insertion methods discussed previously could effectively be used for refinement.



Method-3: Template - A template refers to a specific decomposition of the triangle. One example is to decompose a single triangle into four similar triangles by inserting a new node at each of its edges as show in Figure. The equivalent tetrahedron template would decompose it into eight tetrahedra where each face of the tetrahedron has been decomposed into 4 similar triangles. To maintain a conforming mesh, additional templates can also be defined based on the number of edges that have been split.



Scope of the this Presentation

This article is to explain the typical quality measurement criteria for "elements" and their relative importance from solver and solution convergence/accuracy point of view.

1. Explain the 'Variable'
2. Make sample calculation
3. Highlight relative importance

Though there are many ways to measure "quality" of elements, not only different application emphasize different variable but different software has different variable set as "default" quality parameters. In ICEM, the default variable for elements and quality is as follows:

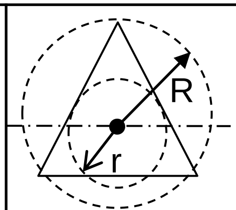
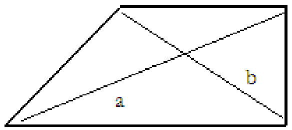
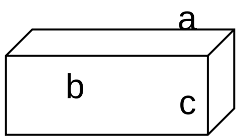
Tri /Tetra	–	Aspect Ratio
Quad/Hexa	-	Determinant
Pyramid	-	Determinant
Prism	-	Minimum of determinant and Warpage

Mesh Generation: Mesh Quality Parameters

- Importance of various Quality parameters is different for Tri / Tetra / Prism elements as compared to Quad/Hex elements. Most often talked about quality parameters are:

1. Aspect Ratio
2. Internal Angle Deviation
3. Jacobian / Determinant
4. Equi-angle Skewness
5. Warpage / Warp Angle
6. Tetra Collapse

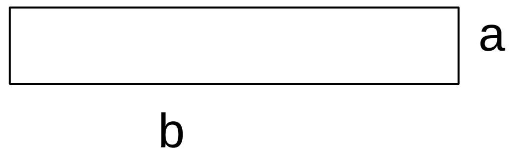
Aspect Ratio

Tri & Tetra	Inscribed Radius / Circum-radius	Ideal shape: Equilateral Δ	 r = a/2. tan30 R= a/2.cos30 r/R = ½ Aspect Ratio for Tri = 1/2.[R/r] Aspect Ratio for Tets = 1/3.[R/r]
Quad	MIN(Diagonals) / MAX(Diagonals)	Ideal Shape: Rectangle	 Aspect Ratio=MAX[a, b] / MIN[a, b]
Hexa	MIN(Edge Lengths) / MAX(Edge Length)	Ideal Shape: Cube, Cuboid	 Aspect Ratio = MAX [a, b, c, …, l] / MIN[a, b, c, …, l]

Mesh Generation: Aspect Ratio – Example

- As defined on the above slide, for an square and cube, Aspect Ratio = 1
- For rectangular size other than square (oblong shape) and Cuboids, aspect ratio can be defined as:

  A.R. = b / a



- For triangular shapes, aspect ration can be calculated with following expressions:

  r = 4.R.sinA/2.sinB/2.SinC/2 and Aspect Ratio = ½ R/r

  For equilateral triangles, A=B=C=60º, sin60/2 = ½

  ➔ Aspect Ratio = ½ . ¼. 1/[sin30.sin30.sin30] = ½. ¼. 1/ [½ . ½. ½] = 1.0

- **For Right-Angled triangles:**

  r = R.[sin(B)+ cos(B) – 1] ➔ Aspect Ratio = ½. 1/[sin(B)+cos(B) – 1]

Some pre-processors such as **ICEM records Aspect Ratio on the scale of 0 ~ 1**. Others such as **GAMBIT, HM, ANSYS** records them to the **scale of 1 ~ ∞.**

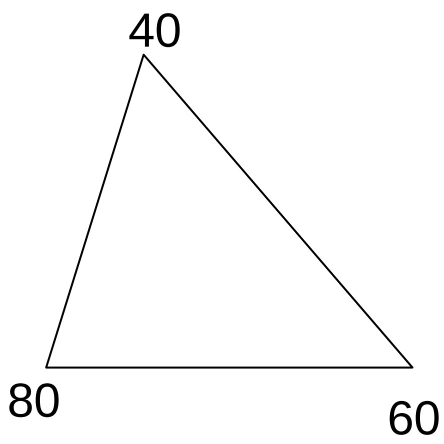
Mesh Generation: Internal Angle Deviation

- This is defined as deviation of angle from the ideal shape. Hence, for triangular and tetrahedral, it is defined as

  $\mathrm{MAX}\ [|60 - \theta_{MIN}|, |\theta_{MAX} - 60|]$

- For rectangles and hexahedrons, it is defined as

  $\mathrm{MAX}\ [|90 - \theta_{MIN}|, |\theta_{MAX} - 90|]$



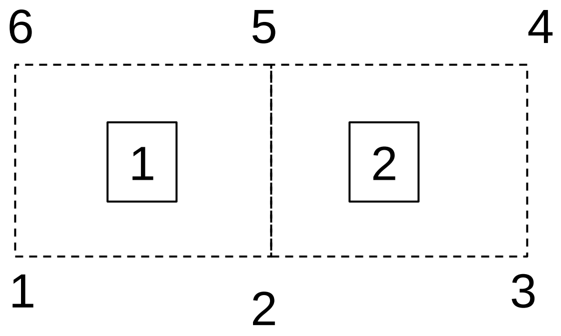
Internal Angle Deviation = MAX[60-40, 80-60] = 20º

The value of Internal Angle Deviation should be as close to zero as possible.

Mesh Generation: Face Handedness

- The nodes of elements follow a pre-determined sequence. Typically, the counter clockwise arrangement is said to have correct pattern. Hence, node connectivity of all the elements should follow the same pattern. For the following two elements, the correct arrangement is:

1	1	2	5	6
2	2	3	4	5



Fluent can display face handedness. Initialize the case, and then go to Display>Contours>Contours of Grid/Face Handedness. Cells with left- handed faces have a cell value of 1. Good cells have a face handedness of 0. That will allow you to find where the bad cells are.

An easier way of displaying the left-handed faces is marking (Adapt -> Iso-Value..) the cells using adaption registers, let's say with Iso-Min =0.5 and Iso-Max=1.5. That will mark the bad cells. If you set Options to “Filled” under Adaption Display Options, then you should easily see where the bad cells are.

Correcting face handedness: In Fluent, try Text User Interface (TUI) command. /grid/modify-zones/repair-face-handedness

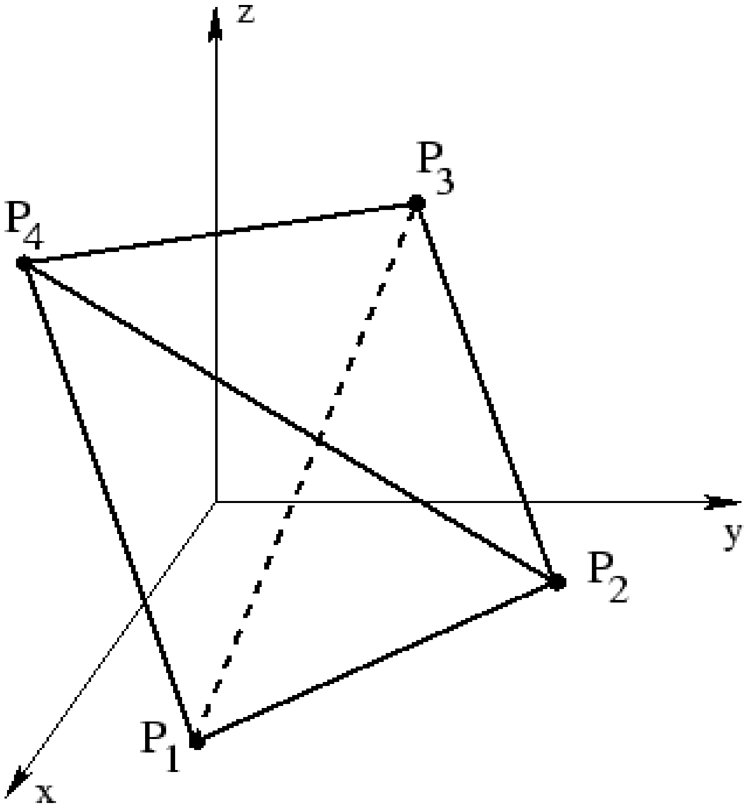
Mesh Generation: Element Type and Applicable Quality Parameters

Parameter	Quad	Tri	Hex	Tetra	Pyramid	Wedge
Area	y	y	x	x	x	x
Aspect Ratio	y	y	y	y	y	y
Diagonal Ratio	y	x	y	x	x	x
Edge Ratio	y	y	y	y	y	y
Equi-Angle Skew	y	y	y	y	y	y
Equi-Size Skew	x	y	x	y	x	x
Mid-Angle Skew	y	x	y	x	x	x
Stretch	y	x	y	x	x	x
Taper	y	x	y	x	x	x
Volume	x	x	y	y	y	y
Warpage	y	x	y	x	x	x

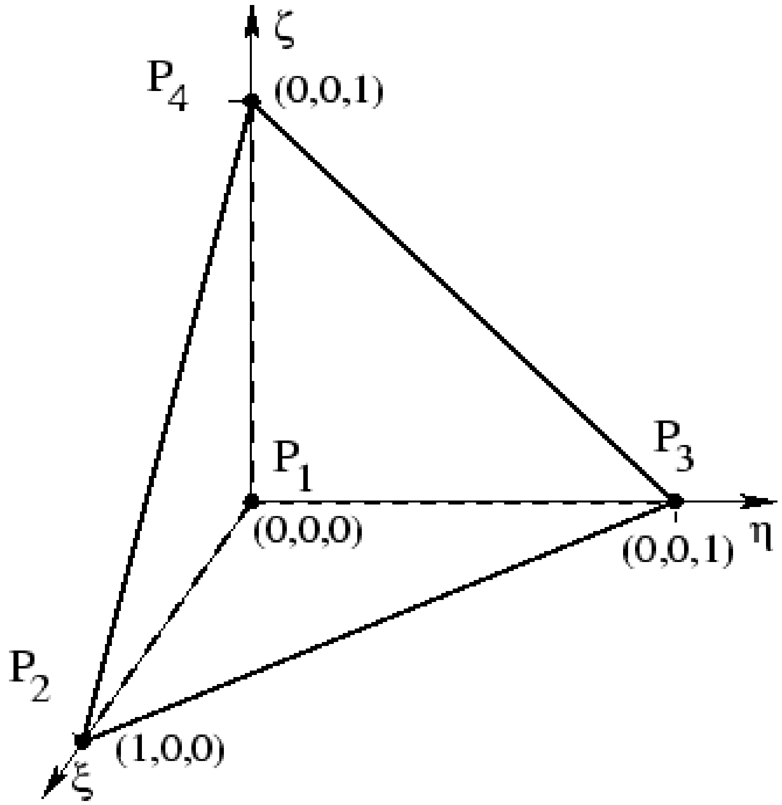
Legend: y→ Defined, x → Not defined

GAMBIT defines "Equi-Angle Skew" as default quality parameter for all elements. ICEM defines following combination as default quality parameters: **Tri/Tetra–Aspect Ratio, Hex/Quad–Determinant/Jacobian**

Mesh Quality Parameters - Jacobian



Tetrahedral element in (X,Y,Z) -coordinate system



Tetrahedral element in normalized (ξ, η,ζ) - coordinate system

$$x = x_1 + (x_2-X_1).\xi + (x_3-x_1)\eta + (x_4-x_1)\ \zeta$$

$$y = y_1 + (y_2-y_1).\xi + (y_3-y_1)\eta + (y_4-y_1)\ \zeta$$

$$z = z_1 + (z_2-z_1).\xi + (z_3-z_1)\eta + (z_4-z_1)\ \zeta$$

$$\mathbf{J} = \begin{vmatrix} x_2-x_1 & x_3-x_1 & x_4-x_1 \\ y_2-y_1 & y_3-y_1 & y_4-y_1 \\ z_2-z_1 & z_3-z_1 & z_4-z_1 \end{vmatrix}$$

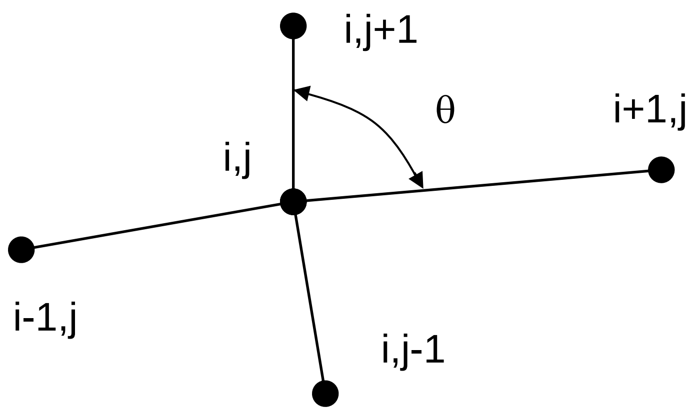
Mesh Quality Parameters - Jacobian

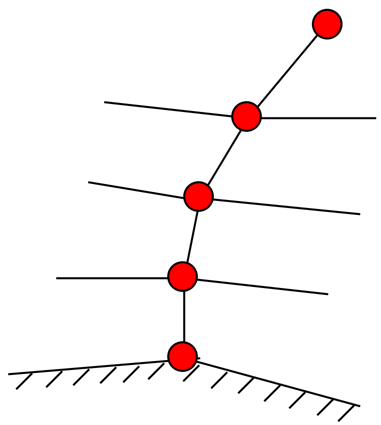
- Jacobian is defined at Element Vertex. When Jacobian matrix is square one, its determinant is called Jacobian determinant or simply Jacobian. In FE mesh, there exists an algebraic function F which maps Global co-ordinates (X, Y, Z) of nodes of a tetra or hex- elements to their local co-ordinate system (ξ, η,ζ). Used in isoparametric mapping it contains the information about the change in scales in the two coordinate systems.
- If Jacobian determinant is positive near a node p, the transformation matrix preserves its orientation near p and vice versa.
- The absolute value of the Jacobian determinant at **p** gives us the factor by which the function *F* expands or shrinks volume near **p**
- The Jacobian is a measure of how close an element is to a perfect shape. A perfect quad element is a square and has a Jacobian of 1.0. A perfect tri element is an equilateral triangle.

Determinant – (Smallest determinant of the Jacobian Matrix / Largest determinant of the Jacobian Matrix) where each determinant is calculated at the each node of the element

- In general, determinant value > 0.3 is acceptable to most solvers.
- Determinant: 3 x 3 x 3 Stencil: Same as 2 x 2 x 2 Stencil but edge mid-points of blocks are added to Jacobian Matrix.

Orthogonality: This quality parameter refers to perpendicularity of mesh with a wall. Grid *orthogonality* is the angle that a grid line makes with the other grid line makes with the other grid lines intersecting at a grid point. Orthogonality is defined so that $\theta \leq 90^o$.





Mesh Quality Parameters – Equi- Angle Skewness

- Qmax = Largest angle in the face or the element
- Qmin = Smallest angle in the face of the element
- Qequiv = Angle of a perfect element, 60 deg for tri / wedge and 90 deg for quad / hexa
- Equi-angle Skewness = 1 – MAX{(Qmax – Qequiv) / (180 – Qequiv), (Qequiv – Qmin) / Qequiv}

GAMBIT's categorization is as follows:

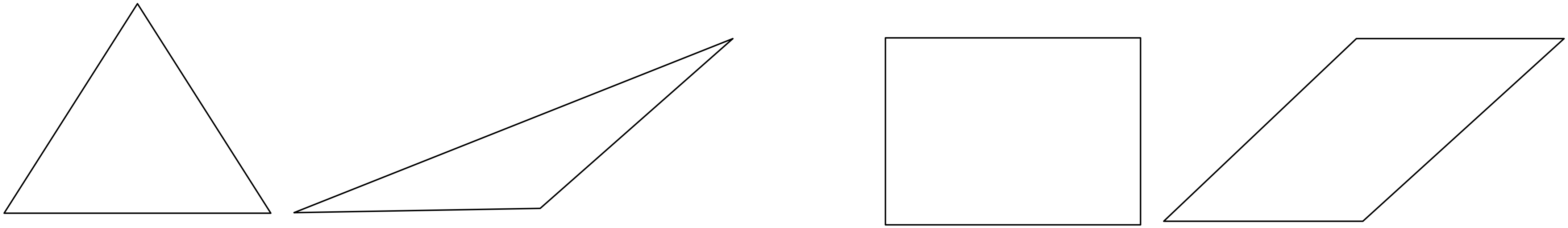
Range	GAMBIT/Fluent
0	Perfect
$0.0 < Q_{EAS} < 0.25$	Excellent
$0.25 < Q_{EAS} < 0.50$	Good
$0.50 < Q_{EAS} < 0.75$	Fair
$0.75 < Q_{EAS} < 0.90$	Poor
$0.90 < Q_{EAS} < 1.0$	Very poor (sliver)
$Q_{EAS} = 1.0$	Degenerate

- In general, high-quality meshes contain elements that possess average values of 0.1 (2-D) and 0.4 (3-D).
- There are two methods to measure skew: (1) Based on the equilateral volume (applies only to triangles and tetrahedra). (2) Based on the deviation from a normalized equilateral angle. This method applies to all cell and face shapes

Mesh Quality Parameters – Skew, Mid-node Angle

Skew:

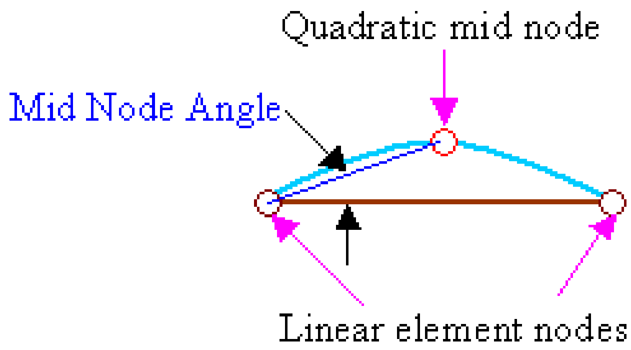
- **Hexa:** For all 6-faces, angle between face normal and vector define by face centres & hexahedral geometric centre is calculated. MAX angle is normalized such that: 1 → Perfect Cube / Cuboid, 0 → Degenerate Element
- **Tri:** Area of the element / area of a perfect equilateral triangle having same circum-circle
- **Quad:** Angle between vectors formed by connecting mid-points of the opposite sides, normalized by dividing with 180°



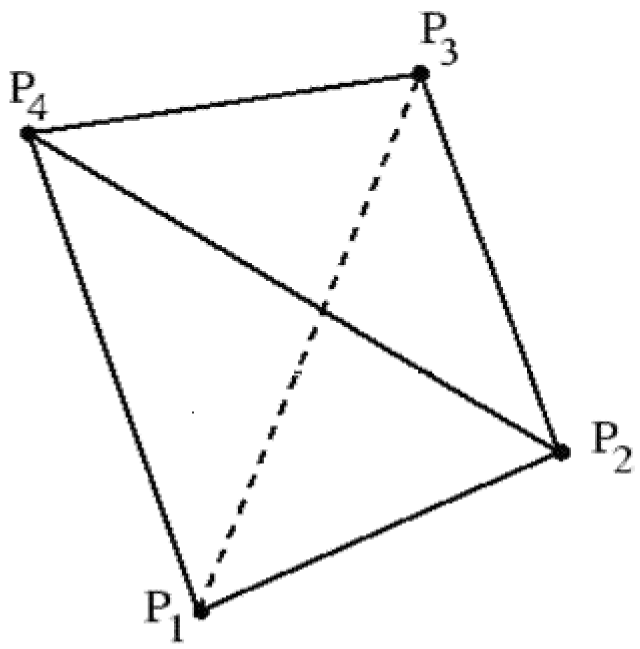
Equilateral Triangle **Highly-Skewed Triangle** **Equiangular Quad** **Highly Skewed Quad**

Mid Node Angle:

- Angle by which quadratic mid-node is off from linear edge



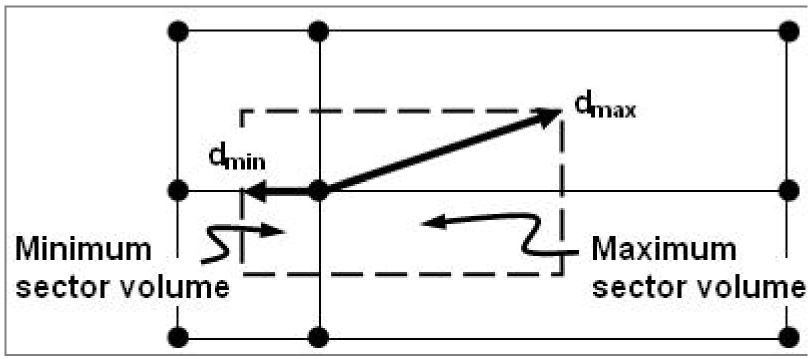
- **Warp:** It applies only to quadrilateral elements and is defined as the variation of normals between the two triangular faces that can be constructed from the quadrilateral face. The actual value is the maximum of the two possible ways triangles can be created.
- **Tetra Collapse:** Collapsed (flat) tetrahedral element will either prevent the solver code from running, or will give inaccurate results. This check computes the distance from the plane of each face of the tetrahedral element to the fourth node for that face. To normalize the value, the meshing software take the ratio of the longest to shortest value as the value to check for the collapse of the tetrahedral element. The default value in FEMAP is 10. In ICEM, it is termed as "Tetra Special", calculated as **"Largest Element Edge Length / the Smallest Height"**.

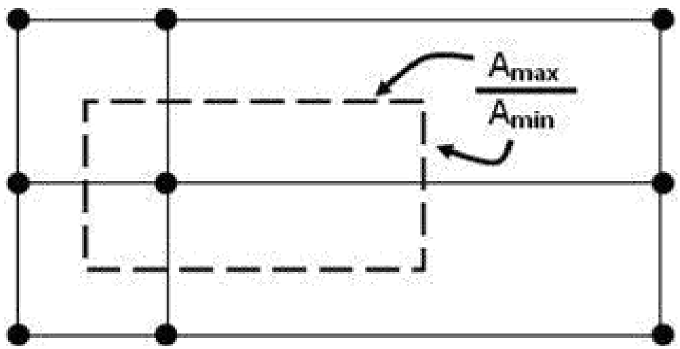


Tetra Special = MAX(a, b, c) / d

CFX – Mesh Expansion Factor

- It involves the ratio of the maximum to minimum distance between the control volume node and the control volume boundaries. Since this measure is calculated relatively expensive to for arbitrarily shaped control volumes, an alternative formulation, ratio of maximum to minimum sector volumes, is used. It involves the ratio of the maximum to minimum integration point surface areas in all elements. Nodal (i.e., control volume) values are calculated as the maximum of all element aspect ratios that are adjacent to the node.





Fluent – Squish Index

- **Cell Squish Index** is a measure of the quality of a mesh, and is calculated from the dot products of each vector pointing from the centroid of a cell toward the centre of each of its faces, and corresponding face area vector. Therefore, the worst cells will have a Cell Squish Index close to 1.

- **Face Squish Index** is a measure of the quality of a mesh, and is calculated from the dot products of each face area vector, and the vector that connects the centroid of the two adjacent cells. Therefore, the worst cells will have a Face Squish Index close to 1.

-- Axi-symmetric geometries must be defined such that the axis of rotation is the x axis of the Cartesian coordinates used to define the geometry. For axi-symmetric cases, during grid check, the number of nodes below the x axis is listed. Nodes below the x axis are forbidden for axi-symmetric cases, since the axi-symmetric cell volumes are created by rotating the 2D cell volume about the x axis; thus nodes below the x axis would create negative volumes.

-- The topological verification in Fluent for is checking the element-type consistency: If a mesh does not contain mixed elements (quadrilaterals and triangles or hexahedra and tetrahedra), FLUENT will determine that it does not need to keep track of the element types. By doing so, it can eliminate some unnecessary work.

-- FLUENT is an unstructured solver, it uses internal data structures to assign an order to the cells, faces, and grid points in a mesh and to maintain contact between adjacent cells. It does not, therefore, require i, j, k indexing to locate neighbouring cells. This gives you the flexibility to use the grid topology that is best for your problem, since the solver does not force an overall structure or topology on the grid. In 2D, quadrilateral and triangular cells are accepted, and in 3D, hexahedral, tetrahedral, pyramid, and wedge cells can be used. Both single-block and multi-block structured meshes are acceptable, as well as hybrid meshes containing quadrilateral and triangular cells or hexahedral, tetrahedral, pyramid, and wedge cells. In addition, FLUENT also accepts grids with hanging nodes (i.e., nodes on edges and faces that are not vertices of all the cells sharing those edges or faces). Grids with non-conformal boundaries (i.e., grids with multiple sub-domains in which the grid node locations at the internal sub-domain boundaries are not identical) are also acceptable.

--Although GAMBIT and TGrid can produce true periodic boundaries, most CAD packages do not. If your mesh was created in such a package, you can create the periodic boundaries using the non-conformal periodic option in FLUENT. This option, however, is recommended only for periodic zones that are planar.

--Grouping Elements to Create Cell Zones in Patran: Elements are grouped in PATRAN using the Named Component command to create the multiple cell zones. All elements grouped together are placed in a single cell zone in FLUENT. If the elements are not grouped, FLUENT will place all the cells into a single zone.

Though various pre-processing software such ICEM CFD, GAMBIT, Hypermesh calculates mesh quality parameters in a comparable fashion, the classification from bad to excellent are normally on opposite scales. For example, ICEM treats an element with Equi-Angle Skewness of 1.0 as "the Best", GAMBIT/Fluent considers the opposite. CFDyna.com suggest recording the mesh quality as per the two tables given below for each simulation so that the results can be compared when required.

Type	No. of Elements	Worst Equiangle Skewness	% of Elements with Skewness > 0.8 (in ICEM < 0.2)
Hexahedron:			
Tetrahedron:			
Prism/Wedge:			
Pyramid:			

Mesh Quality Measurements: Software Dependencies

Equi-angle Skewness Distribution		Aspect Ratio Distribution	
Fluent	ICEM CFD	Fluent	ICEM CFD
0.0~0.1	1.0~0.9	1.00~1.11	1.0~0.9
0.1~0.2	0.9~0.8	1.11~1.25	0.9~0.8
0.2~0.3	0.8~0.7	1.25~1.43	0.8~0.7
0.3~0.4	0.7~0.6	1.43~1.67	0.7~0.6
0.4~0.5	0.6~0.5	1.67~2.00	0.6~0.5
0.5~0.6	0.5~0.4	2.00~2.50	0.5~0.4
0.6~0.7	0.4~0.3	2.50~3.33	0.4~0.3
0.7~0.8	0.3~0.2	3.33~5.00	0.3~0.2
0.8~0.9	0.2~0.1	5.00~10.0	0.2~0.1
0.9~1.0	0.1~0.0	10.0 ~ ∞	0.1~0.0