Mesh Generation and Manipulation

Hrvoje Jasak

`hrvoje.jasak@fsb.hr`

FSB, University of Zagreb, Croatia

Complex Geometry Requirements

Computational Mesh

- A computational mesh represents a description of spatial domain in the simulation: external shape of the domain and highlighted regions of interest, with increased mesh resolution
- Mesh-less methods are possible (though not popular): the issue of describing the domain of interest to the computer still remains
- Mesh generation is the current bottle-neck in CFD simulations. Fully automatic mesh generators are getting better and are routinely used. At the same time, requirements on rapid and high-quality meshing and massively increased mesh size are becoming a problem

Complex Geometry Requirements

Routinely Used Mesh Size Today

- Small mesh for model experimentation and quick games: 100 to 50k cells. Fast turn-around and
- 2-D geometry: 10k to 1m cells. Low-Re turbulent simulations may require more, due to near-wall mesh resolution requirements
- 3-D geometry: 50k to several million cells
- Complex geometry, 3D, industrial size, 100k to 10-50 million cells. Varies considerably depending on geometry and physics, steady/transient flow etc.
- Large Eddy Simulation (LES) 3-D, transient, 1-10 million cells. LES requires very long transient runs and averaging (20-50k time steps), which keeps the mesh resolution down
- Full car aerodynamics, Formula 1: 20-200 million cells for routine use. Large simulations under discussion: 1 billion cells!
- On very large meshes, problem swith the current generation of CFD software becomes a limiting factor: missing parallel mesh generation, data file read/write, post-processing of results, hardware and software prices

Handling Complex Geometry

Complex Geometry Description

- In aerospace applications, geometrical information is usually available before the simulation. In general, this is not the case: for simple applications, a mesh may be the only available description of the geometry
- Domain description is much easier in 2-D: real complications can only be seen in 3-D meshes
- Geometrical data formats
  - 2-D boundary shape: airfoils. Usually a detailed map of $x - y$ locations on the surface. Sometimes defined as curve data
    `http://www.ae.uiuc.edu/m-selig/ads/coord_database.html`
  - Stereo Lithographic Surface (STL): a surface is represented by a set of triangular facets. Resolution can be automatically adjusted to capture the surface curvature or control points. Creation of STL usually available from CAD packages
  - Native CAD description: Initial Graphics Exchange Specification (IGES), solid model etc. In most cases, the surface is represented by Non-Uniform Rational B-Splines or approximated by quadric surfaces. Typically, both are too expensive for the manipulations required in mesh generation and either avoided or simplified

Mesh Generation

Phases of Mesh Generation

- **Geometry clean-up**
  - Very rarely is the CAD description built specifically for CFD – in most cases, CAD surfaces (wing, body, nacelle) are assembled from various sources, with varying quality and imperfect matching. Surface clean-up is time-consuming and not trivial
- **Feature removal**
  - CAD description or STL surface may contain a level of detail too fine to be captured by the desired mesh size, causing trouble with 3-D mesh generation. Feature removal creates an approximation of the original geometry with the desired level of detail
- **Surface mesh generation**
  - In cases where the surface description is not discrete, a surface mesh may be created first
  - STL surface is already a mesh. It may be necessary to additionally split the surface for easier imposition of boundary conditions: inlet, outlet, symmetry plane etc.
  - Surface mesh is usually triangular or quadrilateral. There are potential issues with capturing surface curvature: surface mesh will be considered “sufficiently fine”

Mesh Generation

Phases of Mesh Generation, (cont'd)

- **Volume mesh generation**
  - The main role of the volume mesh is to capture the 3-D geometry
  - The cells should not overlap and should completely fill the computational domain. Additionally, some convexness criteria (FVM) or a library of pre-defined cell shapes (FEM) is included.
  - Computational mesh defines the location and distribution of solution points (vertices, cells etc.) Thus, filling the domain with the mesh is not sufficient - ideally some aspects of the solution should be taken into account.
  - *A*-priori knowledge of the solution is useful in mesh generation. Trying to locate the regions of high mesh resolution ("fine mesh") to capture critical parts of the solution: shocks, boundary layers and simular
  - Quality of the mesh critical for a good solution and is not measured only in mesh resolution
  - Mesh quality measures depend on the discretisation method
    - Cell aspect ratio
    - Non-orthogonality
    - Skewness
    - Cell distortion from ideal shape
    - . . . etc.

Mesh Structure and Organisation

Influence of Mesh Structure

- Some numerical solution techniques require specific mesh types. Example: Cartesian meshes for high-order finite difference method
- Supported mesh structure may severely limit the use of a chosen discretisation method
- With mesh generation as a bottle-neck, it makes sense to generalise the solver to be extremely flexible on the meshing side, simplifying the most difficult part of the simulation process

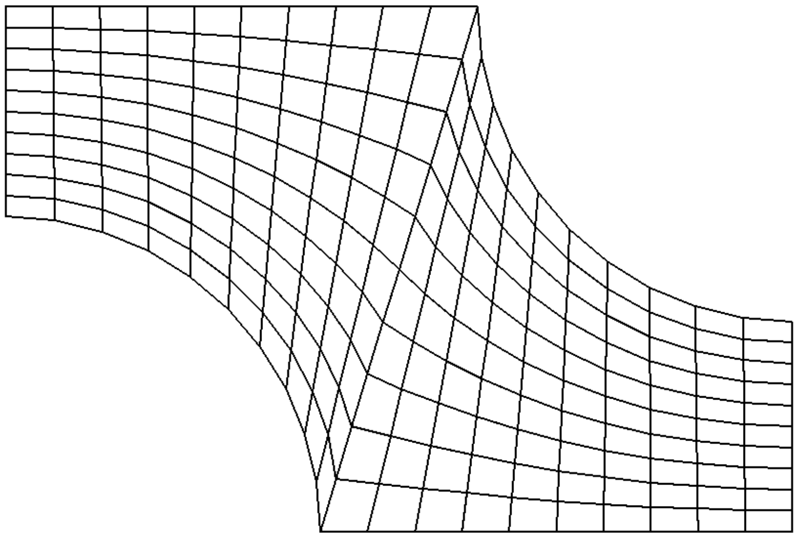
Mesh Types

- Cartesian mesh
- Structured body-fitted mesh
- Multi-block mesh
- Unstructured shape-consistent meshes
- Tetrahedral and hybrid tet-hex mesh
- Overset and Chimera meshes
- Polyhedral meshes

Mesh Structure: Body-Fitted

Structured Body-Fitted Mesh

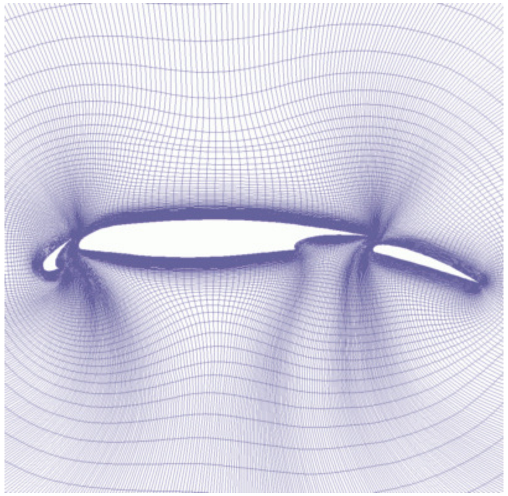
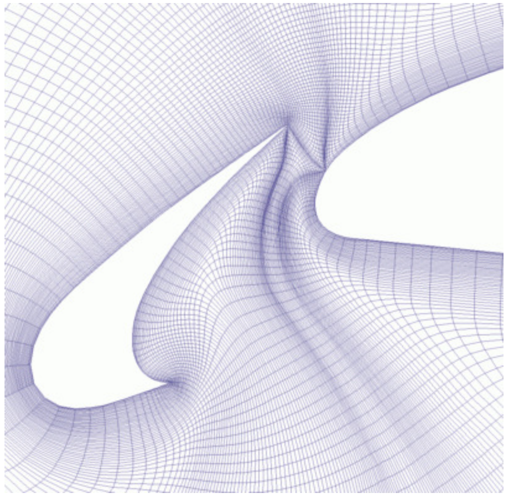
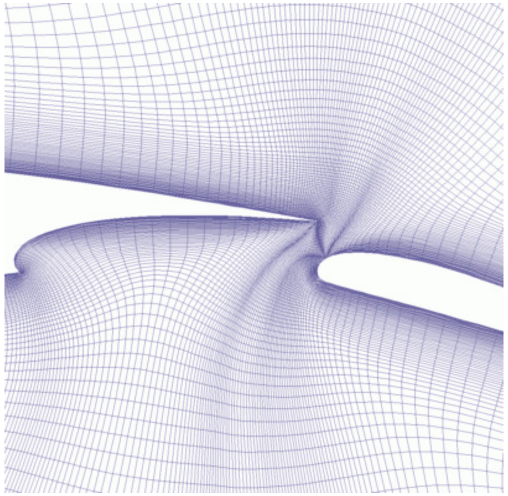
- Body-fitted meshes originate from the non-orthogonal curvilinear coordinate system approach. The case-specific coordinate system is created to fit the boundary
- The mesh is hexahedral and regularly connected. Real geometry can be captured but with insufficient control over local mesh resolution



- The use of contravariant coordinates for the solution vectors was quickly abandoned

Mesh Structure: Multi-Block

- Mesh created as a combination of multiple body-fitted blocks. All block and cells are still hexahedral. In FVM, special coding on block interfaces
- More control over mesh grading and local resolution. This can capture significantly more complex geometry
- Extensive use of smoothing and mesh optimisation tools; initial block decomposition still done by hand
- Mesh generation in 3-D for relatively complex shapes is still hard and time-consuming: mesh interfaces need to match

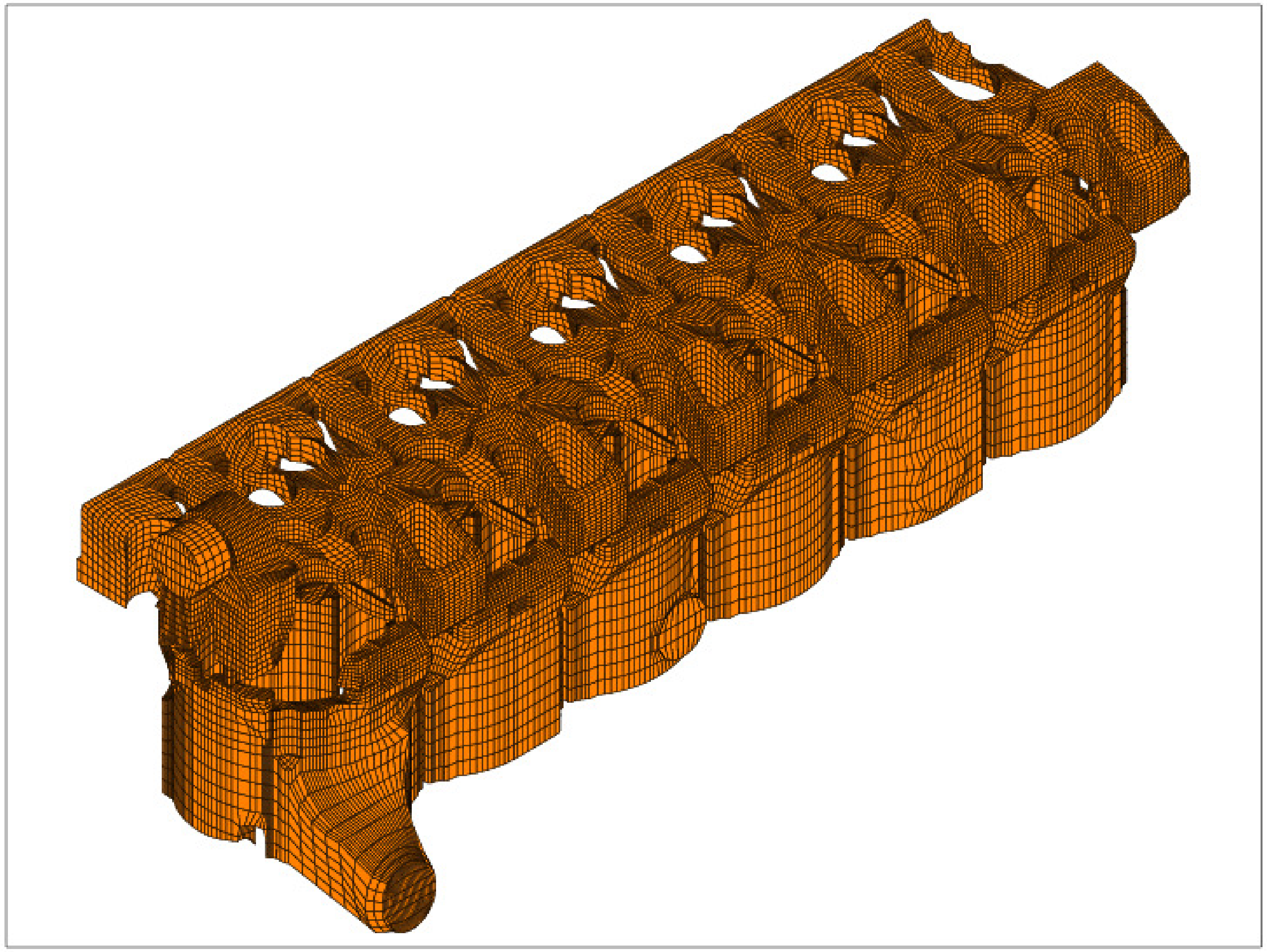




Mesh Structure: Unstructured

- Block connectivity above introduces the concept of **storing mesh connectivity** rather than calculating it: unstructured mesh
- Loose definition of connectivity allows more freedom: hexahedral and degenerate hexahedral meshes: prisms, pyramids, wedges etc. allow easier meshing
- From the numerical simulation point of view, this is a major step forward. Geometries of industrial interest can now be tackled with a detailed description, which satisfies the design engineer
- At this stage, numerical simulation in an industrial setting really takes off. Handling airfoils and single wing or even wing-fuselage assembly is not too difficult. Hand-built meshes for a complete aircraft are still quite difficult

Mesh Structure: Unstructured

At this stage, all meshes are hand-built. Complex 3-D mesh takes 2-3 months



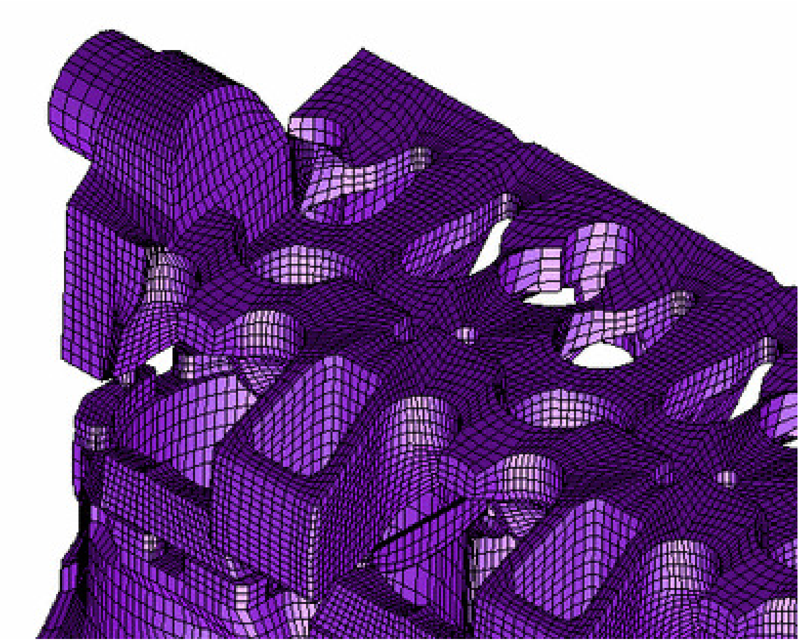
Mesh Structure: Tetrahedral

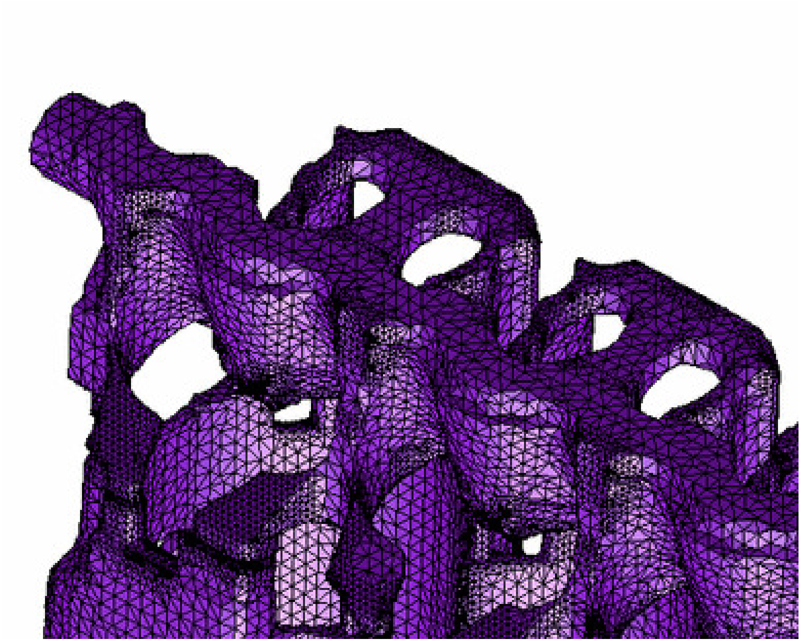
Automatic Meshing: Tetrahedral Mesh

- Tetrahedral mesh are not good from the numerics point of view
- ...but they could be generated automatically!
- In a solver can support tetrahedral meshes, mesh generation time for complex geometry reduces from weeks to hours.
- Great saving in mesh generation effort, faster turn-around of simulations and geometrical variation, mesh sensitivity studies can be performed on realistic geometries
- Tetrahedra are particularly poor in boundary layers close to walls. A **hybrid mesh** is built by creating a layered hexahedral mesh next to the wall. The rest of the domain is filled with tetrahedra. A combined tet-hex mesh is a great improvement in quality
- On the negative side, cell count for a tetrahedral mesh of equivalent resolution is higher than for hexahedra. A part of the price is paid in lower accuracy of the solver on tetrahedra: limited neighbourhood connectivity.

Mesh Structure: Tetrahedral

Automatic Meshing: Tetrahedral Mesh





Tetrahedral Mesh Generation

- **Advancing front method**: starting from the boundary triangulation, insert tetrahedra from the live front using priority lists
- **Delaunay triangulation**: point insertion and re-triangulation. The initial mesh is created by triangulating the boundary. New points are added in a way which improves the quality of the most distorted triangles and creates a convex hull around each point

Mesh Structure: Polyhedral

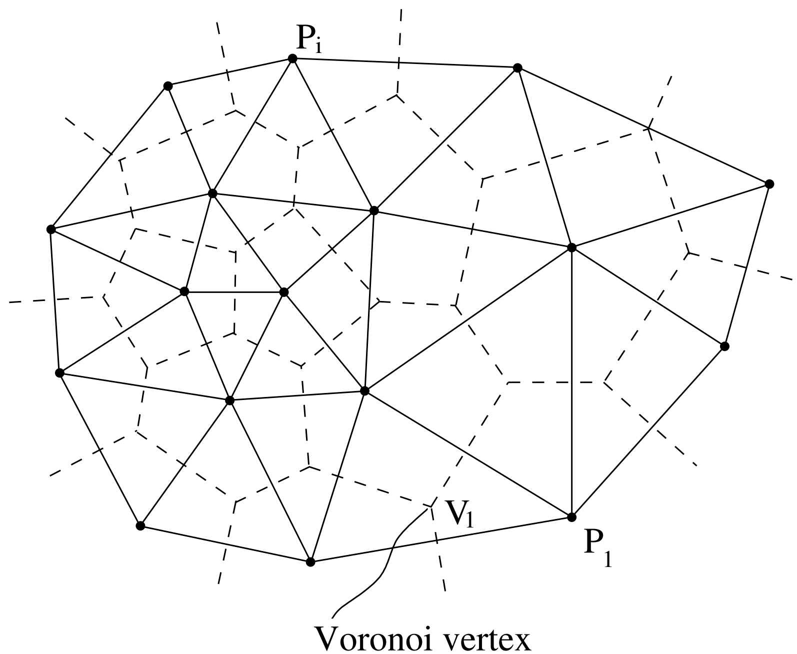
Automatic Meshing: Polyhedral Mesh

- In spite of automatic generation techniques, tetrahedral meshes are not of sufficient quality for industrial use. On the other hand, automatic hexahedral mesh generation has proven to be extremely challenging
- Finite Volume discretisation is not actually dependent on the cell shape: unlike FEM, there are no pre-defined shape functions and transformation tensors. This brings the possibility of **polyhedral mesh support**
- Finite Volume discretisation algorithm is reformulated into loops over cells and faces (still doing the same job)
- Polyhedral meshes are considerably better than tetrahedra, can be manipulated to be predominantly hexahedral, orthogonal and regular and can be created automatically

Mesh Structure: Polyhedral

Automatic Meshing: Polyhedral Tessalated Mesh

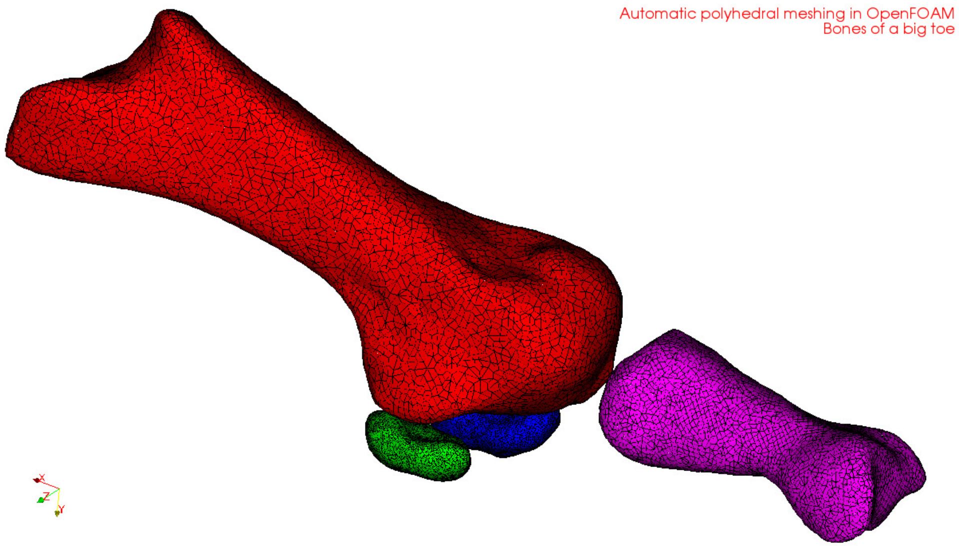
- The Delaunay triangulation algorithm introduces points on proximity rules. During the creation of the mesh, a dual mesh of convex polyhedra is created and cane be extracted.
- Interaction on the tessalated mesh and the boundary needs to be recovered after polyhedral mesh assembly
- Local control of mesh size achieved in the same way as in tetrahedral meshes

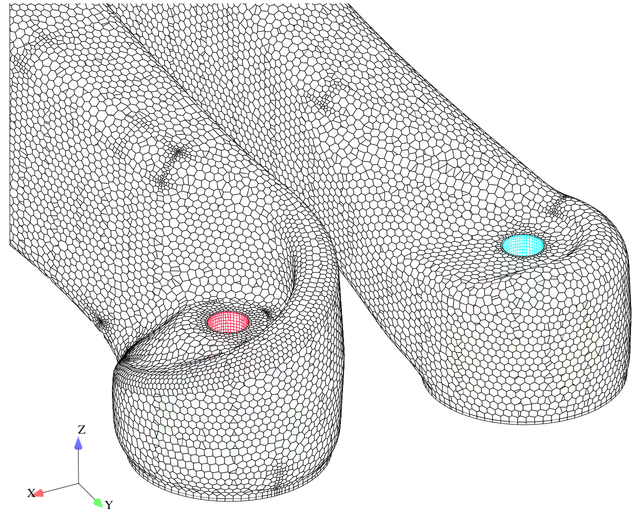


Mesh Structure: Polyhedral

Automatic Meshing: Polyhedral Tessalated Mesh

- Tessalated mesh in 3-D is created mainly of dodecahedra: space-filling shape
- Prismatic near-wall layer will consist of (hexagonal) prisms





Mesh Structure: Cut Cell

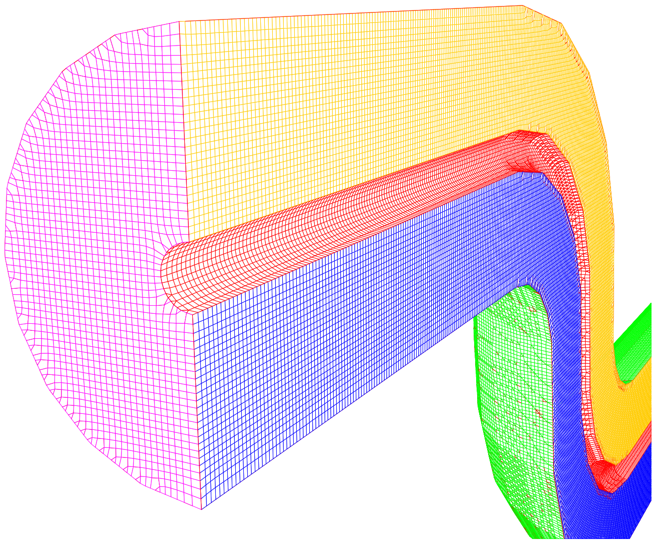
Automatic Meshing: Cut Cell Meshes

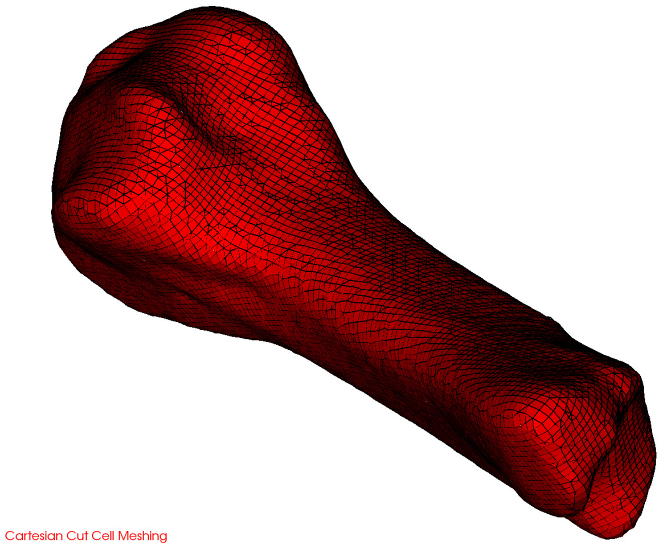
- Most of mesh generation is straightforward: filling space with non-overlapping cells. Even close to boundaries, it is easy to build high quality layered structure
- Problematic parts of mesh generation are related to interaction of advancing generation surfaces or boundary interaction in complex corners of regions where the mesh resolution dos not match the level of detail on the boundary description.
- **Cut cell technology** creates a rough mesh background mesh, either uniform hexahedral or capturing major features of the geometry. The mesh inside of the domain is kept and the one interacting with the boundary surface is adjusted or cut by the surface
- In some cases, the background mesh resolution can be automatically adjusted around the surface to match the local resolution requirements
- Meshes are good quality and generation is fast. Prismatic boundary layers may also be added. In some cases, background mesh adjustment or concave cell corrections are required.

Mesh Structure: Cut Cell

Automatic Meshing: Cut Cell Meshes

- Bulk of the mesh is created of uniform high-quality cells
- Background resolution is easily controlled by the starting mesh
- Before the surface cutting operation, the background mesh is locally refined to match the surface features
- Polyhedra are concentrated on the boundary, sometimes creating small cells
- Algorithm becomes more complex if prismatic layers are needed next to the wall

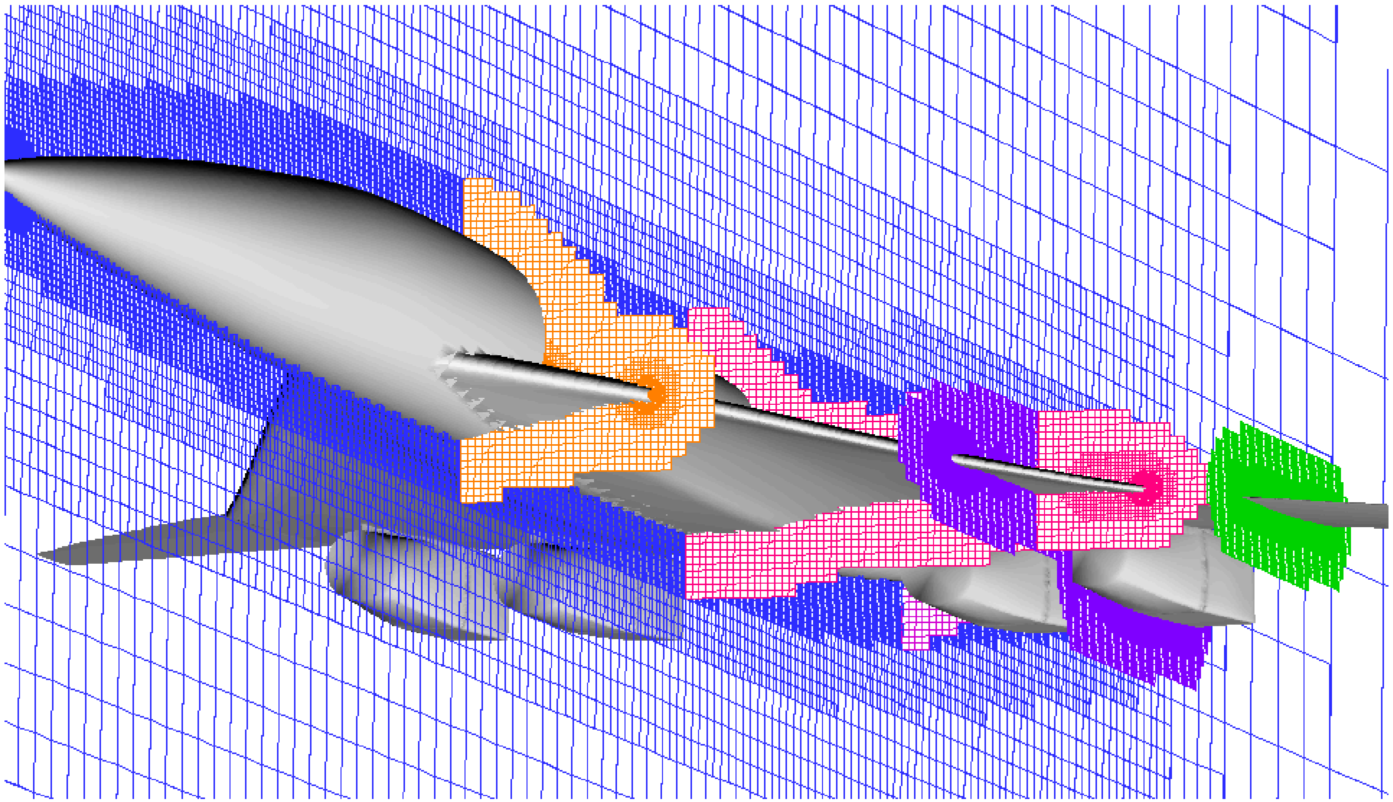




Mesh Structure: Cut Cell

Automatic Meshing: Cut Cell Meshes

- Before the surface cutting operation, the background mesh is locally refined to match the surface features
- Polyhedra are concentrated on the boundary; in some cases, very small cells are created



Manual Meshing: Airfoils

Mesh Structure for 2-D Airfoils

- Manual meshing of airfoil profiles really belongs to the past; it is still indicative to show how mesh handling governs the use of CFD
- O-mesh: NACA0012 example
- C-mesh: NACA32012 example, prettier in raeProfile
- H-mesh
- Hybrid mesh structure: triangular mesh with prismatic layers: twoElement
- Adapting to the geometry: transfinite mapping techniques
- Adapting to the solution: shock capturing with r-refinement
- Meshing multi-element airfoil configurations

Adaptive Mesh Refinement

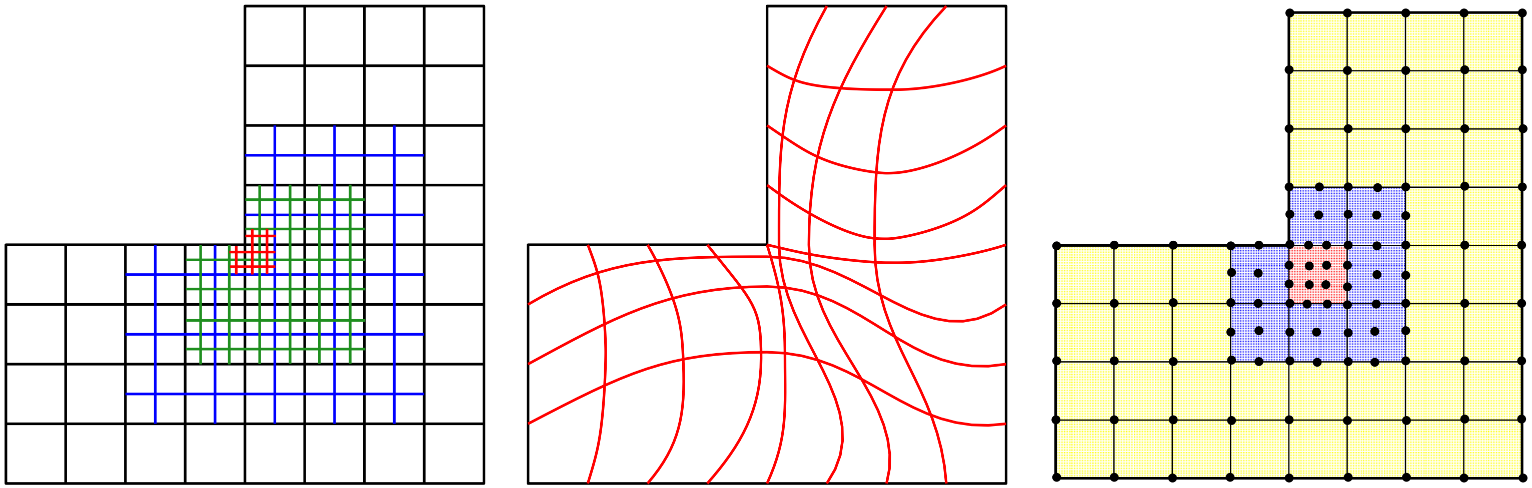
Mesh Resolution

- Mesh structure specifies where the computational points are located. Discretisation practice postulates the shape of solution between the computational points, which is the main source of discretisation error
- A sensible meshing strategy requires high resolution in **regions of interest** instead of uniformly distributing points in the domain. This implies some knowledge of the solution during mesh generation.
- The same can be achieved in an iterative way
    1. Create initial mesh and initial solution
    2. Examine the solution from the point of view of accuracy or resolution in “regions of interest”
    3. Based on the available solution, adjust mesh resolution in order to improve the solution in the selected parts of the domain
    4. Repeat until sufficient accuracy is achieved or computer resources are exhausted
- Performing mesh improvement by hand is tedious and time-consuming. For an automatic procedure, two questions need to be answered:
    - Where to refine the mesh (adjust resolution)?
    - How to change the mesh to achieve the required accuracy

Adaptive Mesh Refinement

Types of Mesh Refinement

- Global refinement: mesh sensitivity studies
- h-refinement: introducing new computational points in regions of interest
- r-refinement: re-organise the existing points such that more points fall into the region of interest
- p-refinement: enriching the space of shape functions in order to capture the solution more closely



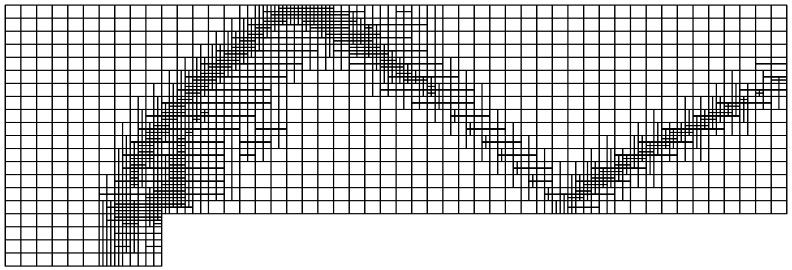
Adaptive Mesh Refinement

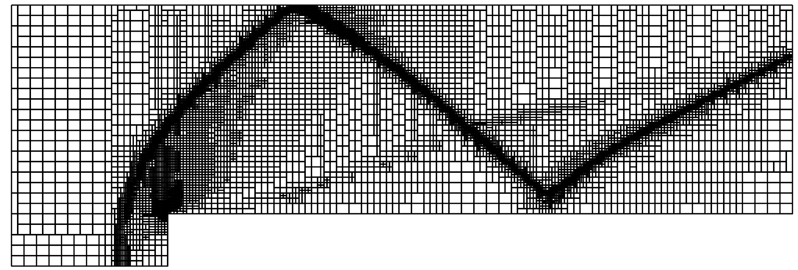
Automatic Mesh Refinement

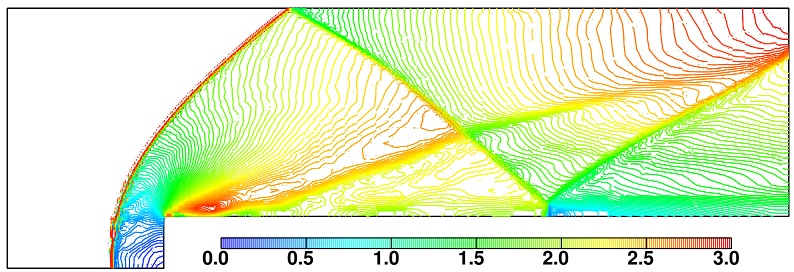
- **Error indicators**: highlight regions of interest. Example: magnitude of the second pressure gradient, Mach number distribution etc.
- **Error estimates**: apart from the spatial information (error distribution), they provide guidance on the absolute error level
- Traditionally, mesh adaptation was a part of the CFD solver instead of mesh generator. In cases where the refinement algorithm resorts to cell splitting, we may end up with a faceted surface representation instead of a smooth surface, which compromises the results.
- Solution: geometrical description of the boundary needs to be available from the solver instead of trying to recover the data from the original (coarse) mesh
- A further step is related to the specification of boundary conditions. In, for example, wind tunnel simulations, the velocity and turbulence at the inlet plane in shown from the measured data and interpolated onto the inlet patch of the mesh. Ideally, the boundary condition should be associated with space or with the boundary description, avoiding problems with interpolation. This leads to issues of CAD integration, which is beyond our scope

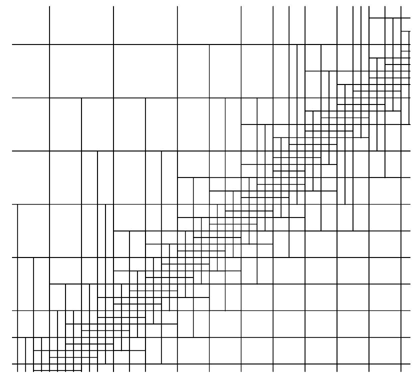
Adaptive Mesh Refinement

Mesh Adaptivity on Shocked Flows





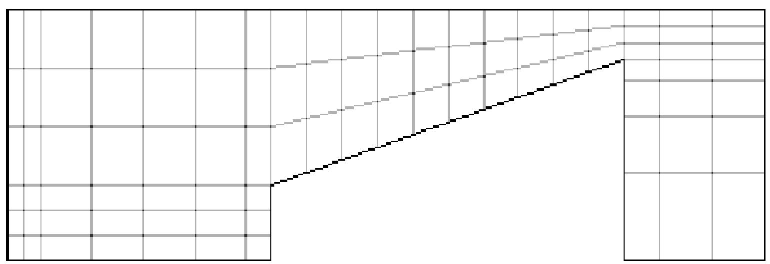


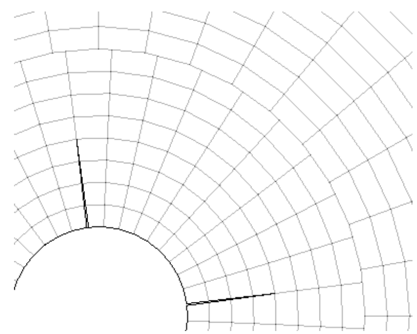


Dynamic Mesh Handling

Topological Mesh Changes

- In cases of extreme shape change, moving deforming mesh is not sufficiently flexible: deforming the mesh to accommodate extreme boundary deformation would introduce high discretisation errors
- Mesh motion can be accommodated by adding or removing computational cells to accommodate the boundary deformation. This is associated with higher discretisation errors and complications in the algorithm, but is sometimes essential
- Common types of topological changes:
  - Attach/detach boundary
  - Cell layer addition/removal
  - Sliding interface





- Typically, several topological modifications are used together