Birds of a Feather Sessions Summary

BoF #1: Challenges in Geometry Modeling

Facilitated by: Nigel Taylor, MBDA UK Ltd

This topic attempted to address the challenges encountered during creating, preparing and sharing geometry for meshing. More specifically:

Creating Geometry (facilitated by William Jones, NASA Langley): The scope of topics to be addressed by this group includes (but is not limited to): methods of construction; modeling platforms and their constraints; model parameterization; topology choices; modeling provenance, checking prior to release.

Preparing Geometry (facilitated by Mark Gammon, ITI): The scope of topics to be addressed by this group includes (but is not limited to): model repair; model de-featuring; model simplification; Outer Mold Line manipulation.

Sharing Geometry (facilitated by John Dannehoffer III, Syracuse University): The scope of topics to be addressed by this group includes (but is not limited to): converting models from one format to another, either from a proprietary format (e.g. MCAD-CATIA, CREO, NX, etc.) to a non-proprietary format (IGES, Parasolid, STEP, etc.), or vice-versa; use of modeling platforms different from that in which the model was originally created; handling models supplied by third parties.

Creating Geometry

The discussion focused on the following areas: the starting point for geometry generation, the intended purpose of geometry and the Cad process.

1. What is the starting point?
   - Drawings, digitized data, analytic expressions?
   - Usually the assumption is that CAD is the starting point, which is a bad one.
   - The type of data capture employed is important to understanding CAD.
2. Intended purpose
   - Manufacturing, CFD, CSM?
   - Does it need to be simplified for its purpose?
   - Should we start with simpler mathematical forms and build up the geometry?
   - If it is for CFD analysis, typically there is mostly interest in OML.
3. The CAD process
   - is based on abstract notion
   - never starts clean
   - not parameterizable in most cases.
   - Often someone digitizes a new part and adds it to existing CAD.

Preparing Geometry

The discussion focused on the following points:

- Techniques should be developed to distinguish between need for CAD repair or need for alternate meshing strategies.
- Determining appropriate tolerancing is routinely an issue (example: gap between surfaces larger than desired surface mesh spacing).
- Automation is of high priority for defeaturing and OML extraction
- Tangential intersections are unrepairable
- Grid refinement typically reveals geometry issues (example: same as above)
- User Boundary Condition tags often lost/altered during geometry repair
- It's not possible to satisfy all end users from multiple disciplines with one master geometry and a common "fix"
- Scan data repair hard to automate and difficult to handle; no topology in CAD files, no parameterization methodology.
- CAD vendors should be engaged to help resolve these problems.

Sharing Geometry

The discussion focused on the following areas: converting, versioning, formatting differences and third party representations.

1. Converting
    - There is no Validation & Verification process for geometry
    - Maintaining in the geometry files the desired tolerance for meshing purposes is a big issue
    - The topological information for the geometry is often lost between the CAD system and the mesh generator
    - Direct access of geometry by the mesh generator via a kernel should be considered instead of exchange files.
    - File/part heritage should be conveyed/retained in the geometry file or via kernel, so users can tell how surfaces were created/developed.
2. Versioning
    - Particularly useful for propagating/journaling changes
    - Should take into account purpose of geometry; separate manufacturing from CFD meshing versions
3. Geometry exchange can propagate differences due to
    - underlying math
    - undocumented internal process
    - repeatability issues
    - tessellated surface surrogates
4. Issues related to third party representations
    - Surface attribution (process and software)
    - Body deformation info (for aero-elastic applications)

- Dealing with construction entities
- Variety of parametric models (families)
- Cost quantification
- Problems remain hidden until use

BoF #2: Fundamental Meshing Issues

Facilitated by: Konstantinos Vogiatzis, Engility Corp./DoD HPCMP PETTT

This topic attempted to address the fundamental issues encountered during mesh generation pertaining to surface and volume meshing as well as mesh adaptation for both structured and unstructured meshes with various element types. More specifically:

Surface meshing (facilitated by James Masters, NAS): Issues addressed may include initial surface discretization type (parameterized patches, tessellation), large number of patches, slivers, maintaining initial CAD edges/angles, surface validity during remeshing (non-manifold, free edges, pierced faces), surface smoothing, surface resolution and transition.

Volume meshing (facilitated by Peter Eiseman, Program Development Corp): This sub-topic may focus on BL meshing (first layer thickness, aspect ratios, number of layers, layer growth, thickness control, highly convex/concave regions), BL to volume mesh transition, volume mesh density and growth, hybrid meshing.

Adaptive meshing (facilitated by Todd Michal, Boeing): Issues related to automatic and manual mesh adaptation. Discussion may include topics such as error estimates for adaptation, feature identification, methods for manual and automatic mesh modification on various grid types (unstructured, overset, structured, etc.), and geometry requirements for mesh adaptation.

Surface Meshing

The discussion focused on the following areas: purpose, surface-volume mesh coupling, automation, surface definition, and surface validity.

1. Purpose. Surface mesh resolution maybe be driven by
   - physics (targeted phenomena) and/or
   - geometric concerns and limitations (size/number of features to resolve).
2. Surface-volume mesh coupling
   - Limitations are imposed due to behavior of BL/strand meshes in highly concave/convex areas and where proximity refinement is required. This in turn affects the desired thickness of BL mesh and the volume mesh growth rate.
3. Automation

- The need and frequency of spatial control for surface remeshing and resolution dictates some degree of automation.
- Edge insertion for patch splitting and control volume definition (cylinders, boxes, spheres) are mostly done manually and after visual inspection of the initial surface.
- Anisotropic meshing still requires some iterations before the desired mesh resolution is obtained.

4. Surface definition: Views were exchanged on how different software handle large number of patches and slivers that are not required for proper surface definition but are products of geometry creation/assembly (edge/patch merging etc)
5. Surface validity/quality. Maintaining surface validity and quality requires access to the underlying geometry definition.
   - Projection to CAD
   - Surface checks during remeshing
   - Define CAD-based remeshed surface quality metrics

Volume Meshing

The discussion focused on the following areas: automation, control and size.

1. Mesh Automation
   - Need to minimize the human error (which is a given), especially when a large number of meshes is required. The major issue for automation is how control is achieved.
2. Mesh Control
   - Curvature/BL information from surface geometry and mesh
   - Wake location/shock location – mesh alignment through surface topology information
   - Impose volume density control sources
   - Mesh - boundary condition coupling
3. Mesh Size
   - Currently at 10^10/10^11 elements. The future goal is 10^12, which can only be attained through parallel mesh generation (HPC).

Adaptive Meshing

The discussion focused on the following areas: usage limitations , criteria for adaptation, efficiency, geometry handling and mesh quality.

1) Why is adaptation not widely used?
   a) Availability – adaptation methods are not widely available. Few commercial tools have adaptive capability or existing capabilities are limited.
   b) Solver robustness – many flow solvers are not robust running on adapted meshes
   c) Mesh reuse limitations – adapted meshes are generated for each run condition. This can be expensive. Not able to generate one mesh and reuse for many flow conditions.
   d) Lack of computational resources – not enough resources to run adapted meshes, particularly when filling a database requiring hundreds or thousands of solutions.
   e) Verification and Validation
   f) Reluctance to invest effort to new techniques ("why should I switch"?).

2) Criteria for adaptation
   a) A priori - user decides where to adapt mesh and manually defines regions to refine.
   b) Error Estimates – adapt mesh (coarsen/refine) based on local solution error.
   c) HPR – Adaptation could involve mesh resolution (H), spatial discretization algorithm order (P) or moving nodes (R).

3) Efficiency
   a) There is a cost to modify the mesh. This can hopefully be offset by converging the grid to an answer with fewer degrees of freedom.
   b) Difficulty to compare adapted results to fixed grid results. Accuracy is unknown for fixed grid solution so how do you obtain a comparable adapted solution for comparison?
   c) Adaptation is still new. No single-best approach to gain maximum efficiency.
   d) Hardware has improved so adaptivity does not necessarily increase productivity (a brute force, very fine fixed mesh may be faster than adapting)
   e) Mesh solver compatibility - adapted meshes can slow solution convergence
   f) Mesh topology

4) Geometry Handling
   a) Adapting the surface mesh requires access to the geometry definition.
   b) Need to track geometry association (e.g. parametric u/v) for each node in surface mesh
   c) Process needs to be coupled end to end (geometry association can not be lost at any step along the way)
   d) Geometry tolerance requirements are much higher for adapted meshes as the mesh may refine to a size that is comparable to or finer than flaws in the geometry. Adaptive methods must be fault-tolerant to geometry flaws.

5) Adapted Mesh Quality Criteria
   a) Solver convergence. Error must be reduced asymptotically with adaptation.
   b) What's good enough? How do we decide when to stop adapting? Is the mesh good enough for the analysis? What level of error remains in the solution?
      i) Error estimates can provide some of this information.

ii) User must be able to identify level of accuracy that is needed (not always known).

BoF #3: Standards and Compatibility

Facilitated by: Hugh Thornburg, Engility Corp./DoD HPCMP PETTT

This session was intended to address issues with standards, best practices and guidelines for the construction of a mesh, the storage, transfer and manipulation of the mesh and the suitability of the mesh for the given purpose. More specifically:

Meshing guidelines(facilitated by Carolyn Woeber, Pointwise): This topic will focus on the meshing guidelines provided by the High Lift Prediction Workshop organizers. The guidelines were intended to provide information to guide the construction of meshes with the desired properties. The focus will be on the utility of the guidelines in sufficiently capturing the required mesh characteristics, and communicating and guiding the subsequent construction.

Fitness of the Mesh for the Intended Purpose (facilitated by Carl-Olivier Gooch, UBC): This topic will focus on techniques for analysis of the baseline meshes. For example accurate representation of the provided geometry, validity of elements, and adherence to resolution requirements.

Meshing Portability (facilitated by Nicholas Wyman, Pointwise): This topic will focus on standards/formats for transferring mesh output from a given mesh generator to various modeling tools, and tools for analysis and manipulation/editing of the mesh. The session will also discuss version control and adherence to standards or tools such as CGNA and HDF5.

Meshing Guidelines

The discussion focused on the following points:

- Meshing guidelines should be evidence based.
- Any deviations should have specific justifications.
- Performing meshing studies on the effects of selected guidelines, including time to implement, would be beneficial:
    - Constant vs. variable boundary layer growth rate
    - Number of layers of constant height cells off the wall
    - Constant vs. scaled growth rate for a grid family
    - Effect of number of grid layers used in boundary layer mesh
    - Effect of cell type on boundary layer
    - Area Ratio variations for surface mesh
    - Aspect Ratio variations for surface mesh
    - Effect of spacing at element root/tips
    - Cell-centered vs. node-centered meshes
- What is the best starting point for generating a mesh family according to guidelines?
    - Guidelines provide medium grid specifications but problems often occur when that mesh is refined for fine and extra fine levels.

- Guidelines say that flow aligned cells should be provided in wake regions but there was no experimental visualization or geometry provided to indicate where the wake occurred for configuration.
- Specifying the chord and spanwise spacing based on the local device chord over-constrained the meshes from two standpoints:
    - Creation of high aspect ratio cells around TE.
    - The local device chord for the flap was different than that for the wing at the root. The resulting cells were mismatched across the gap.
- Automation or a well-defined process for generating a consistent unstructured mesh family would be welcome.
- Some participants felt they would get more out of the workshop by performing parameterized mesh study (prev. slide) than by spending that time generating mesh families.

Fitness for Purpose

The discussion focused on the following points:

- To some extent mesh fitness is also solver dependent, apart from purpose-dependent; solvers (should) provide own metrics
- Depending on purpose, several methods of evaluating mesh fitness can be employed
    - “Standard” grid quality metrics such as visual inspection, maximum dihedral angle and cell volume ratio
    - Solution (solver) sensitivity to mesh perturbation
    - Minimize entropy production
    - Propagation of solution (ex. MMS)
    - Adaptive mesh indicators
- When is a better mesh worth it?

Mesh Portability

The discussion focused on the following points:

- Some mesh formats offer validators (CGNS), which can generate useful and meaningful error output, but continuing version support is needed; most users are not aware/don't know how to use it.
- Mesh format should capture user intent in a useful way
- Universal (widely-used), library-based formats should be further explored ((HDF5, ADIOS, etc)
- Will things be done “on the fly” by 2030? Legacy formatting support and versioning will still be required.
- There should be agreement on how the data is stored and what the content is
    - Family
    - Description (that propagates through the whole transfer process)
    - Maintain naming conventions
- Need for future proofing format to accommodate 4D and higher order meshes
- Large meshes for HPC applications exhibit partitioning and transfer difficulties. Should we move toward storing a recipe for rebuilding the mesh (library-based, how general can this be)?