Top 5 misunderstandings on (good) mesh

Posted on October 27, 2011 by shengwei on
http://caewatch.com/top-5-misunderstandings-on-good-mesh/

Although there are quite a few mesh-free (mesh-less) FEA and CFD codes, meshing is still one of the most important tasks for most CAE users. The importance of generating high quality mesh can never be overemphasized. meshing-world-map-tri.

But how to define a high quality, or more preciously good, mesh? Reading the output of mesh quality report in your meshing software is only the essential step; you need make judgements whether the mesh is good enough for your physical problem.

Unfortunately, there are a lot of misunderstanding of good mesh. Nowadays, it is hard to find meshing course in engineering departments. The numerical algorithm in most engineering schools is optional. So, it is not surprising, the new generation CAE users lack some fundamental knowledge on how meshing works in a CAE system.

Here are the top 5 misunderstandings on good mesh.

#1. Good mesh must follow CAD model well.

More and more CAE users are designers, as explained in the previous post. They are generally well trained in CAD, and they tend to bring all details to analysis. They believe more details means more close to the reality.

This is not true, most of the time. Good mesh need resolve physics, not follow the CAD model.

The purpose of CAE simulation is to get physical quantities: stress, strain, displacement, velocity, pressure. CAD model is simply an abstraction of physical objects. Lots of details are not relevant to your analysis, or has marginal impact on your calculation. Therefore, knowing the physics in your system is essential. Good mesh should simplify CAD model and placing nodes based on physics.

This implies you can generate good mesh only if you know the physics in your system.

#2. Good mesh is always good.

So many times, we can see some CAE users work hard to get high quality mesh by changing meshing size, decomposing geometry and defeaturing. They carefully check the mesh quality output in the meshing software.

This is necessary. But do not over-do it, because good mesh is not always good. It depends on the physic problem to be simulated.

For example, you generate a very good mesh that captures the flow around an airfoil perfectly and calculates all forces accurately. Now if you change the flow attack angle from 0 to 45 degree, is your mesh still good? Very likely not.

A good mesh is always associated with the physics. When you change your boundary conditions, or change your load conditions, or change analysis types, or change flow models, good mesh (with the identical geometry) may turns to be bad mesh.

#3. Hexahedron is always better than tetrahedron.

Most old text books will tell you hexa (quad) mesh is better than tetra (tri) and show you how larger the numerical errors may be introduced by tetra (tri) mesh. Sometimes, this is true, especially 15 or 20 years ago.

Historically, people prefer hex mesh due to: 1). at that time, only structured mesh can be used for most CFD solvers ; 2). less cell (element) count (so, a lot of saving in RAM and CPU time); 3). unstructured solver was not matured.

The solver technology developments in most commercial FEA and CFD codes in last decade have led to the similar results for hex and tetra mesh for most problems . Of course, tetra mesh normally need more computing resources during solving stage. But his can be easily offset by the time saved in mesh generation. The accuracy advantage of hex mesh is no longer existing anymore, for most engineering problems.

For some special applications, e.g., wind turbine, pump, or airplane, hex mesh is still preferred, because of 1). industry convention; 2). well-understood physics (most users know how to align the mesh); 3). special tools to generated hex mesh for such geometries.

However, for most FEA and CFD users, if the geometry is slightly complicated, it is just a waste to spend time on hex meshing. Your results will not be better, most of time, if not always. The (solver) computing time saved with hex mesh is marginal compared with time wasted in mesh generation.

#4. Good mesh can only not be generated by automatic meshing.

When the software vendor need justify that his meshing software is in the high-end (of course price is usually in the high end as well), he will tell you the software allows you to all sorts of manual controls. The hidden information is only manual controls can generate better mesh.

Yes, to salesman, good mesh need manual control. But to engineer, you need understand this is misleading. A good meshing software should have the intelligence to analyze the underlying geometry: calculating curvatures, finding gaps; finding small features; finding hard edges; finding sharp angles, applying sensible default settings. . . .

This is what automatic meshing should do. For most users, software can gather more info, and more accurately, on the underlying geometry. So, the software should be able to apply better settings to get better quality mesh compared with a lot of users users.

Of course, for advanced users who use the software everyday and work on similar geometries for years, the story may be different. Such users also know the physics very well, and meshing software has no way to understand physic problems to be solved.But the percentage of such power users is decreasing day by day.

Anyway, for mesh quality, a good automatic meshing software should perform far better then most non-experienced users unless you have a lousy meshing software. Manual control is for experienced users who know the physics very well.

#5. Good mesh must have a large cell/node count.

Because HPC resource is easy to access, even a FYP student may try to solve 10-20 million cells CFD problems. In the eyes of most CAE users, large cell count means high fidelity and therefore all physics can be resolved.

This is not true, because some physics must be modeled. For example, in CFD, if you are going to use standard wall function, then any nodes in the viscous layer simply makes your wall function invalid. You are not only wasting computing resources, but also producing unphysical results.Even for LES, excessive fine mesh may lead to larger errors and unphysical solutions.

Fine mesh does not mean good mesh.The purpose of meshing is to get solutions in discrete locations. Good mesh is the mesh that serves your project objectives. So, as long as your results are: 1) physical; and 2) accurate enough for your project, your mesh is sufficiently good.

Another example of this misunderstanding is that most users always use full 3D model. In their eyes, 3D full model is "realistic". However, if the problem is symmetry, using portion of your geometry (therefore less nodes/elements) will produce better results because the symmetric condition is exactly enforced. If the problem is axis symmetric, 2D results is far more accurate than most 3D results. Anyway, most new generation CAE users does not have the time to fully understand the physics of the system to be simulated. It is hard for them to do any simplifications. So, it is not surprising, the CAE problem is getting bigger and bigger, unnecessarily.

Currently, CAE still relies on meshing. Good mesh should:

1. **be able to resolve the physics to be studied;**
2. **have reasonable quality so that the solver will not fail;**
3. **simplify the geometry based on physics;**
4. **be problem specific;**
5. **meet your project requirements**