Automatic mesh sizing specification of complex three dimensional domains using an octree structure

M. Turner[a,*], D. Moxey[a], J. Peiró[a]

[a]Department of Aeronautics, Imperial College London, London SW7 2AZ, UK

Abstract

A system for automatically specifying a distribution of mesh sizing throughout three dimensional complex domains is presented, which aims to reduce the level of user input required to generate a mesh. The primary motivation for the creation of this system is for the production of suitable linear meshes that are sufficiently coarse for high-order mesh generation purposes. Resolution is automatically increased in regions of high curvature, with the system only requiring three parameters from the user to successfully generate the sizing distribution. This level of automation is achieved through the construction of an octree description of the domain, which targets the curvature of the surfaces and guides the generation of the mesh. After the construction of the octree, an ideal mesh spacing specification is calculated for each octant, based on a relation to the radii of curvature of the domain surfaces and mesh gradation criteria. The system is capable of accurately estimating the number of elements that will be produced prior to the generation process, so that the meshing parameters can be altered to coarsen the mesh before effort is wasted generating the actual mesh.

Keywords: Octree; automatic mesh generation; background mesh; three dimensional; complex domain

1. Introduction

One of the major bottlenecks in the generation of curvilinear meshes that are suitable for high-order finite element computations, is the creation of coarse linear grids. Without a coarse initial mesh, that serves as a starting point for the process, the goal of automatic high-order generation will remain elusive. At present, the generation of coarse linear meshes is a balancing act, requiring adjustments to the mesh spacing definition to achieve a mixture of coarseness, geometric accuracy and suitable mesh gradation. This procedure therefore involves a large number of iterations of the mesh, making it time-consuming. If this process is to be automated, we must remove as much user interaction with the initial linear mesh as possible. The purpose of this paper is to address this challenge, by presenting a system for automatically obtaining a distribution of mesh sizing $\delta(\mathbf{X})$, where δ is the mesh size at the point of coordinates $\mathbf{X}$, for complex three-dimensional domains, to be used alongside any unstructured linear mesh generator.

To achieve this, an octree representation of the domain is used. Octree and quadtree structures have long been used in mesh generation, both in the top-down construction of the mesh and in assisting with the meshing process. In a top-down construction, the domain is subdivided until the boundary of domain intersects tree, conforming to a predefined

---

*Corresponding author.
    E-mail address: m.turner14@imperial.ac.uk

mesh specification. The octants, or quads, are then cut and warped to conform to the domain boundary [1]. When tree structures are used to assist the meshing process, typically, a background grid is used to provide information on mesh spacing to the generation processes. The use of quadtree structures, in the parametric plane, to automatically generate surface meshes is well researched and documented [2].

The use of an automatic mesh specification driven by surface topology using an octree structure is not completely new [3]. However, this reference noted that the computational cost and controllability of the method was poor, due to the use of the octree background grid, to the extent that the authors created a new system for automatic mesh definition which did not require a background grid [4]. In the context of generating coarse linear grids for high-order meshing, these conclusions are somewhat different. Indeed, we propose an efficient procedure. This is accomplished by actively keeping the algorithms involved in the construction of the octree as simple as possible.

Our system first generates an initial octree of the domain, which is constructed with little need for geometric conformity. That is, octants are not trimmed so that the edges of boundary octants lie on the geometric surface, but instead so that each boundary encompassing octant describes a small region of the surface that has nearly uniform curvature. As such, the computational demands of the octree are minimised. The curvature of the domain surface is then related to a ideal mesh spacing specification, which is propagated from the surface octants to the rest of the domain using gradation criteria. Finally, we note that the ability to accurately predict element counts before generation represents a crucial feature for high-order meshing, as this allows the user to easily coarsen the mesh. We show that in our approach, before any mesh generation, our system can predict the number of elements in the final mesh to an accuracy within 20%.

2. Mesh spacing distribution

As we relate the mesh sizing specification to the radius of curvature of the boundary of the domain, the accurate determination of the surface curvature is vital. The curvature at a point on the surface is calculated using the principal curvatures, which is explicitly defined in [5]. The radius of curvature for a given point, R, is taken to be $R = \min\left\{\frac{1}{k_1}, \frac{1}{k_2}\right\}$, where k_1 and k_2 are the two principle curvatures at the point obtained from the CAD representation. This leads to an isotropic mesh specification, since the directionality of the curvature is not considered.

To obtain a specification for the mesh sizing, we relate the radius of curvature R of the surface to a mesh size parameter δ through

$$\delta = 2R\sqrt{\varepsilon(2-\varepsilon)}, \tag{1}$$

where ε is a user-defined parameter that controls the sensitivity of the system to curvature. Heuristically, decreasing ε will increase the number of elements specified for a given degree of curvature. This definition was successfully first introduced to enable the use of curvature-aware high-order meshing, shown in [5]. Since R has the range $0 < R \le \infty$, where $R = \infty$ denotes a flat surface, 'sensible' bounds are placed on range of δ so that $\delta_{\min} \le \delta \le \delta_{\max}$. The parameter $\delta_{\min}$ allows the user to control and limit the density of elements in regions of high curvature. $\delta_{\max}$ places a limit on the size of the elements to be found in the final mesh. The three constants $\delta_{\min}$, $\delta_{\max}$ and ε represent the only user-specified parameters required.

To specify a spatial distribution of mesh sizing, the domain is described using an octree structure. This is a tree structure where each parent octant possesses exactly 8 child octants. Each child octant may be further subdivided, leading to a hierarchical graph-based data structure. The octree aligns well with the description of three-dimensional domains, since the subdivision of each octant into eight children equates directly to the division of each coordinate direction in two.

In the final octree, each location in the domain is contained within exactly one leaf octant (i.e. an octant that has no child octants). The octant which contains a given location can be identified quickly by cascading through the octree from the master octant. Although this is very fast for querying mesh spacing in the generation stage, this gain is far outweighed by the cost of constructing the octree. Here, every effort is made to mitigate the computational cost of constructing the octree by always subdividing equally and performing no post-processing, or warping, of the octants for the purpose of boundary conformity.

3. Automatic mesh spacing algorithm

This section outlines each stage of the process for obtaining a mesh spacing specification within a domain.

Construction of the initial octree. At the start of the process, the representative radius of curvature at sampled locations on the surface is used to construct the octree. To generate this octree a master octant, which encompasses the entire domain, is subdivided recursively based on the condition that, if the octant contains any of the domain surface and the ratio of the maximum and minimum radii of curvature found is greater than 1.1, the octant will divide. This value strongly influences the number of octants that will be created in the octree and could be relaxed to decrease computational cost at the expense of fidelity. We recommend a value of 1.1 because this ensures optimal quality meshes for the geometry. Additionally an octant will not subdivide if any of the dimensions of the resulting divided octants are less than $\delta_{\min}$. This places sensible bounds on the depth of the octree to be constructed.

Smoothing of the octree. Once the initial octree is constructed, the leaf octants are further subdivided, so that all neighbours of any given octant are either half or double the spatial dimensions of the octant. This ensures that when the mesh spacing specification is propagated through the domain, the distance over which the propagation is made is representative of the octant sizes. This facilitates a smooth gradation of mesh sizing specification throughout the domain.

Surface specification smoothing. Prior to propagation into the interior of the domain, the mesh specification of the surface octants is smoothed to ensure good mesh gradation. If the condition

$$\frac{|\delta_i - \delta_j|}{l_{ij}} \leq 0.1 \tag{2}$$

is not satisfied, between an octant and any of its neighbours, the neighbouring octant's specification of δ is adjusted to conform to the criteria. In this equation, δ_i and δ_j denote the mesh sizing specification of the octant and its neighbour and l_{ij} is the distance between them.

Domain propagation. Finally, the information of mesh sizing δ is propagated to the interior of the domain. This is done by considering all octants that do not have a mesh specification defined. We then inspect each neighbouring octant. If any of these neighbours have a spacing specification, then δ for the unspecified octant is defined to conform to

$$\frac{|\delta_i - \delta_j|}{l_{ij}} \leq 0.3 \tag{3}$$

If more than one of the neighbours has a spacing specification, the minimum δ_i is used for the current octant. This process is repeated until all octants have a spacing specification. The δ specifications are then smoothed using the same criteria. The mesh gradation parameters, 0.1 and 0.3, were found through trial and error and found to give optimal meshes, however, these parameters could become user input parameters.

Element count estimation. Prior to any mesh generation, an estimation of the number of elements to be created in the mesh is made. The estimated element count is evaluated as the sum of the number of tetrahedra that would fit into octants that are within the domain i.e.

$$N_{predicted} = \sum_i N_i = \sum_i \frac{V_o^i}{V_t^i} \tag{4}$$

where i is a leaf octant that is interior to the domain, V_o^i is the volume of the octant and V_t^i is the volume of a tetrahedra of dimension δ as specified for the octant.

4. Results

To perform the mesh generation, the octree structure is first passed to a in-house surface mesh generation code, and then to TetGen [6]. We demonstrate this procedure on an aircraft geometry. The accuracy of the predicted element count is also examined.

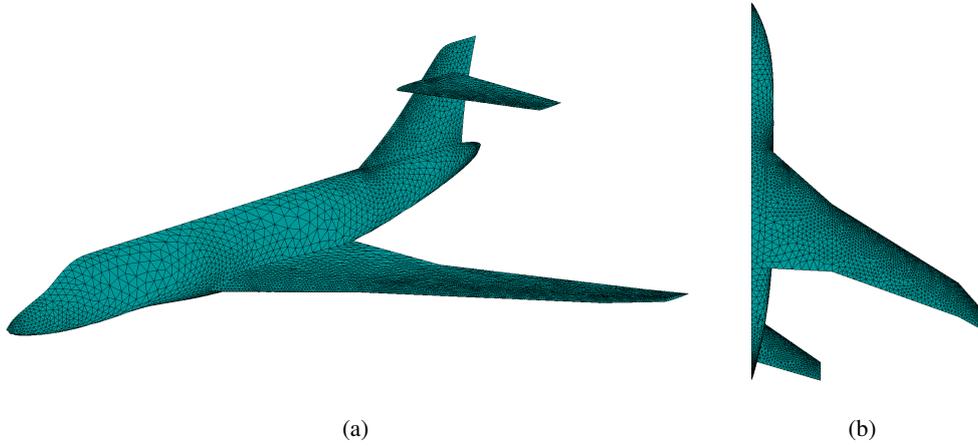(a)   (b)

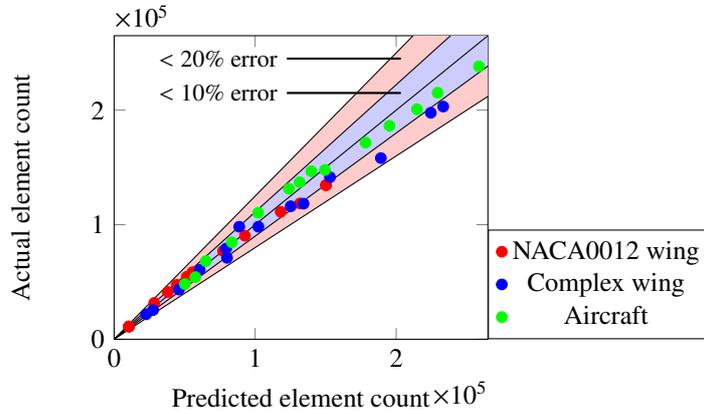Figure 1: Surface mesh of an aircraft geometry.

Figure 2: Predicted vs actual element count for various meshes using the automatic octree system

Figure 1 shows a mesh obtained using the automatic specification process on an aircraft geometry. There is a wide range of curvature across the surface, with many regions of high curvature such as the nose, tail cone and the leading edges of the three wing profiles. The figure highlights that the system has correctly identified these areas of curvature and increased resolution accordingly. Further to this the figure shows that the mesh has been defined to smoothly transition between element sizes.

For the aircraft, as well as additional cases of a NACA0012 extruded wing and a more complex anti-symmetric swept wing, which we omit for brevity, we show the *a priori* predicted element count against the amount that are produced. For each test case, the three user-defined parameters δ_{min}, δ_{max} and ε were varied in turn and at a wide range of values to produce meshes of various coarseness. Figure 2 shows that for all the domains, the error in element estimation is always below 20% and, in most cases, is well below 10%. This represents a significant capability to accurately estimate the potential element count.

5. Extension to high-order

As mentioned in the introduction, the primary purpose of this system is to automatically create adequate linear meshes as a starting point for high-order mesh generation. For these purposes, the mesh needs to be much coarser than in a standard mesh. However, increased resolution in high geometric curvature regions is still required to ensure that the high-order description of the surface remains accurate. To demonstrate suitability for this purpose, a much
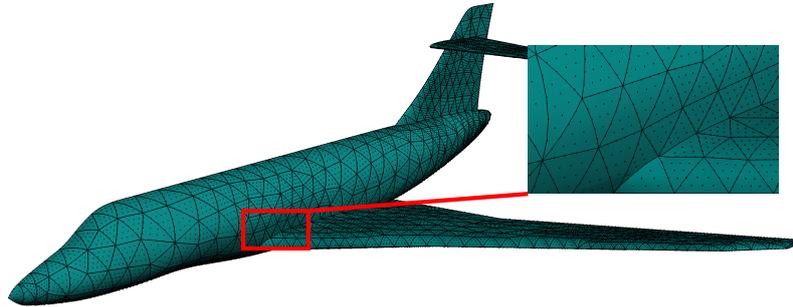
4

Figure 3: High-order surface mesh of the aircraft geometry.

coarser mesh of the aircraft geometry was made. This mesh was of around 35,000 tetrahedra, compared to the earlier example of around 260,000 tetrahedra in Figure 1.

We then process this mesh by placing additional high-order nodes on the surface, the distribution of which is optimised to the CAD geometry through the approach described in [5]. The resulting sixth-order surface mesh is shown in Figure 3. Even with a far coarser mesh, it is clear that our system places increased resolution in the areas of high curvature.

6. Conclusions

We have presented a procedure for the automatic definition of mesh sizing distribution within complex three dimensional domains. This is achieved using an octree representation of the domain, a relation between surface curvature and mesh size, and a method for propagating this information into the domain. We have shown that the system accurately and effectively targets regions of high curvature within the domain and increases the resolution, ensuring a good description of the domain. Further to this it has also been shown that the growth in element sizes away from the domain boundaries is smooth. In addition the system's ability to predict the number of elements that will be produced in the final mesh is shown to be of significant enough accuracy to be a very useful tool for users to control element count and mesh coarseness. Furthermore, the system presented here represents a precursor stage to the generation of high-order meshes and has been shown to be suitable for this task.

7. Acknowledgements

References

[1] A. Cunha, S. Canann, S. Saigal, Automatic boundary sizing for 2D and 3D meshes, in: AMD Trends in unstrucutred mesh generation, AMSE, 1997, pp. 65–72.
[2] Z. Xiao, J. Chen, Y. Zheng, L. Zeng, J. Zheng, Automatic Unstructured Element-sizing Specification Algorithm for Surface Mesh Generation, Procedia Engineering 82 (2014) 240–252.
[3] L. Kania, S. Pirzadeh, A Geometrically-Derived Background Function for Automated Unstructured Mesh Generation, AIAA Computational Fluid Dynamics Conference 17 (2005) 18.
[4] S. Z. Pirzadeh, Advanced Unstructured Grid Generation for Complex Aerodynamic Applications, AIAA Journal 48 (2010) 904–915.
[5] S. Sherwin, J. Peiró, Mesh Generation in curvilinear domains using high-order elements, International Journal for Numerical Methods in Engineering 53 (2001) 207–223.
[6] H. Si, TetGen, a Delaunay-Based Quality Tetrahedral Mesh Generator, ACM Transactions on Mathematical Software 41 (2015) 11:1–11:36.