

High-order Curved Prismatic Mesh Generation Using Minimum Distance Fields

Jay Sitaraman, Beatrice Roget, Vinod Lakshminarayan and Michael Brazell

Abstract A new method for directly generating high-order curved prismatic boundary layer meshes is developed. The algorithm utilizes the iso-surface of minimum distance field for both point-placement and smoothing purposes. Minimum distance fields are computed such that they are consistent with the high-order boundary representation. High-order curved prismatic mesh results are demonstrated for the NASA common research model.

1 Introduction

Unstructured meshes near wall boundaries typically utilize anisotropic prismatic meshes to capture the viscous boundary layer. High-order flow solution processes, especially when using finite-element type discretization with internal degrees of freedom within each element, require these anisotropic meshes to be curved and consistent with the boundary representation. Currently, the method for generating body conforming curved meshes involve two steps (1) Generation of a coarse linear mesh using well-established unstructured meshing practices based on advancing fronts [1, 2, 3, 4] and (2) propagation of the surface curvature through elastic deformations [5, 6, 7], attenuation of the surface deformation through algebraic techniques [8] or splitting macro-elements using iso-parametric approach [9]. A com-

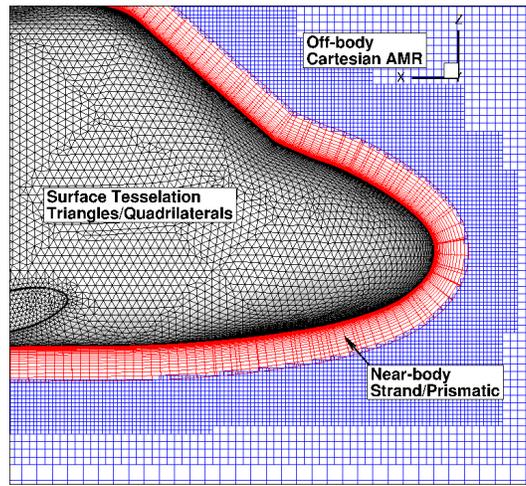
Jay Sitaraman
PGA LLC at Sunnyvale CA, e-mail: jsitaraman@gmail.com

Beatrice Roget
STC Corp, Moffet Field CA, e-mail: beatrice.f.roget.ctr@mail.mil

Vinod Lakshminarayan
STC Corp, Moffet Field CA, e-mail: vinod.k.lakshminarayan.ctr@mail.mil

Michael Brazell
University of Wyoming, e-mail: mbrazell@uwyo.edu

mon challenge mentioned in all of these works involves curving of the anisotropic boundary layer cells, that will intersect the wall boundary in the absence of significant deformation consistent with the wall-curvature. Most of these works also involved creation of mixed element unstructured meshes that include tetrahedra. In contrast to these efforts, herein we propose a different approach that involves direct creation of the curved boundary layer cells rather than an *a posteriori* process that involves generation of a linear mesh. Further, we propose to use a dual-mesh approach where the prismatic boundary layer mesh transitions to a Cartesian AMR system a short distance from the wall as shown in Figure 1. Therefore, the challenge here is to create a curved semi-structured prismatic mesh often referred to as a “strand mesh” a short distance from the wall. To achieve this goal, we extend the methodology presented in our previous work [10] for linear prismatic mesh generation to high-order curved meshes. Our method involves using the iso-surface of the minimum distance field as a guide for point placement and the subsequent smoothing process.



(a) Strand-Cartesian grid system

Fig. 1 Strand/Cartesian simulation framework. Body conforming strand (prismatic with no topology change in layers) grid in red and Cartesian grid in blue. This work aims at creating a high-order curved body conforming strand grid.

2 Methodology

Meshing methodology involves three major steps (1) Placing strands (line connecting a wall surface node and corresponding outer surface node) such that the result-

ing mesh is nearly valid and (2) Constrained smoothing of the outer surface nodes to improve the quality of the mesh and (3) Sub-division of the macro strand cell formed by the high-order surface triangle and its counterpart on the outer boundary according to a prescribed anisotropic distribution normal to the surface.

2.1 Initial strand placement

Initial strand placement involves finding the initial location of the end points of each strand. Two concepts are used for this purpose, the best visibility direction and the iso-surface of the distance field.

Best visibility direction:

In most prismatic mesh generation approaches (advancing front or direct placement), the local normal direction is used for the extrusion of the nodes. The simplest way to compute a local normal direction at each node is by averaging the normals of all the facets that are associated with this node. The simple averaged normal, both unweighted and area weighted, often leads to issues because of the bias in averaging caused by the difference in the number of geometric regions and topological regions that enclose the given node. Aubry et al. [11] proposed a much more robust approach, using the concept of the most normal normal (MNN), i.e. the direction that maximizes the minimum angle between the surrounding faces as the optimal direction of choice. For high-order surfaces, the vertex normals are computed by applying the MNN algorithm to set of normals composed of the exact cell normals of the curved triangles that bound the vertex.

Closest Vertex on the Iso-Surface (CLOVIS) algorithm:

The iso-surface of minimum distance at I_L , is defined as the locus of points that are at a given fixed distance L from the discrete surface tessellation \mathcal{S} . Examples of minimum distance iso-surfaces are shown in Figure 2(a). The goal of the initial point placement algorithm is to compute point positions on I_L corresponding to each surface node on \mathcal{S} , such that the number of invalid elements is minimized. As shown in Figure 2(b), the closest vertex to each surface node on I_L is a good candidate for point placement, because it automatically creates a desirable bending of strands in regions of concavity. The number of strands near the concave ridges/corners that bend is directly correlated to the iso-surface distance L , i.e. larger iso-surface distances would cause more strands in a larger region to bend away from the original best visibility direction. However, the point distribution on the outer envelope surface is not ideal for generating the next mesh level, and a smoothing process needs to be applied to improve the quality of the envelope mesh. Given a tessellated surface, \mathcal{S} , the isosurface of the distance field at L is defined as:

$$I_L = \{P \in R^3 \mid \text{MinDist}(P, \mathcal{S}) = L\}, \text{ where } \text{MinDist}(P, \mathcal{S}) = \min_{A \in \mathcal{S}} \|\vec{AP}\| \quad (1)$$

For any surface vertex A , the closest point P on the isosurface at a distance L is any point that satisfies:

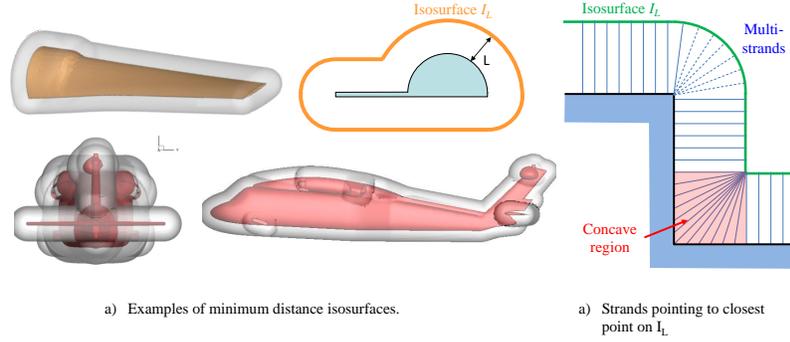


Fig. 2 Choosing a strand vector towards the closest point on iso-surface of distance automatically results in desirable strand distribution in concave regions.

$$\begin{cases} \text{MinDist}(P, \mathcal{S}) = L \\ \|\vec{AP}\| = \min_{\text{MinDist}(M, \mathcal{S})=L} \|\vec{AM}\| \end{cases} \quad (2)$$

The optimization process entails sliding the end point of the vector (P) on the iso-surface of minimum distance and locating it such that the segment AP has the shortest length. The fact that the isosurface of distance field is only known implicitly by its mathematical description makes the solution of Eq 2 challenging. A discrete solution to the continuous optimization problem can be obtained by constructing an approximate tessellated isosurface using a marching-cube method. However, this approach was found to lack robustness and computational efficiency. Instead, an algorithm was designed to efficiently compute, for each surface vertex, the closest point on the actual analytical description of I_L shown in Eq 1. This method is referred to as the CLOVIS algorithm (Details in Ref [10]). The method makes intensive use of an efficient routine to compute $\text{MinDist}(P, \mathcal{S})$, which returns the shortest distance to surface \mathcal{S} from a point P . We use a divide and conquer approach that uses an Alternating Digital Tree (ADT) for the culling process. The primary extension required in this context is the ability to compute minimum distance and point of minimum distance to a high-order curved triangle. We accomplish this by utilizing a bound constrained minimization process that utilizes the natural coordinates (s, t) within a curved triangle as design variables. To accelerate the process, the curved triangle is first split into a set of linear sub-elements and the closest linear sub-element is located through the analytical distance formula available for linear triangles. The natural coordinates of the linear triangle is then used as the initial guess for the constrained minimization process. Note that the CLOVIS algorithm is highly parallel and threadable and can be independently applied to any point on the surface and hence naturally extends to usage in a high-order context where multiple points associated with an element will need to find their counterpart on the outer boundary of the prismatic mesh.

2.2 Mesh smoothing, constrained to the iso-surface, using spring analogy

To improve the mesh quality, a smoothing algorithm is applied, which is loosely based on a linear spring analogy with constraints enforced so that strand end points remain on the iso-surface of distance. For high-order elements, a tessellation composed of the linear sub-elements that connect all the control points is utilized such that the smoothing is applied uniformly to all of them. In this method, each edge on

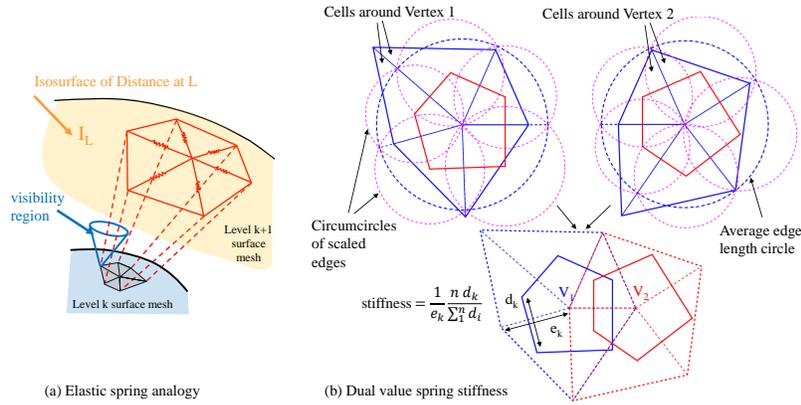


Fig. 3 Envelope mesh smoothing using elastic spring analogy.

the envelope surface is treated similar to a linear spring, as illustrated in Figure 3(a). The component of elastic force normal to the iso-surface of distance is removed to restrict the movement of the nodes towards the surface (mesh collapse) during the smoothing process. After the elastic force vectors are computed, the new position of the strand end nodes are computed using a carefully designed time-marching scheme, with constraints applied to ensure all points remain on or above the iso-surface of distance and within their respective region of visibility. The constraint is enforced as follows: For points in a convex region, point are moved along the local normal to the isosurface (vector joining the point to its closest point on the surface) until it is located on the isosurface. For points in a concave region, this operation is performed only if the point is located below the isosurface and the distance to the surface is decreasing. Otherwise, the point is allowed to move above the iso-surface, but the strand length is limited so that the distance to the surface does not exceed a heuristic limit. The smoothing iterations are terminated when a valid mesh of sufficient quality is achieved.

3 Results

High-order boundary layer meshing strategy is applied to NASA common research model surface mesh available as part of the 5th high-order workshop [12]. A coarse quadratic surface mesh composed of 8304 6-node triangles is used as the wall boundary. Figure 4 shows the overall mesh system, with the envelope of the curved prismatic mesh super-imposed over the curved wall boundary. Sufficient smoothness is obtained on all parts of the domain with the outer envelope following the minimum distance iso-surface of the curved wall boundary. Figure 5 shows details of the curved prismatic layers near the moderately convex wing leading edge and highly concave wing fuselage intersection. Again, sufficient smoothness of both the anisotropic layers as well as the prismatic outer layer can be observed.

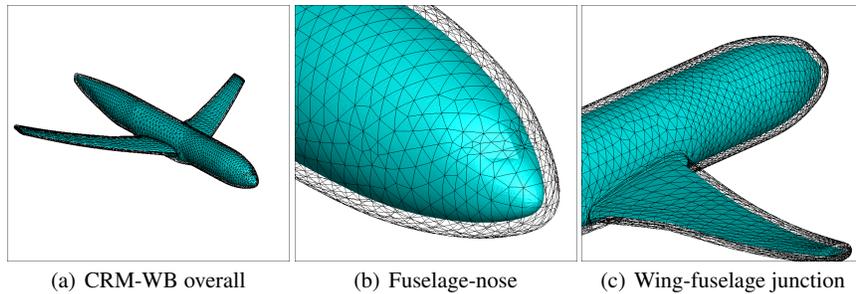


Fig. 4 Wall boundary and prismatic envelope of the high-order strand mesh.

4 Conclusions

The methodology for generating linear prismatic meshes using minimum distance fields is extended to generation of high-order curved prismatic meshes. The primary extension is the inclusion of a distance function calculator for high-order curved surfaces. Results are favorable, indicating ability for generation of smooth curved meshes for a widely available and moderately complex wing-body geometry. Next step involves flow computation on these mesh systems using an overset framework that uses a high-order near-body solver and high-order Cartesian AMR solver.

Acknowledgements Material presented in this paper is a product of the CREATE-AV Element of the Computational Research and Engineering for Acquisition Tools and Environments (CREATE) Program sponsored by the U.S. Department of Defense HPC Modernization Program Office.

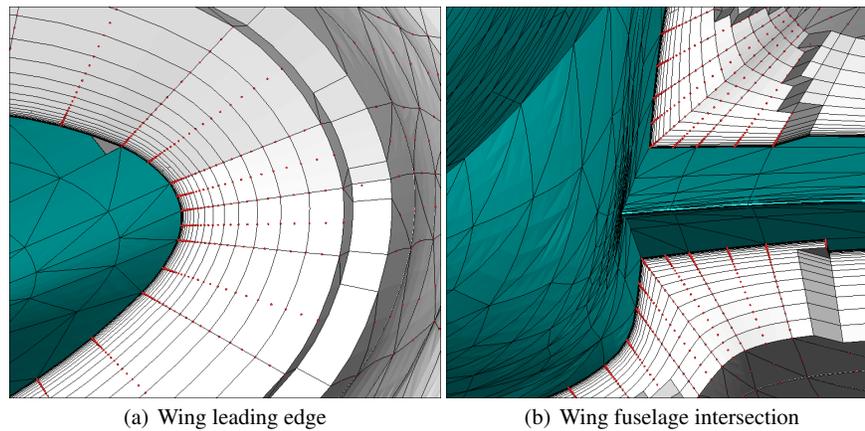


Fig. 5 Curved boundary layer mesh details at wing leading edge (convex curvature) and wing-fuselage intersection (concave curvature). Meshes are shown by masking cells that have coordinates above planes $X > 184.1\text{mm}$ and $Y > 1380\text{mm}$ respectively. Plots have staggered boundaries because mesh is not coordinate-aligned. Red points represent degrees of freedom within an element.

References

1. Shahyar Pirzadeh. Unstructured viscous grid generation by the advancing-layers method. *AIAA journal*, 32(8):1735–1737, 1994.
2. Yannis Kallinderis, Aly Khawaja, and Harlan McMorris. Hybrid prismatic/tetrahedral grid generation for viscous flows around complex geometries. *AIAA journal*, 34(2):291–298, 1996.
3. David Marcum and J Gaither. Mixed element type unstructured grid generation for viscous flow applications, 1999.
4. John P Steinbrenner. Construction of prism and hex layers from anisotropic tetrahedra. In *22nd AIAA Computational Fluid Dynamics Conference*, page 2296, 2015.
5. Michael Turner, Joaquim Peiró, and David Moxey. A variational framework for high-order mesh generation. *Procedia engineering*, 163:340–352, 2016.
6. Meire Fortunato and Per-Olof Persson. High-order unstructured curved mesh generation using the winslow equations. *Journal of Computational Physics*, 307:1–14, 2016.
7. Laslo T Diosady and Scott M Murman. A linear-elasticity solver for higher-order space-time mesh deformation. In *2018 AIAA Aerospace Sciences Meeting*, page 0919, 2018.
8. Steve L Karman, J T Erwin, Ryan S Glasby, and Douglas Stefanski. High-order mesh curving using wcn mesh optimization. In *46th AIAA Fluid Dynamics Conference*, page 3178, 2016.
9. Julian Marcon, Michael Turner, Joaquim Peiro, David Moxey, Claire Pollard, Henry Bucklow, and Mark Gammon. High-order curvilinear hybrid mesh generation for cfd simulations. In *2018 AIAA Aerospace Sciences Meeting*, page 1403, 2018.
10. Beatrice Roget, Jay Sitaraman, Vinod Lakshminarayanan, and Andrew Wissink. Prismatic mesh generation using minimum distance fields. In *Tenth International Conference on Computational Fluid Dynamics (ICCFD10)*, pages ICCFD10–219, 2018.
11. Romain Aubry and Rainald Löhner. On the ”most normal”normal. *International Journal for Numerical Methods in Biomedical Engineering*, 24(12):1641–1652, 2008.
12. 5th international workshop on high order cfd methods. In *AIAA Scitech 2018, Jan 7-11*. AIAA, 2018.