Improved Hexahedral Meshing on Biological Models

Sara Richards
Department of Civil and Environmental Engineering
Brigham Young University
Email: slj42@email.byu.edu

Steven Benzley
Department of Civil and Environmental Engineering
Brigham Young University
E-mail: seb@byu.edu

Jason Shepherd
Sandia National Laboratories
Email: jfshep@sandia.gov

Mike Stephenson
M.B. Stephenson & Associates
Email: mbsteph@sandia.gov

***Abstract*— Certain applications of the finite element method require hexahedral meshes for the underlying discretization. A procedure, known as THexing, which is guaranteed to produce an all-hex mesh is to begin with a tetrahedral mesh and then subdivide each element into four hexahedra. This research presents a method for improving the THex approach, known as Diced THexing, or DTHexing. The DTHex approach is based on general coarsening tools. An initial triangle surface mesh is coarsened and smoothed iteratively until a coarse mesh of reasonable quality is obtained. The volume is then easily meshed using a tetrahedral scheme, then refined using 'h' type modifications. The goal of this method is to 1) improve the quality of elements in the finite element mesh and 2) decrease the number of overall nodes. The DTHex approach has been successful at improving models on biological meshes without increasing node count. This research was conducted using the CUBIT software.**

I. Introduction

Finite element analysis is a numerical approach for analyzing multivariable systems with piece-wise approximations. Although the first applications of the finite element method were limited, it is now widely incorporated in many disciplines such as aeronautical engineering, structural engineering, computational fluid dynamics, microelectronics, groundwater flow, aerodynamics, computational medicine, mechanical engineering, and electrical engineering. As computing power continuously increases, the type and complexity of problems that can be solved also increases.

Because finite element analysis is a numerical procedure, its success is closely tied to the accuracy of the discretization (i.e. the mesh). While it would be ideal to use extremely fine meshes in all cases, it is not computationally feasible to perform analyses on such refined models. Rather, the analyst must try to get the "best mesh possible" for the available computing resources. Some analysis codes have been written to work only with hexahedral elements, since it has been shown that hexahedral elements have some desirable qualities that allow them to perform better than their tetrahedral counterparts for a given number of degrees of freedom [1][2][5][6][22]. Since the early 1990's, research has produced several methods for producing hexahedral meshes [3][13][15][16][17][21]. However, none of these methods produces the "best possible" mesh in all situations, and significant user intervention is still required to produce acceptable meshes. To date, the only method that is guaranteed to generate all-hexahedral meshes on arbitrary geometries is known as THexing[20]. THexing is the process of splitting each element in a tetrahedral finite element mesh into four hexahedral finite elements as shown in Figures 1 and 2.

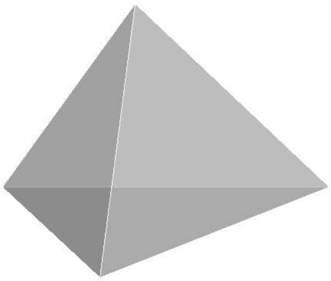


Fig. 1. A single tetrahedron

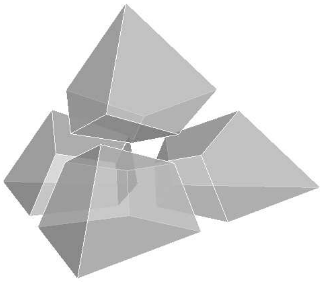


Fig. 2. THexing a single tetrahedron

One of the newer applications of the finite element method for three dimensional problems is in biological modeling. Initial biological meshes are often processed using scanning and imaging devices and software. These meshes typically have a single topological surface, a large number of features, regions of high curvature, holes, long narrow filaments, and a large number of nodes. In addition, there are often many inaccuracies and inconsistencies in the mesh that are generated in the data transfer process which require significant modification before meshing is possible. In almost all cases, the mesh is created as a triangle surface mesh. Tetrahedral meshing is then possible for the volume, using existing meshing technology[9][10]. Once a valid tetrahedral mesh is created, it can be THexed to form an all-hexahedral mesh. THexing is a simple process, but often generates more nodes that can be accomodated by the analysis code. Even though THexing is not ideal, it is the only fully automatic method currently available for meshing most biological models. This research explores the possibilities of improving the THexing approach. We present an algorithm known as DTHex that is based on general coarsening and refinement procedures. The goal of this research is to develop a general approach for meshing complex models that produces elements with better quality than THexed meshes while decreasing the number of nodes for analysis to thousands instead of millions.

II. Overview of Algorithm

DTHex, which stands for "Diced THexing" is an improved method for generating all-hexahedral mesh elements using an indirect hex-meshing approach. This research is not intended to be a solution to the all-hexahedral meshing problem for all classes of geometry, but proposes a solution to a specialized problem, namely hexahedral meshing for biological models. It is hoped that with time the algorithm will be able to expand to include additional classes of analysis problems. The focus is on improving the THex approach, since that method is the most commonly used for complex models, and has a great potential for impact in this domain.

The input to the algorithm is an initial surface triangulation. For biological models, this is the geometric faceted representation that is read in from the file. Geometric features are extracted based on angles between triangles [19]. The steps of the algorithm are given below.

- *Verify valid boundary conditions* – Every non-boundary edge must be connected to two triangles, and no edges can overlap.
- *Coarsen the mesh* – The algorithm used here is an edge-collapsing algorithm, based on surface curvature approximations. Edge swapping is then applied recursively to improve element quality and node valence.
- *Smooth the mesh* – There are two methods of smoothing that are employed in this research. The first is global smoothing, and the second is local smoothing.
- *Mesh the volume* – This may be performed using any tet meshing software[9][10].
- *Refine the mesh* – THex the mesh, and then further refine by splitting each hex into eight hexes.

The final result is a mesh with better quality and decreased node count. Examples will be given in Section VI.

III. Coarsening

Coarsening or decimation is the process of removing entities such as nodes, edges or faces from a triangulation. Coarsening is the most important step of the algorithm, since the surface mesh that is produced after coarsening determines the quality and topology of the final hexahedral mesh. The coarsening algorithm applied in this research is based on a curvature based sizing function that employs edge collapsing and edge swapping.

A. Edge Collapsing

Edge collapsing is the process of systematic vertex removal by edge deletion in a finite element mesh. The most difficult problems encountered in edge collapsing are determining a valid stopping criterion and preserving quality. Determining a valid stopping criterion involves defining what is "coarse enough" without sacrificing surface definition. The approach used in this thesis was to develop a stopping criteria that is individualized based on surface curvature at each point. The work of Frey and Borouchaki of Inria [4][7][8] was used as a basis for the edge collapsing algorithm presented here.

1) Curvature-Based Sizing Function: A curvature-based discrete sizing function is calculated for the surface using a method proposed by Frey and Borouchaki in [8]. No underlying knowledge of surface curvature is needed to calculate an approximation for curvature. One need only know the normal ν_P and tangent vector τ_P at a point on the surface (Figure 3). Frey and Borouchaki define the osculating circle of a point P on curve $\gamma(s)$. This approximation reduces to a second-order polynomial Taylor series approximation for small Δs. The point Q can be approximated by the formula:

$$Q = P + \tau_P \Delta s + \frac{\nu_P}{2\rho_P} \Delta s^2$$

where Q belongs to the plane defined by the vector $\vec{PQ}$ and the normal vector at P, ν_P. One may easily solve for an approximation to ρ_P. The algorithm is given below.

```
Step 1: Define starting point P;
Step 2: Get all attached edges to P;
Step 3: Set ρ_P = large number
        For i = 1 to number of edges
            Set Q_i = end node of edge i;
                Find PQ_i = P − Q_i;
                Find ν_P = surface normal at P;
                Set cos(θ) = PQ_i • ν_P/||PQ_i||
                Set ρ_I = ||PQ_i||/2 cos(θ)
                If ρ_I > ρ_P
                   ρ_P = ρ_I
                end If;
        end For;
Step 4: return ρ_P;
```

This algorithm loops through the edges surrounding a node, and finds the smallest osculating circle. The ideal edge length can then be determined by defining a coefficient α such that $h(P) = \alpha\rho_P$ where $h(P)$ is the Euclidean length at P (Figure 4). A variable ϵ is defined so that

$$\delta/\rho_P \leq \epsilon$$

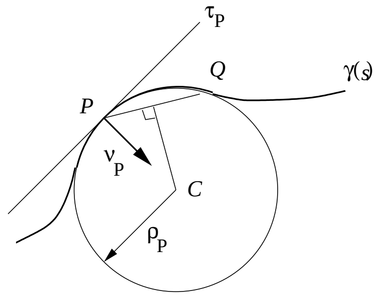


Fig. 3. Osculating circle approximation

where δ/ρ_P is the ratio of the distance between the line PQ and the osculating circle and the radius of curvature. This variable ϵ may be referred to as the coarsening factor. It can then be shown that

$$\alpha \leq 2\sqrt{\epsilon(2-\epsilon)}$$

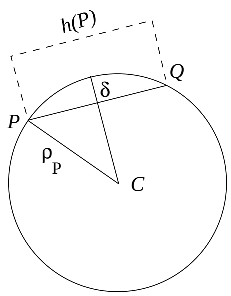


Fig. 4. Graphical interpretation of distance criterion

2) Edge Collapse Algorithm: The basic edge collapse algorithm shown below is a modification of [7].

- Let AB be an edge such that $l = L_n(AB)$ and let h_a, h_b be the sizing function parameters at A and B.
- Let $P_0(=A), P_1, \ldots, P_n$ be the vertices adjacent to B given in cyclic order counter-clockwise to the surface normal.
- Compute $l_i = L_n(AP_i), 2 \leq i \leq n-1$.
- If $\forall i, 2 \leq i \leq n-1, l_i > l$ and $l_i < \frac{1}{l}$, collapse B to A.

A discrete curvature-based sizing function is defined at each node $h(P)$. Given the size at each point, a normalized edge length may be determined using a linear interpolation scheme between points which yields the following formula for the normalized edge length.

$$l(AB) = \frac{d(AB)}{h(B)-h(A)}[Log\frac{h(B)}{h(A)}]$$

The goal of the algorithm is to coarsen such that all edge lengths on the surface have a normalized edge length equal to 1. An additional parameter is added to limit the size of any single edge to a user defined maximum, so that in the case of a perfectly flat surface, the user may determine a limit on edge length.

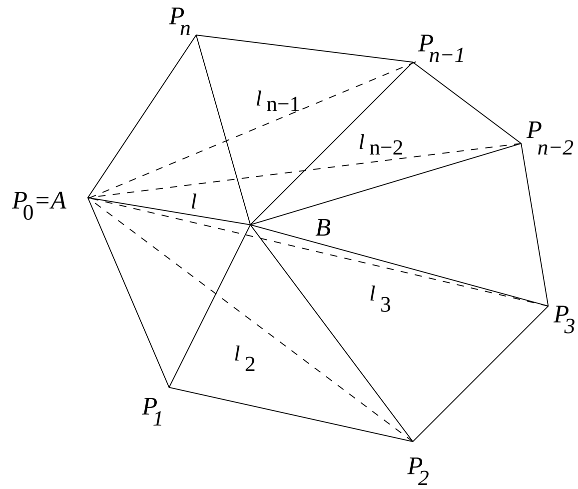


Fig. 5. Edge Collapsing Procedure

The following figures show the curvature based sizing function applied to a biological model of a tympanic membrane, a very thin membrane in the inner ear. All of the biological models in this paper are used with permission from researchers at the University of Utah (Jim Weiss and Andy Anderson). Previous coarsening methods do not adequately capture the sharp curvature near the edges while maintaining a coarse enough mesh on the rest of the model. The curvature-based coarsening scheme is able to adequately capture this curvature without sacrificing quality. The quality metric that is used in this paper is the shape metric as defined by Knupp (See reference [12]). The average shape quality, minimum, maximum and standard deviation are given for each example. A value of 1.0 indicates a perfectly shaped element (e.g. an equilateral triangle or a cube). A value of 0 indicates an inverted element. Table I gives quality metrics for elements coarsened at various levels of ϵ. Quality measures deteriorate as element count decreases, but these measures improve after smoothing and refinement are applied to the mesh. For ϵ = 0.01, the average quality is 0.9105, compared to 0.9397 for the original mesh. Average quality for ϵ = 0.1 and ϵ = 0.5 are 0.8963 and 0.8825 respectively. In all cases, the minimum quality increases from the original mesh at 0.1244, and the maximum quality stays near 1.0. Resultant meshes can be seen in Figures 6,7,8, and 9.

TABLE I

SHAPE QUALITY FOR MESHES USING CURVATURE BASED SIZING FUNCTIONS (FIGURES 6 TO 8.)

	Avg.	Std. Dev.	Min.	Max.
Original Mesh	0.9397	0.06240	0.1244	1.000
ϵ = 0.01	0.9105	0.07864	0.3288	1.000
ϵ = 0.1	0.8963	0.09119	0.1933	0.9991
ϵ = 0.5	0.8825	0.1076	0.2715	0.9998

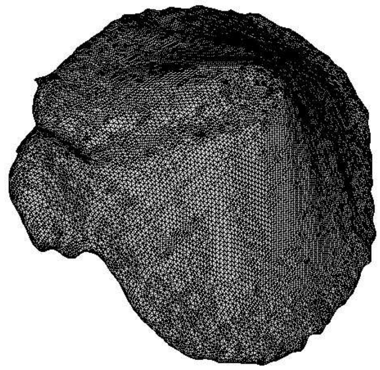


Fig. 6. Original mesh

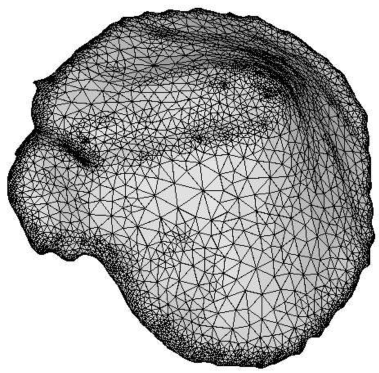


Fig. 7. Coarsening with ϵ = 0.01

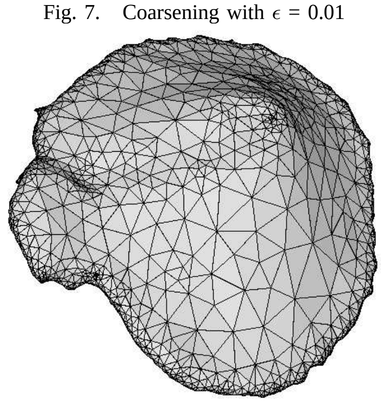


Fig. 8. Coarsening with ϵ = 0.1

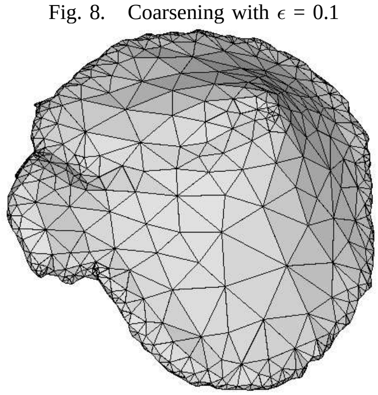


Fig. 9. Coarsening with ϵ = 0.5

IV. SMOOTHING

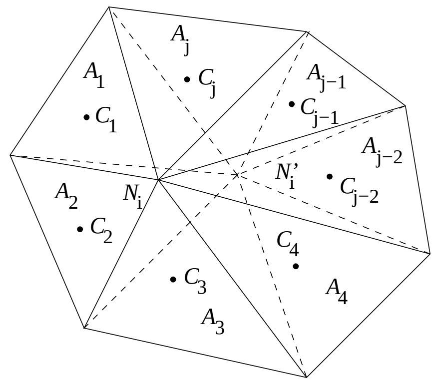
Coarsening cannot always guarantee well-shaped elements. Smoothing, or node moving, is a method to improve element quality. Two methods of smoothing-global smoothing and local smoothing-were used in this research. Global smoothing applies an iterative smoothing method over the entire surface after coarsening is complete. This method produces good quality elements, but takes significant processor time. Because of this drawback, an alternative smoothing procedure was introduced for this research known as local smoothing. In local smoothing, the nodes are smoothed during the coarsening process, after each edge collapse and edge swap. Each of these techniques will be discussed in greater detail in this section.

A. Global Smoothing

Global smoothing is a process of improving element quality over a surface or volume by iterative node movement. The term global refers to the way the smoothing algorithm is applied over an entire surface, node by node, and does not refer to the solution of a PDE for the surface. During each iteration, node movement should decrease, until it has reached some minimum tolerance. The centroid area pull method [11] is a smoothing method that is applied to nodes in a triangular surface mesh. The goal is to create elements of equal area over the surface. This approach was chosen initially in this research because of the balance between speed and quality. The algorithm given below corresponds to Figure 10.

Step 1: Start with a list of nodes on the surface;
Step 2: Loop through the list of nodes;
    For i = 1 to number of nodes;
        Set N_i = i^{th} node in the list;
        For j = 1 to number of triangles attached to N_i;
            Calculate total area $A = \sum A_j$;
        end For;
        For j = 1 to number of triangles attached to N_i;
            Calculate $w_j = \frac{A_j}{A}$;
            Find C_j = the center of A_j;
            Calculate $N\prime_i + = C_j w_j$;
        end For;
        Set $movement = N_i - N\prime_i$;
        If $movement >$ a small tolerance;
            Add neighbors of N_i to the list of nodes;
        end If;
Step 3: Stop when there are no more nodes in the list;

This method of smoothing produced good quality elements, but in practice took significant time because the entire mesh was smoothed after every coarsening iteration, instead of just smoothing the edges that changed. In addition, it did not respect the sizing functions, but instead tried to make all the mesh edges the same length. For these reasons, it was decided that a local smoothing method was needed to adequately capture mesh features.



$$w_j = \frac{A_j}{\Sigma A_j}$$

$$N_i = \Sigma\, C_j w_j$$

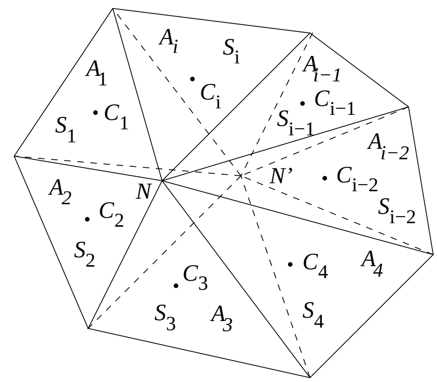
Fig. 10. Global smoothing

B. Local Smoothing

Local smoothing is similar to global smoothing in its methodology. However, local smoothing includes only a small number of nodes, instead of the entire surface. This is possible in coarsening because only one vertex is removed at a time. The benefits of this approach are that it significantly increases the speed of the algorithm and that it respects sizing constraints. Like the centroid area pull method, the localized smoothing technique that was employed in DTHex tries to equalize element area. It also averages the sizing functions of each node on the triangle as a secondary weight. The new node position is a combination of these two weighting functions. Two coefficients c_1 and c_2 are defined, where $c_1 + c_2 = 1$. These factors represent how much the element area and sizing function parameters affect the node movement. Increasing the weight on the triangle size c_1 generally improves the quality, while increasing the weight on the sizing function c_2 more accurately captures the geometry. The algorithm for local smoothing is shown below. See also Figure 11.

Step 1: Input a single node N;
Step 2: Input c_1 and c_2;
Step 3: For i = 1 to number of triangles attached to N;
    Calculate total area $A = \sum A_i$;
    Calculate sum of sizing functions $S = \sum 1/S_j$;
  end For;
  For i= 1 to number of triangles attached to N;
    Calculate $w1_i = \frac{A_i}{A}$;
    Calculate $w2_i = \frac{1/S_i}{S}$;
    Find C_i = the center of A_i;
    Calculate $N\prime + = C_i(c_1 w1_i + c_2 w2_i)$;
  end For;

Localized smoothing is applied to the node to which each edge is collapsed. It is also applied to nodes after edge swapping. Local smoothing does not optimize quality as well as a global technique, because each node is only moved once. However, the benefits of increased speed and sizing accuracy outweigh the costs. In the examples, the local smoothing actual outperformed global smoothing in quality measurements as well due to the fact that global smoothing would time out before it converged.



$$w1_i = \frac{A_i}{\Sigma A_i} \qquad w2_i = \frac{1/S_i}{\Sigma 1/S_i}$$

$$N' = \Sigma\, C_i (c_1 w1_i + c_2 w2_i)$$

Fig. 11. Local smoothing

In the following example quality and speed were compared for two different values of ϵ with global and local smoothing. The global smoothing technique took 631 seconds for ϵ=0.01 versus 16.6 seconds for local smoothing on ϵ=0.01. For ϵ=0.05, global smoothing took 455 versus 21.2 seconds. It is interesting to point out that the direction of time increase is reversed between global smoothing and local smoothing. In other words, the increase in ϵ results in a decrease in time for global smoothing, but an increase in time for local smoothing. This is because in global smoothing the time to smooth is proportional to the number of elements in the mesh, while in local smoothing, the time to smooth is proportional to the number of edge collapses. In quality comparisons, the global smoothing had an average quality of 0.7662 versus 0.9105 with ϵ=0.01. Figures 12 and 14 show the resultant meshes. For ϵ=0.05, the global smoothing had a quality of 0.6587 versus 0.8987. In Figure 13 the mesh has started to collapse from the boundary towards the center. This is a problem that was observed with several global smoothing applications. Such a mesh is unsuitable for any type of analysis. By comparison, the local smoothing performed very well in this application. (See Figure 15).

V. Refinement Procedures–THexing and Dicing

After a suitable surface mesh is created using the curvature based sizing function, edge swapping, and smoothing, a tetrahedral mesh is created for the interior of the model. In this research, the tetrahedral mesh was created using the CUBIT software which is based on the work of [9] and [10]. The final steps are to refine the mesh using THex and Dicing procedures.

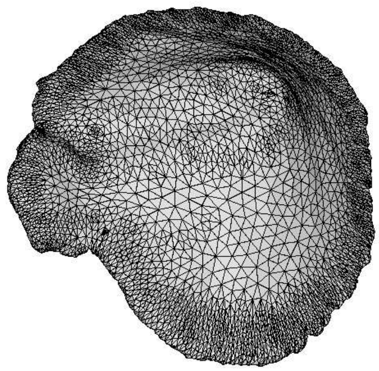


Fig. 12. Global smoothing with $\epsilon = 0.01$

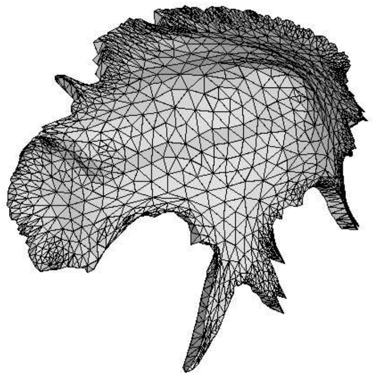


Fig. 13. Global smoothing with $\epsilon = 0.05$

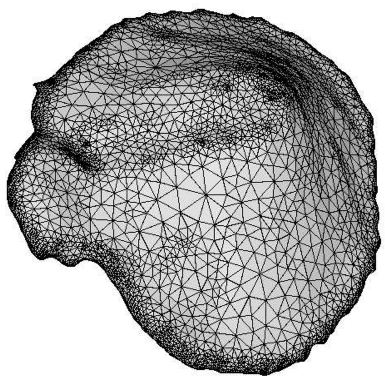


Fig. 14. Local smoothing with $\epsilon = 0.01$

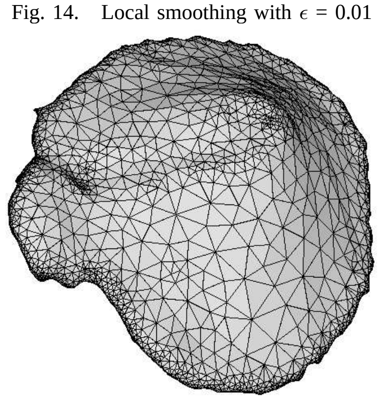


Fig. 15. Local smoothing with $\epsilon = 0.05$

TABLE II

AVERAGE SHAPE QUALITY AND CPU TIME FOR GLOBAL AND LOCAL SMOOTHING METHODS (SEE FIGURES 12 TO 15)

	ϵ	Shape	CPU Sec.
Global Smoothing	0.01	0.7662	631
	0.05	0.6587	455
Local Smoothing	0.01	0.9105	16.6
	0.05	0.8987	21.2

A. THex

THexing was introduced as the process of splitting every tetrahedron in the mesh into four hexahedra. THexing is a simple procedure, and it is used frequently because of its ease of implementation. For example, Pelessone and Charman [20] use this procedure in adaptive non-linear structural analysis where local refinement of a hexahedral mesh is needed. The advantages of THexing are its speed, lack of user intervention, geometric generality, orientation insensitivity, and element size control. The disadvantages are that it has poor boundary sensitivity, increased element count, and poor element quality. Another problem with the THexing is the node projection problem. When edges that are on convex surfaces are split during THexing, the interior node must be projected inward. See Figure 16. This often creates inverted elements. There are different approaches to solving this problem, but those are outside the scope of this research. The assumption here is that the curvature-based coarsening will help to alleviate that problem by using smaller elements on highly curved segments.

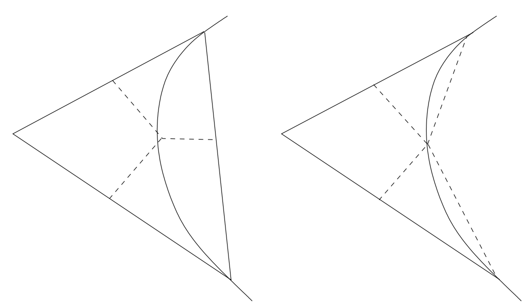


Fig. 16. Node projection problem on concave surfaces

B. Dicing

Dicing is a refinement procedure described by Melander [14] as a method of creating multi-million element meshes. Using the dicing method, every hexahedral element can be subdivided into smaller hexahedra. This is an 'h' modification that can improve analysis accuracy by producing a mesh with smaller element size, and improved angles. Dicing should be used with caution on mesh with a large number of nodes, since the increase in node count may exceed analysis capabilities. However, if used in conjunction with proper coarsening techniques, it can help to increase the overall quality of the mesh.

VI. Biological Examples

The main application of this research is for a specific class of hexahedral mesh generation, namely biological models. The stated goal was to improve element quality and reduce node count for these complex meshes. With that purpose in mind, this section highlights the features of the DTHex algorithm with several example problems. All of the examples will be compared to the existing THex approach in terms of quality and node count.

A. Sphere

The first example is a simple sphere. The sphere is an ideal mesh model because it is entirely convex. Because it is convex, the mesh does not develop any negative Jacobian elements when it is THexed as all projections are outward. The original sphere is meshed with a triangle scheme at an edge length size of one tenth the radius. The surface is coarsened with a coarsening factor of $\epsilon = 0.1$. The final mesh has an average shape quality of 0.8298 compared to a value of 0.6018 for the THexed mesh (see Table III). Figures 17 to 20 shows the model at various stages in the DTHex process. Figure 21 compares the THex approach to the DTHex approach. The final number of nodes with THexing is over 18000 while with DTHex it is around 2500.

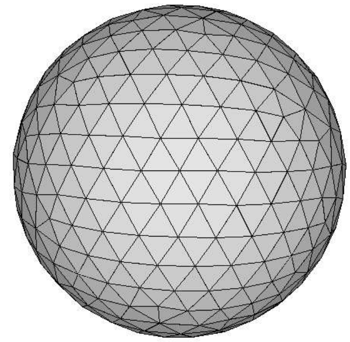


Fig. 17. Original mesh (728 triangles, 1092 edges, 366 nodes)

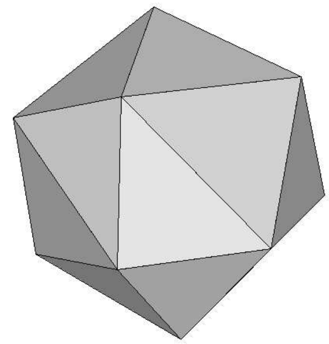


Fig. 18. Mesh after coarsening (20 triangles, 30 edges 12 nodes)

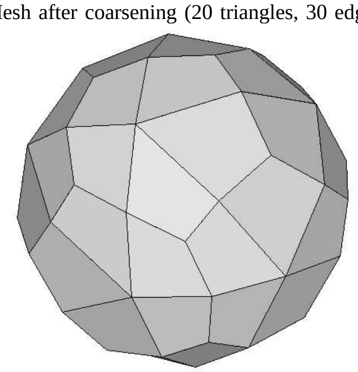


Fig. 19. Mesh after THexing (80 hexes, 60 faces, 120 edges, 125 nodes)

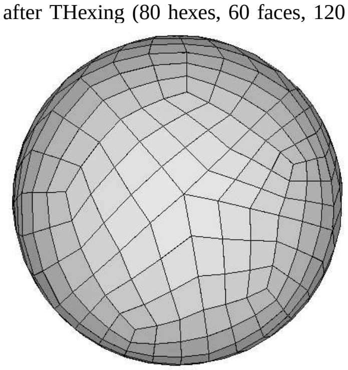


Fig. 20. Final hexahedral mesh (2160 hexes, 540 faces, 1080 edges, 2473 nodes)

TABLE III
SHAPE QUALITY FOR MESHES IN FIGURES 20 AND 21.

	Avg.	Std. Dev.	Min.	Max.
THex Mesh	0.6018	0.08286	0.3439	0.8230
DTHex Mesh ϵ = 0.1	0.8298	0.08310	0.5332	0.9809

B. Patella

The second example is a patella model (See Figure 22 to 25). The model appears to be very smooth, but it has regions of sharp dips and valleys with high curvature. The non-curvature-based coarsening function did not perform well on this analysis because it did not capture these features. The curvature-based coarsening algorithm was able to maintain the relative size of features while creating as coarse a mesh as possible in the regions of low curvature. The final hexahedral mesh did have some negative Jacobian elements, but there were many fewer than with the non-curvature-based coarsening function. Further research will be necessary to determine if it is possible to entirely eliminate these negative Jacobian elements. One temporary solution is to constrain the nodes so they don't project to the surface during the THexing process. Since the mesh approximates the surface curvature by its sizing function this may be a valid option if the user does not mind sacrificing

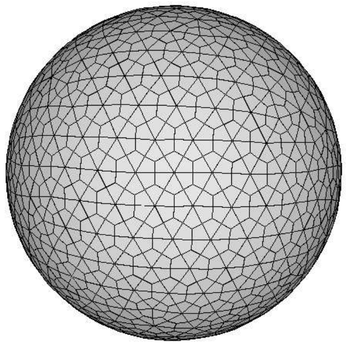


Fig. 21. A simple sphere meshed with the THex scheme (16176 hexes, 2184 faces, 4368 edges, 18647 nodes)

some of the surface definition. The user would be able to specify the amount of desired surface definition by setting the ϵ variable. Figure 26 compares the THex approach to the DTHex approach. The final number of nodes with THexing is 162000 while with DTHex it is around 120000. The average quality shown in Table IV is 0.5424 for the THex mesh and 0.6618 for the DTHex mesh.

TABLE IV
SHAPE QUALITY FOR MESHES IN FIGURES 25 AND 26.

	Avg.	Std. Dev.	Min.	Max.
THex Mesh	0.5424	0.1007	0.000	0.8084
DTHex Mesh ϵ = 0.1	0.6618	0.1114	0.000	0.9549

C. Tympanic Membrane

The third example is a tympanic membrane (See Figures 27 to 30). The tympanic membrane is part of the inner ear. This mesh is difficult because it is quite thin. This makes coarsening around the edges of the membrane difficult where there is a 180 degree turn over a very short distance. Curvature-based coarsening performed as expected by maintaining relatively small elements on the outer border, while allowing larger elements on the flat regions. Another problem with this model is the number of elements in the initial mesh. Re-meshing is not possible with faceted biological models because of the way they are defined. Thus the coarsest possible hexahedral mesh for this model is over 1.7 million nodes before the introduction of the DTHex scheme. DTHex decreases that number to 300000 nodes while also improving the quality and respecting curvature. The THex approach is shown in Figure 31. Average quality shown in Table V is 0.6320 for the DTHex approach compared to 0.5807 for the THex mesh.

VII. CONCLUSION

Hexahedral meshing is challenging because of the layered nature of hexahedral sheets. For this reason, it has been difficult to come up with an all-hexahedral meshing scheme that

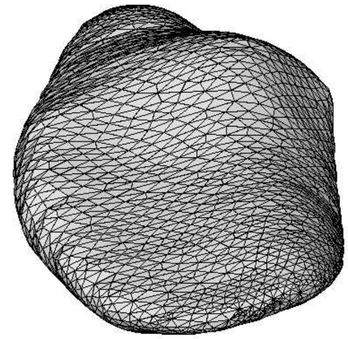


Fig. 22. A patella meshed using the DTHex scheme with $\epsilon = 0.1$ (a) Original mesh (5862 triangles, 8793 edges, 2933 nodes)

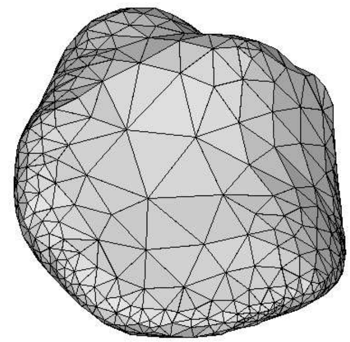


Fig. 23. Mesh after coarsening (1146 triangles, 1719 edges, 575 nodes)

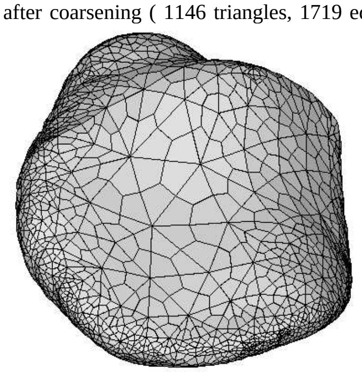


Fig. 24. Mesh after THexing (13704 hexes, 3438 faces, 6876 edges, 16647 nodes)

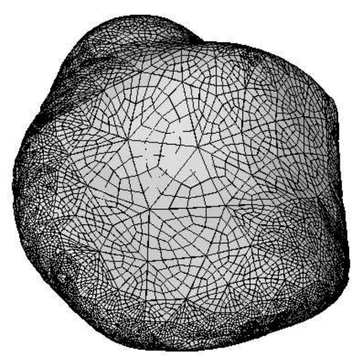


Fig. 25. Final hexahedral mesh (109632 hexes, 13752 faces, 27504 edges, 118955 nodes)

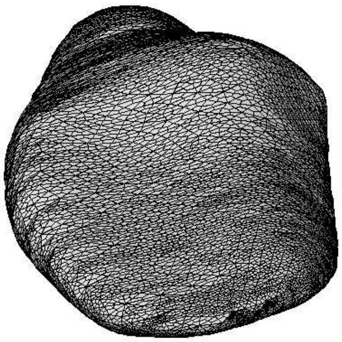


Fig. 26. A patella meshed with the THex scheme (141156 hexes, 17586 faces, 35172 edges, 161731 nodes)

TABLE V

SHAPE QUALITY FOR MESHES IN FIGURES 30 AND 31.

	Avg.	Std. Dev.	Min.	Max.
THex Mesh	0.5807	0.07854	0.000	0.8190
DTHex Mesh ϵ = 0.1	0.6320	0.1300	0.000	0.9562

is robust enough to produce good meshes on all geometries. THexing is one approach that has been used with moderate success. The work presented in this paper has shown that the DTHex method of producing all-hexahedral meshes can produce meshes with better quality and fewer elements than the classical THex approach. An additional benefit of the method is the ability to create meshes that approximate surface curvature.

This research has potential for great impact in the field of biological meshing. While there are other coarsening algorithms available in other meshing packages, the synthesis of parallel coarsening and refinement with curvature-based sizing functions, and local smoothing is unique to this research. The improved meshes will allow biological analysis to become more viable by decreasing the computing power necessary to perform the analyses.

This research also provides a framework for further studies on improving node projection techniques. It may be possible to completely eliminate the inverted elements that are formed when nodes on concave surfaces are projected inward. Further studies may also be done on automatically determining which ϵ to use during coarsening.

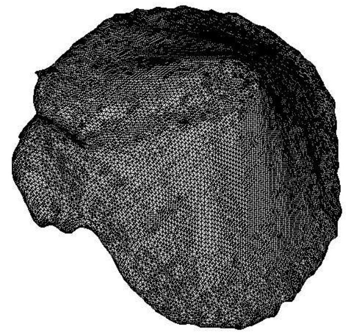


Fig. 27. A tympanic membrane meshed using the DTHex scheme with $\epsilon = 0.1$. Original mesh (49340 triangles, 74010 edges, 24672 nodes)

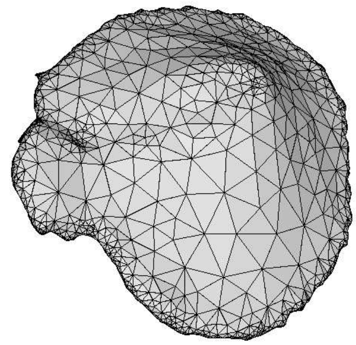


Fig. 28. Mesh after coarsening (4424 triangles, 6636 edges, 2214 nodes)

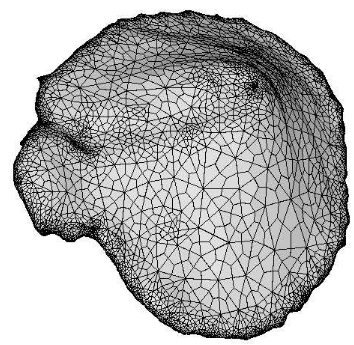


Fig. 29. Mesh after THexing (33144 hexes, 13272 faces, 26544 edges, 42899 nodes)

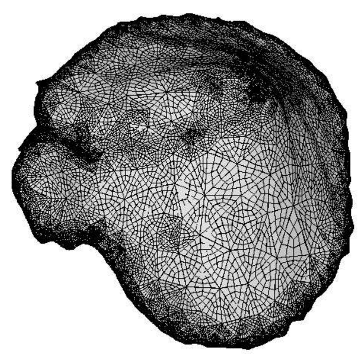


Fig. 30. Final hexahedral mesh (265152 hexes, 53088 faces, 106176 edges, 297933 nodes)

REFERENCES

[1] BENZLEY, S., PERRY, E., MERKLEY, K., AND CLARK, B. A comparison of all-hexahedral and all-tetrahedral finite element meshes for elastic and elasto-plastic analysis. In *Proceedings, 4^{th} International Meshing Roundtable* (1995), Sandia National Laboratories, pp. 179–191.

[2] BLACKER, T. Meeting the challenge for automated conformal hexahedral meshing. In *Proceedings, 9^{th} International Meshing Roundtable '00* (2000), Sandia National Laboratories, pp. 11-19.

[3] BLACKER, T. The cooper tool. In *Proceedings, 5^{th} International Meshing Roundtable '96* (1996), Sandia National Laboratories, pp. 13–29.

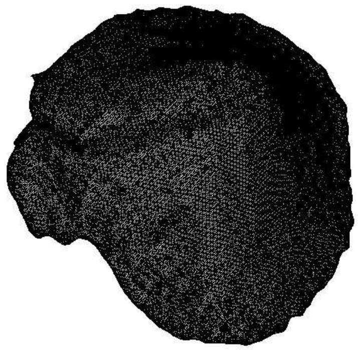


Fig. 31. A tympanic membrane meshed with the THex scheme (1546064 hexes, 148020 faces, 296040 edges, 1748575 nodes)

[4] BOROUCHAKI, H. AND FREY, P. Optimization tools for adaptive surface meshing. *Trends in Unstructured Mesh Generation 220* (1997), pp. 81–88.

[5] BUSSLER, MICHAEL L., AND RAMESH, A. The eight-node hexahedral element FEA of part designs. *Foundry Management and Technology v 121 n 11* (1993), pp. 26–28.

[6] CIFUENTES, A., AND KALBAG, A. Performance study of tetrahedral and hexahedral elements in 3-d finite element structural analysis. *Finite Elements in Analysis and Design 12* (1992), pp. 313–318.

[7] FREY, P.J. AND BOROUCHAKI, H. Unit surface mesh simplification. *Trends in Unstructured Mesh Generation 220* (1997), pp. 51–56.

[8] FREY, P.J AND BOROUCHAKI, H. Geometric evaluation of finite element surface meshes. *Finite Elements in Analysis and Design v 31* (1998), pp. 33–53.

[9] GEORGE, P. *Automatic Mesh Generation*. John Wiley & Sons, 1991.

[10] GEORGE, P.L., HECHT, F., AND SALTEL, E. Automatic mesh generator with specified boundary. *Computer Methods in Applied Mechanics and Engineering v 92* (1991), pp. 269–288.

[11] JONES, R.E., *QMESH: A Self-Organizing Mesh Generation Program* SLA - 73 - 1088, Sandia National Laboratories, (1974).

[12] KNUPP, P. Algebraic mesh quality metrics for unstructured initial meshes. *Finite Elements in Design and Analysis v 39 n 3* (2003), pp. 217-241.

[13] KNUPP, P. Next-generation sweep tool: a method for generating all-hexahedral meshes on two-and-one-half dimensional geometries. In *Proceedings, 7^{th} International Meshing Roundtable '98* (1998), Sandia National Laboratories, pp. 505–513.

[14] MELANDER, D., TAUTGES, T., AND BENZLEY, S. Generation of multi-million element meshes for solid-based geometries: the dicer algorithm. *Trends in Unstructured Mesh Generation 220* (1997), 131–135.

[15] MESHKAT, S. AND TALMOR, D. Generating a mixed mesh of hexahedra, pentahedra and tetrahedra from an underlying tetrahedral mesh. *International Journal for Numerical Methods in Engineering v 49* (2000), pp. 17-30.

[16] MINGWU, L., AND BENZLEY, S. A multiple source and target sweeping method for generating all hexahedral finite element meshes. In *Proceedings, 5^{th} International Meshing Roundtable '96* (1996), Sandia National Laboratories, pp. 217–225.

[17] MIYOSHI, K., AND BLACKER, T. Hexahedral mesh generation using multi-axis cooper algorithm. In *Proceedings, 9^{th} International Meshing Roundtable '00* (2000), Sandia National Laboratories, pp. 89–97.

[18] MURDOCH, P. *The Spatial Twist Continuum: A Dual Representation of the All Hexahedral Finite Element Mesh*. PhD thesis, Brigham Young University, 1995.

[19] OWEN, S. AND WHITE, D. Mesh-based geometry. *International Journal for Numerical Methods in Engineering v 58* (2003),pp. 375-395.

[20] PELESSONNE, D. AND CHARMAN, C.M. Adaptive finite element procedure for non-linear structural analysis. *Developments in Pressure Vessels and Piping v 301* (1995),pp. 191–198

[21] TAUTGES, T., BLACKER, T., AND MITCHELL, S. The whisker weaving algorithm: a connectivity-based method for all-hexahedral finite element meshes. *International Journal for Numerical Methods in Engineering 39* (1996), pp. 3327–3349.

[22] WEINGARTEN, V.I. The controversy over hex or tet meshing. *Machine Design v 66 n 8* (1994), pp. 74–78.

[23] WEISS, M. *Data Structures and Algorithm Analysis*. Benjamin Cummings Publishing, Redwood City, CA, 1995.