

Converting a Tetrahedral Mesh to a Prism-Tetrahedral Hybrid Mesh for FEM Accuracy and Efficiency

Soji Yamakawa
Mechanical Engineering
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213
+1-412-268-8880
soji@andrew.cmu.edu

Kenji Shimada
Mechanical Engineering
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213
+1-412-268-3614
shimada@cmu.edu

Abstract

This paper presents a computational method for converting a tetrahedral mesh to a prism-tetrahedral hybrid mesh for improved solution accuracy and computational efficiency of finite element analysis. The proposed method inserts layers of prism elements and deletes tetrahedral elements in sweepable sub-domains, in which cross-sections remain topologically identical and geometrically similar along a certain sweeping path. The total number of finite elements is reduced because roughly three tetrahedral elements are converted to one prism element. The solution accuracy of the finite element analysis improves since a prism element yields a more accurate solution than a tetrahedral element. Only previously known method for creating such a prism-tetrahedral mesh was to manually decompose a target volume into sweepable and non-sweepable sub-volumes and mesh each sub-volume separately. The proposed method starts from a cross-section of a tetrahedral mesh and replaces the tetrahedral elements with layers of prism elements until prescribed quality criteria can no longer be satisfied. The method applies a sequence of edge-collapse, local-transformation, and smoothing operations to remove or displace nodes located within the volume to be replaced with a layer of prism elements. Series of computational fluid dynamics simulations and structural analyses have been conducted, and the results verified a better performance of prism-tetrahedral hybrid mesh in finite element simulations.

Keywords: Tetrahedral Element, Prism Element, Hybrid Mesh, Mesh Conversion, Finite Element Method

CCS: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling – Geometric Algorithms, Languages, and Systems; J.6 [Computer Applications]: Computer-Aided Engineering – Computer-Aided Design

1. Introduction

This paper presents a computational method for converting a tetrahedral mesh to a prism-tetrahedral hybrid mesh for accuracy and efficiency of finite element simulation. The method replaces tetrahedral elements filling a sweepable volume, in which cross-sections remain topologically identical and geometrically similar along a certain sweeping path. The total number of finite elements is reduced by the conversion because roughly three tetrahedral elements are converted to one prism element. The overall accuracy of the finite element simulation is improved since a prism element yields a more accurate solution than a tetrahedral element due to the presence of a higher order terms in a shape function, which is used for interpolating physical quantity within the element.

The element type has a considerable impact on the accuracy of the finite element simulation. A volume mesh typically consists of four types of elements, tetrahedral (tet), hexahedral (hex), prism, and/or pyramid elements as shown in Figure 1. Each type of element performs differently in the finite element simulation due to the difference of the shape function, which interpolates physical quantity within the element. A hex element yields the most accurate solution, and an all-hex mesh is preferred for a finite element analysis when available. Although a tet mesh can be created easily by an automatic mesh generation scheme [George 1999;Shewchuk 2002], it requires more elements than an all-hex mesh to obtain an equally accurate finite-element solution.

However, despite numerous attempts, no known method can automatically create an all-hex mesh of adequate quality for an arbitrary geometric domain [Blacker and Meyers 1993;Hariya et al. 2006;Kwak and Im 2002;Lai et al. 2000;Ledoux and Weill 2007;Maréchal 2001;Mitchell 1998;Quadros and Shimada 2002;Schneiders 1996;Shepherd et al. 2000;Staten et al. 2005;Taghavi 2000;Tautges et al. 1996;White and Tautges 2000;Yamakawa and Shimada 2002].

A hybrid mesh is an alternative to an all-hex mesh. The most common form of a hybrid mesh is a hex-dominant mesh [Meshkat and Talmor 2000;Meyers et al. 1998;Owen and Saigal 2000;Yamakawa and Shimada 2002], in which most of the volume is filled with hex elements, and the rest with prism and tet elements, and pyramid elements are inserted between a quadrilateral face and triangular faces as shown in Figure 2. However, some finite-element solvers do not accept a pyramid element, and it becomes necessary to connect two triangular faces

to a quadrilateral face. Such a connection is called a non-conformal connection. Some solvers accept neither pyramid element nor non-conformal transition, and applicability of a hex-dominant mesh is significantly limited.

Another form of a hybrid mesh is a prism-tet mesh. A conformal prism-tet mesh can be created by filling sweepable volumes with prism elements and the rest with tet elements as shown in Figure 3. It includes neither pyramid elements nor non-conformal transition. This type of prism-tet mesh effectively reduces the number of elements, while providing a more accurate finite element solution, compared to a tet mesh of similar density.

The only method to create this type of prism-tet mesh was manually decomposing the geometric domain into sweepable and non-sweepable sub-domains, and creating prism elements in the sweepable sub-domains and tet elements in non-sweepable sub-domains. However, an inappropriate decomposition may yield low-quality tet elements. When this happens, the user needs to go back to the original geometric model and start over the manual decomposition.

For example, although the middle section of the leg-bone model shown in Figure 4 is clearly sweepable, the boundary between the sweepable and non-sweepable sub-domains is not so clear. A conservative choice is only taking short section of the bone as a sweepable sub-domain, and it yields too few prism elements to gain full benefit of the prism elements. An aggressive choice is to take entire bone as a sweepable domain, and it yields distorted elements over bone features. An optimal choice is in between conservative and aggressive choices, and it should fill as large volume as possible with prism elements within certain element-quality criteria. Since it is difficult to accurately estimate mesh quality before actually making a mesh, the user needs to repeat volume decomposition and meshing over and over again until optimal volume decomposition is reached.

The proposed method is advantageous compared to the conventional manual volume decomposition approach because:

- (1) No manual volume decomposition is necessary.
- (2) The method can be mostly automated.

The proposed method progressively inserts layers of prism elements starting from a cutting plane. During the process, it applies sequence of mesh-modifying operators called local transformation [Joe 1995;Klingner and Shewchuck 2007], smoothing [Calvo and Idelsohn 2001;Freitag and Plassmann 2001;Knupp 2000;Zhou and Shimada 2000], and edge collapse [Molinari and Ortiz 2002]. Although these operators effectively modify a tet mesh, the outcome depends on the order in which the operations are applied. For example, an operation applied earlier can adversely affect an outcome of later operations.

The proposed method overcomes the order-dependency problem of the mesh-modifying operations by devising a data structure for keeping track of and reversing the changes made by the mesh-modifying operations. With the data structure that enables recovery from an unsatisfactory outcome, the proposed method can try different sets of operations until a satisfactory outcome is obtained. If an outcome is unsatisfactory, the proposed method restores the previous state of the mesh, modifies the guess, and applies a sequence of mesh-modifying operations again until the outcome is satisfactory or no more reasonable guess can be made.

In the following, the proposed mesh-conversion method is explained in details in Section 2. Some experimental results including finite element simulations are presented in Section 3, and Section 4 concludes the paper.

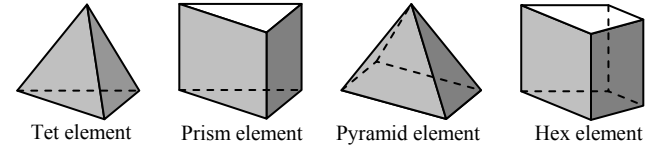


Figure 1 Four types of volume element

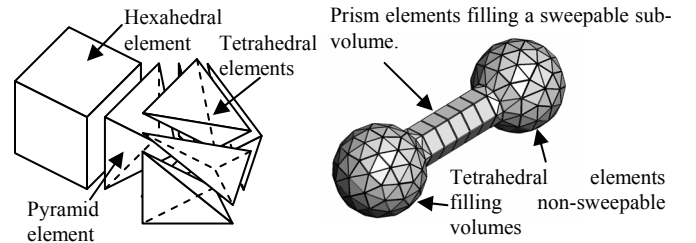


Figure 2 Transition from a hex element to tet elements through a pyramid element

Figure 3 Prism-Tet hybrid mesh of a dumbbell

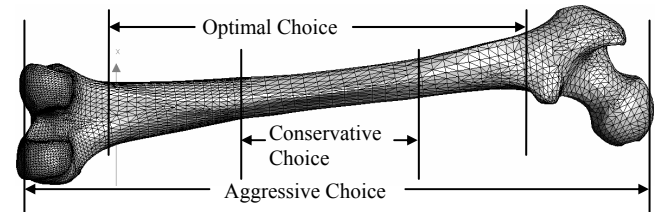


Figure 4 Possible Manual Volume Decomposition of Sweepable Sub-Domain

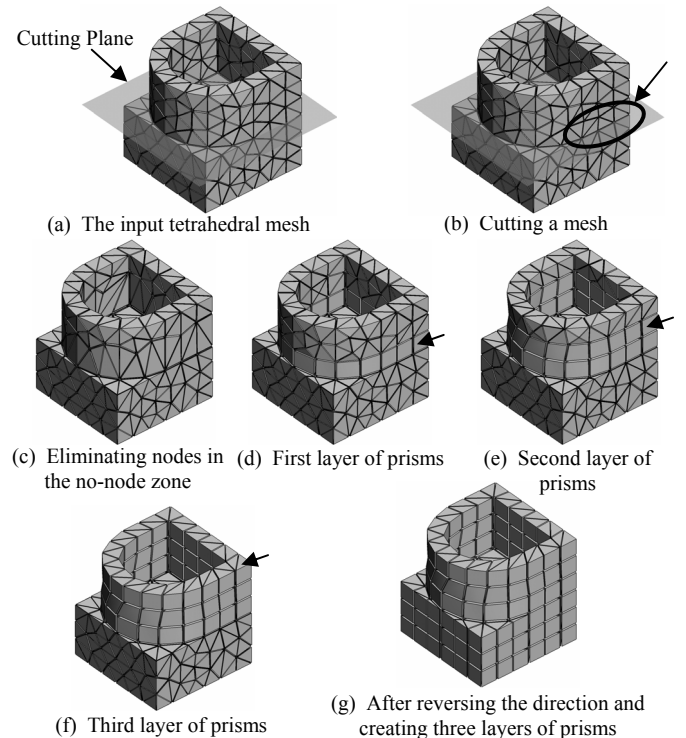


Figure 5 Overview of the proposed method

2. Prism-Layer Insertion

2.1 Overview of the Proposed Method

The proposed method initiates a prism-layer insertion process from a cutting plane that passes through a tet mesh. The proposed method first cuts a mesh by a cutting plane as shown in Figures 5 (a) and 5 (b). The cutting is achieved by eliminating all the edges intersecting the cutting plane by moving a node or inserting a node on the edge. The thickness of a prism layer h can be given by the user, or can be deduced from the average edge length of the triangular faces lying on the cutting plane.

The cutting plane is the initial frontal plane, which is denoted as S_f , and defined as:

$$(\mathbf{p} - \mathbf{o}(S_f)) \cdot \mathbf{n}(S_f) = 0,$$

where \mathbf{p} is a point on the surface, $\mathbf{o}(S_f)$ is a location of one of the nodes lying on S_f , and $\mathbf{n}(S_f)$ is the normal of S_f . $\mathbf{n}(S_f)$ also defines the direction in which layers of prisms grows. The growth plane S_1 is a plane that is parallel to S_f and passes through a point $\mathbf{o}(S_f) + h\mathbf{n}(S_f)$.

The proposed method attempts to insert a layer of prisms between S_f and S_1 . Such a layer cannot be inserted unless all nodes between S_f and S_1 are removed or displaced. In fact, remaining nodes should be reasonably distant from S_1 . Therefore, a guarding plane S_2 is introduced, which is parallel to S_f and passes through a point $\mathbf{o}(S_f) + h'\mathbf{n}(S_f)$ where $h' > h$. For making all the examples in this paper, $h' = 1.2h$ is used. The proposed method tries to remove all the nodes between S_f and S_2 before a layer of prisms is inserted as shown in Figure 5 (c). The detail of the node-removal process is described in Section 2.4.

Once all the nodes between planes S_f and S_1 are removed a layer of prisms with thickness of h is inserted on the frontal plane as shown in Figure 5 (d). Note that some node may remain between S_1 and S_2 to maintain the positive volume of elements.

A post-insertion check is performed to check the quality of the affected elements after the layer of prisms is inserted. If an element violates a user-specified quality threshold, the insertion and the modifications should be retracted, and the previous state of the mesh should be restored. The data structure for keeping track of the modifications is thus necessary. Detail of the layer insertion and the discussion of the data structure are described in Sections 2.5 and 2.8.

If no element violates the quality threshold, the frontal plane is updated to become the next S_f , and S_1 and S_2 are updated accordingly. Then the process of layer insertion is repeated until no more growth is possible without violating a quality threshold.

If the volume of the tet elements in front of the frontal plane is small, and the height of the tet elements is less than h , the tet elements will be deleted and an additional prism layer is inserted to fill the remaining height as shown in Figure 5 (f).

After the prism layers stop growing, the frontal plane is set back to the initial cutting plane with opposite normal direction, and inserts as many prism layers as possible as shown in Figure 5 (g).

2.2 Cutting Tet Elements by a Cutting Plane

In the first step, the proposed method makes a cut in a tet mesh by a cutting plane. The cutting plane is the initial frontal plane, on which a layer of prism elements are inserted, and it can be an arbitrary plane or can be one of the planar surface of the tet mesh.

When an edge is intersecting with the cutting plane, one of the nodes of the edge is moved to the intersection, or a new node is inserted on the edge at the intersection, and tet elements sharing the edge are subdivided accordingly.

One of the options, moving a node or inserting a new node, is selected based on the quality of the elements affected by the modification. The proposed method selects the option that maximizes the quality of the worst-shaped element. There are many choices of the quality measurements, and this research uses radius ratio, which is the radius of the circumscribed sphere divided by the radius of the inscribed sphere. An equilateral tetrahedron gives the radius ratio of 3.0, and the radius ratio approaches infinity as the shape of the element becomes thinner.

2.3 Testing the Growth Potential

A layer of prism elements can be inserted only when the cross section of the mesh with the frontal plane S_f is compatible with the cross section with the growth plane S_1 . Two cross sections are compatible if they are topologically equal and geometrically similar. To assure the compatibility, the proposed method checks the following two conditions before inserting each prism layer.

- (1) Two cross sections with S_f and S_1 are topologically identical.
- (2) No point on the cross section with S_f is more than $2h$ away from the cross section with S_1 when projected onto S_1 , or vice-versa.

A layer of prism elements will be inserted between S_f and S_1 , only if the two conditions are satisfied.

2.4 Node Removal by Edge-Collapse, Smoothing, and Local Transformation

The most crucial step of the prism-layer insertion is the node removal, in which the nodes between the frontal plane and the guarding plane are deleted or displaced.

A node of a tet mesh can be deleted by applying edge collapse [Molinari and Ortiz 2002], which merges two nodes of an edge into one node. If edge $\mathbf{p}_0 - \mathbf{p}_1$ is collapsed in order to delete \mathbf{p}_0 , the elements sharing the edge are first deleted, and then \mathbf{p}_0 is moved to and merged together with \mathbf{p}_1 .

Despite the simplicity of the edge-collapse operation, it does not always succeed. Let E_i be the i th tet element sharing node \mathbf{p}_0 , and assume all E_i s have a positive volume. T_i is one of the triangular faces of E_i that is not using \mathbf{p}_0 , and the T_i plane is a plane on which T_i is lying. Node \mathbf{p}_0 can move freely without making the volume of E_i negative as long as \mathbf{p}_0 does not cross the T_i plane. Therefore the volume in which node \mathbf{p}_0 can move freely without making a negative-volume element is a convex

volume constrained by the T_i planes. If an edge is connecting p_0 to a node that is outside the constraining volume, the edge cannot be collapsed.

For example, edge $p_0 - A$ shown in Figure 6 (a) cannot be collapsed because collapsing the edge moves p_0 across a plane on which triangle BCD is lying and will create a negative-volume tetrahedron $ABDC$.

A mesh smoothing scheme [Calvo and Idelsohn 2001; Freitag and Plassmann 2000; Freitag and Plassmann 2001; Knupp 2000; Zhou and Shimada 2000] may make an edge collapsible by moving nodes of the constraining triangles. However, there is no guarantee that a mesh smoothing scheme makes a particular edge collapsible.

Another potential solution is to apply local transformation [Joe 1995; Klingner and Shewchuck 2007]. The local transformation modifies node connections of a tet mesh and improves the quality of the elements.

To collapse edge $p_0 - A$ to remove node p_0 , the proposed method first moves p_0 toward A until volume of one of the tet element becomes zero as shown in Figure 6 (b). Tetrahedron $BDCp_0$ shown in Figure 6 (b) is a zero-volume tetrahedron. The local transformation will then try to find a configuration that removes the zero-volume tetrahedron. One potential outcome for the example shown in Figure 6 (b) is replacing tetrahedrons $BDCp_0$, $DBFp_0$, $DEFp_0$, and $CDEp_0$ with $BFEp_0$, $CBEp_0$, $FEDB$, and $DCBE$ as shown in Figure 6 (c). Edge $p_0 - A$ can be collapsed without making a negative-volume element after this transformation. Another potential outcome is replacing tetrahedrons $BDCp_0$, $ABCP_0$, and $ACBD$ ($ACBD$ is not drawn in the figure) with tetrahedrons $ABDp_0$ and $CADp_0$. In this case, $p_0 - A$ also becomes collapsible.

When node p_0 needs to be deleted and p_0 is connected to more than one edge that is collapsible or becomes collapsible after smoothing and/or local transformation, the proposed method chooses to collapse the edge that minimizes the radius ratio of the worst shaped element.

Even with smoothing and local transformation, there is a possibility that none of the edges connected to a node can be collapsed. If a node between S_f and S_2 remains undeleted, the proposed method moves the node to the normal direction of S_f as far as possible until the volume of one of the connected tet elements reaches zero. Although this temporarily yields a zero-volume element, such an element can be removed or improved by the subsequent smoothing and local transformation applied after a layer of prism elements is inserted.

2.5 Inserting a Layer of Prisms

After the nodes between S_f and S_1 are eliminated, a layer of prism elements is inserted between S_f and S_1 as shown in Figure 7. An angle-based smoothing scheme [Zhou and Shimada 2000] is then applied to the nodes lying on the growth plane, and a smoothing scheme called volume equalization [Yamakawa and

Shimada 2002] is applied to the nodes within two edges apart from the growth plane. The quality-improving local transformation is also applied to the tet elements incident to the growth plane.

2.6 Making the Final Layer

When the volume of the elements on the positive side of the frontal plane is small, the elements can simply be deleted, and a layer of prism elements can fill the volume. When the following 3 conditions are met, the proposed method deletes the elements on the positive side of the frontal plane, and fills the volume by a layer of prism elements.

- (1) The height of the elements on the positive side of the frontal plane is shorter than h' .
- (2) When projected onto the frontal plane, every node lies within $2h$ from or inside of the cross section with the frontal plane.
- (3) Deletion of the volume over the frontal plane does not change the genus of the mesh.

2.7 Mesh Quality Check

After a prism layer is inserted, the proposed method tests the quality of the elements created or deformed by the insertion. The quality of a tet element is measured by radius ratio, which is the circumscribed-sphere radius divided by the inscribed-sphere radius. The quality of a prism element is measured by the largest angle made by the two of the three edges connecting two triangular faces of a prism element. The proposed method also checks if the Jacobian determinant at each corner of the elements is positive. If this condition is satisfied, the frontal plane is updated to the other side of the inserted prism layer, and the processes described in Sections 2.3, 2.4, 2.5, and 2.6 are repeated to create another layer.

If the condition is not satisfied, the proposed method retracts all the changes made for inserting a prism-layer. This backtracking is made possible by using the data structure discussed in Section 2.8.

2.8 Data Structure for Keeping Track of the Changes of the Mesh-Modifying Operations

The proposed method intensively applies the mesh-modifying operations, local transformation, smoothing, and edge collapse. As discussed in Section 1, the outcome of the operations is order dependent, and it is very difficult to predict the outcome of a sequence of multiple operations; the outcome is unpredictable until the operations are actually applied.

When the outcome is unsatisfactory, the operations must be retracted, and the previous state of the mesh must be restored. However, without a data structure that keeps track of the mesh modifications, an entire mesh needs to be saved before the operations are applied, and it must be re-loaded when the outcome of the modifications were unsatisfactory. Saving and re-loading a mesh with large number of nodes and elements takes long time, and it is impractical to do it after every modification.

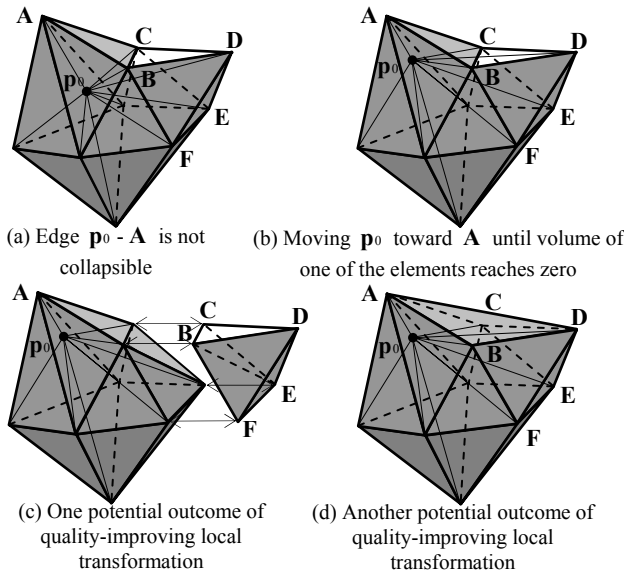


Figure 6 Applying Quality-Improving Local Transformation to make an edge collapsible

Instead of saving and re-loading an entire mesh, we have implemented a data structure specifically for keeping track of the changes made by the mesh-modifying operations. All the changes made by the operations are recorded, and when an outcome is unsatisfactory, the modifications are retracted by back-tracking the record.

In the current implementation, the program first tries to remove nodes between S_f and S_2 in the order they appear in the data structure. If some nodes remain undeleted, the program then tries the reverse order. If both attempts fail, the program tries to delete the nodes that remained undeleted in the first two attempts first, then the rest of the nodes. We believe there could be a better order that is more likely to succeed in one attempt and yields higher-quality elements. It is one of the future research topics.

3. Results

3.1 Mesh Conversion Results

The proposed method can deal with rotating cross sections such as a screw as shown in Figure 8. An all-prism mesh shown in Figure 8 (b) was created by applying the proposed mesh-conversion method to a tet mesh shown in Figure 8 (a). The cutting plane was the surface of the screw head connecting to the screw. In each step of the prism-layer insertion, locations of the nodes placed on the growth plane are calculated so that the original boundary of the mesh is preserved as much as possible.

A prism-tet hybrid mesh of an engine exhaust manifold shown in Figure 9 (b) was created from a tet mesh shown in Figure 9 (a). The proposed method successfully filled 82.6% of the total volume with prism elements, and the total number of elements was reduced from 101,024 to 42,304. Such a mesh is suitable for a CFD analysis, for example analyzing and optimizing the performance of an automobile engine. The proposed method can deal with a curved sweeping path with a small modification. However, no known method can find a curved sweeping path from a tetrahedral mesh, and it is one of the future research topics.

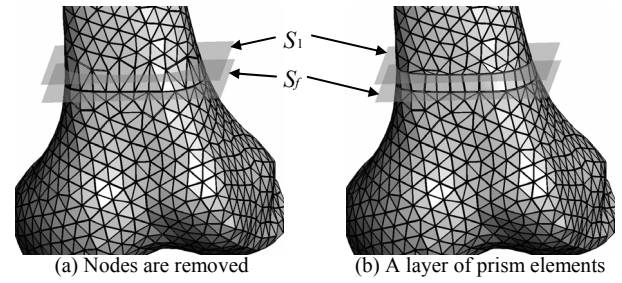


Figure 7 Inserting a layer of prism elements after removing nodes between the frontal plane and growth plane

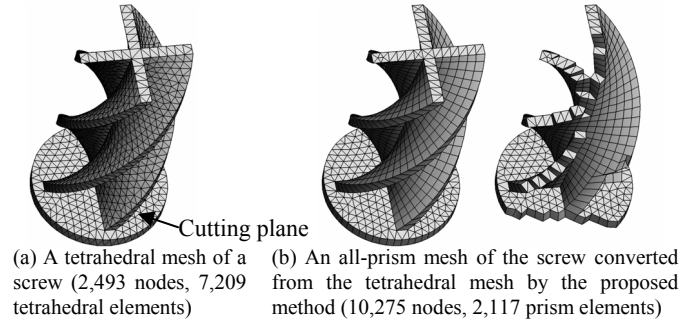


Figure 8 A tet mesh of a screw and an all-prism mesh converted from the tet mesh

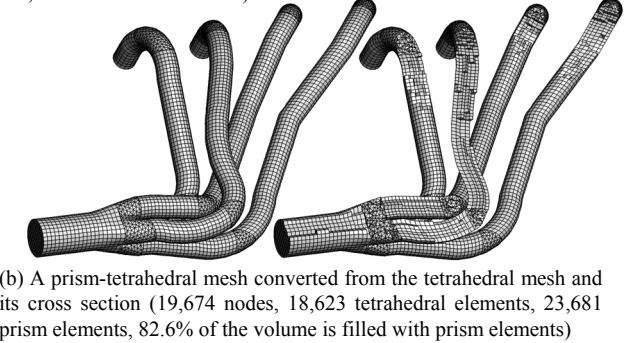
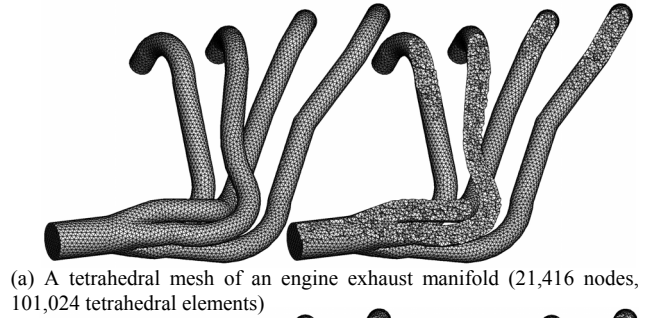


Figure 9 A tet mesh of an engine exhaust manifold and a prism-tet mesh converted from the tet mesh

3.2 CFD Analysis

3.2.1 Flow in a Pipe with a Bifurcation

A series of CFD simulations has been performed with prism-tet and tet meshes of a different mesh resolution to test the convergence characteristics.

The CFD simulations were performed using a commercial solver, Star-CD [CD-adapco], for a target domain, a bifurcating pipe with a rectangular cross section. Due to the symmetry, the fluxes coming out of the two outlets must be equal. However,

asymmetric node distribution yields a small difference between the two outlet fluxes. The difference between the two outlet fluxes must converge to zero as the mesh resolution increases.

Examples of prism-tet and tet meshes used in the series of CFD simulations are shown in Figure 10. The flux errors between the two outlets calculated by prism-tet meshes and all-tet meshes are plotted in Figure 11. The estimations of maximum error from prism-tet meshes and all-tet meshes are shown by the dashed lines in Figure 11.

The results indicate that prism-tet mesh created by the proposed method performed better than an all-tet mesh in a CFD simulation since a prism-tet mesh yielded a more accurate solution than an all-tet mesh with fewer elements.

3.2.2 Flow in an Air-Conditioner Duct

CFD simulations of an air-conditioner duct have been performed to demonstrate an applicability of a prism-tet mesh to a problem of a realistic setting. The inlet of the duct is 5.0m x 1.6m and connected to the main line of the air-conditioning system. The duct has four outlets with dimensions of 1.15m x 1.6m, 2.4m x 0.5m, 2.4m x 0.7m, and 2.4m x 1.4m. The inlet air flow speed is 4m/s. The purpose of the analysis is to measure how much of the air was delivered to each room from the outlets.

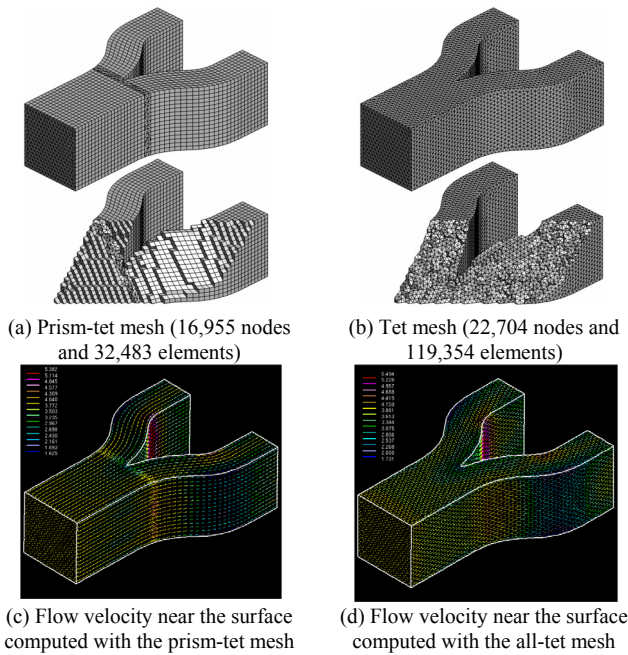


Figure 10 A prism-tet hybrid mesh and a tet mesh used in the comparison and plot of the result

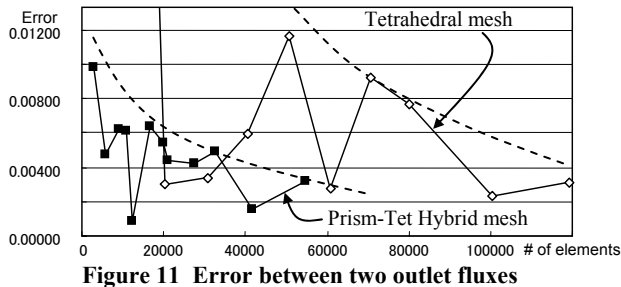


Figure 11 Error between two outlet fluxes

Table 1 Flux coming out from the outlets

Outlet	1	2	3	4
Prism-Tet mesh with boundary-layer elements	9.83476 m/s	2.87455 m/s	5.46455 m/s	20.3875 m/s
Tet mesh with boundary-layer elements	9.78998 m/s	2.89621 m/s	5.47693 m/s	20.3966 m/s

First, a tet mesh was created and a prism-tet mesh was created from the tet mesh by the proposed method. Then, ten layers of boundary-layer elements, 8mm thick in total, were inserted on the boundary wall by a conventional boundary-layer-meshing scheme [Garimella and Shephard 2000; Ito and Nakahashi 2002]. Since a boundary-layer element created on a quadrilateral face becomes a hexahedral element, the prism-tet mesh includes some hexahedral elements after boundary-layer elements were inserted. The meshes used in the simulations are shown in Figures 12 (a), and 12 (c).

Star-CD took 230 iterations, 1,502 seconds, and 414MB of memory to reach a steady-state solution with a tet mesh with boundary-layer elements. In contrast, it took only 193 iterations, 543.5 seconds, and 203MB of memory with a prism-tet mesh with boundary-layer element. The two results shown in Table 1 have only 0.5% difference, which is sufficiently accurate for estimating the air flow coming out of the outlets. The flow velocity plots with particle tracking are shown in Figures 12 (b) and 12 (d). The results indicate that the computational time and the memory usage can be considerably reduced by using a prism-tet mesh with boundary-layer elements.

3.3 Structural Analysis

The performance of prism-tet meshes has been tested by a series of structural analysis of a leg bone. To simplify the problem for the testing purpose, it is assumed to be a uniform isotropic material, and Young's modulus and Poisson ratio are 15GPa and 0.26 respectively. The length of the bone is 0.4m. The leg bone is cantilevered at the bottom, and 36kg (350N) is applied at the top end. Examples of the meshes used in the analyses are shown in Figures 13 (a) and 13 (c). Obtained stress plots are shown in Figures 13 (b) and 13 (d).

A commercial FEM package, ANSYS [ANSYS], calculated the solutions. The maximum principal stresses and maximum displacements as a function of the number of elements are plotted in Figures 13 (e) and 13 (f) respectively.

The convergence characteristic of prism-tet meshes is clearly better than tet meshes. Prism-tet meshes reach the convergence at around 60,000 elements, and tet meshes at around 10,000 elements. The computational time needed for reaching the convergence for prism-tet meshes and tet meshes are 198 seconds and 255 seconds respectively. By using a prism-tet mesh, the number of elements can be reduced by 40% and the computational time by 22% to obtain a convergent solution.

4. Conclusions

This paper has presented a computational method for converting a tet mesh into a prism-tet mesh by replacing some tet elements with layers of prism elements. The method effectively reduces the number of elements, and a resultant prism-tet mesh yields a more accurate solution than an all-tet mesh since a prism element approximates the finite element solution better than a tet element

does. A series of finite element simulations was performed to verify the superior performance of a prism-tet hybrid mesh and the effectiveness of the proposed mesh conversion method.

References

- ANSYS Inc. (<http://www.ansys.com>), ANSYS.
- Blacker, T.D. and Meyers, R.J. 1993. Seams and Wedges in Plastering: A 3-D Hexahedral Mesh Generation Algorithm. *Engineering with Computers* 2, 9, 83-93.
- Calvo, N.A. and Idelsohn, S.R. 2001. All-Hexahedral Mesh Smoothing with a Node-Based Measure of Quality. *International Journal for Numerical Methods in Engineering* 50, 1957-1967.
- CD-adapco (<http://www.cd-adapco.com>), Star CD.
- Freitag, L.A. and Plassmann, P. 2000. Local Optimization-based Simplicial Mesh Untangling and Improvement. *International Journal for Numerical Methods in Engineering* 49, 109-125.
- Freitag, L.A. and Plassmann, P.E. 2001. Local Optimization-based Untangling Algorithms for Quadrilateral Meshes. In *Proceedings of 10th International Meshing Roundtable*, 397-406.
- Garimella, R.V. and Shephard, M.S. 2000. Boundary Layer Mesh Generation for Viscous Flow Simulations. *International Journal for Numerical Methods in Engineering* 49, 193-218.
- George, P.L. 1999. Tet Meshing: Construction, Optimization and Adaptation. In *Proceedings of 8th International Meshing Roundtable*, 133-141.
- Hariya, M., Nishigaki, I., Kataoka, I. and Hiro, Y. 2006. Automatic Hexahedral Mesh Generation with Feature Line Extraction. In *Proceedings of 15th International Meshing Roundtable*, Birmingham, Alabama, 453-467.
- Ito, Y. and Nakahashi, K. 2002. Unstructured Mesh Generation for Viscous Flow Computations. In *Proceedings of 11th International Meshing Roundtable*, Ithaca, NY, 367-378.
- Joe, B. 1995. Construction of Three-Dimensional Improved-Quality Triangulations Using Local Transformations. *SIAM Journal on Scientific Computing* 16, 6, 1292-1307.
- Klingner, B.M. and Shewchuck, J.R. 2007. Aggressive Tetrahedral Mesh Improvement. In *Proceedings of 16th International Meshing Roundtable*, Seattle, WA, 3-23.
- Knupp, P.M. 2000. Achieving Finite Element Mesh Quality via Optimization of the Jacobian Matrix Norm and Associated Quantities. Part II - A Framework for Volume Mesh Optimization and the Condition Number of the Jacobian Matrix. *International Journal for Numerical Methods in Engineering* 48, 1165-1185.
- Kwak, D.-Y. and Im, Y.-T. 2002. Remeshing for Metal Forming Simulations-Part II: Three-Dimensional Hexahedral Mesh Generation. *International Journal for Numerical Methods in Engineering* 53, 2501-2528.
- Lai, M., Benzley, S. and White, D. 2000. Automated Hexahedral Mesh Generation by Generalized Multiple Source to Multiple Target Sweeping. *International Journal for Numerical Methods in Engineering* 49, 261-275.
- Ledoux, F. and Weill, J.-C. 2007. An Extension of the Reliable Whisker Weaving Algorithm. In *Proceedings of 16th International Meshing Roundtable*, Seattle, WA, 215-232.
- Maréchal, L. 2001. A New Approach to Octree-Based Hexahedral Meshing. In *Proceedings of 10th International Meshing Roundtable*, 209-221.
- Meshkat, S. and Talmor, D. 2000. Generating a Mixed Mesh of Hexahedra, Pentahedra and Tetrahedra from an Underlying Tetrahedral Mesh. *International Journal for Numerical Methods in Engineering* 49, 17-30.
- Meyers, R.J., Tautges, T.J. and Tuchinsky, P.M. 1998. The "Hex-Tet" Hex-Dominant Meshing Algorithm as Implemented in CUBIT. In *Proceedings of 7th International Meshing Roundtable*, 151-158.
- Mitchell, S.A. 1998. The All-Hex Geode-Template for Conforming a Diced Tetrahedral Mesh to any Diced Hexahedral Mesh. In *Proceedings of 7th International Meshing Roundtable*, 295-305.
- Molinari, J.F. and Ortiz, M. 2002. Three-Dimensional Adaptive Meshing by Subdivision and Edge-Collapse in Finite-Deformation Dynamic-Plasticity Problems with Application to Adiabatic Shear Banding. *International Journal for Numerical Methods in Engineering* 53, 1101-1126.
- Owen, S.J. and Saigal, S. 2000. H-Morph: an Indirect Approach to Advancing Front Hex Meshing. *International Journal for Numerical Methods in Engineering* 49, 289-312.
- Quadros, W.R. and Shimada, K. 2002. Hex-Layer: Layered All-Hex Mesh Generation on Thin Section Solids via Chordal Surface Transformation. In *Proceedings of 11th International Meshing Roundtable*, 169-180.
- Schneiders, R. 1996. A Grid-based Algorithm for the Generation of Hexahedral Element Meshes. *Engineering with Computers* 12, 168-177.
- Shepherd, J., Mitchell, S.A., Knupp, P. and White, D. 2000. Methods for Multisweep Automation. In *Proceedings of 9th International Meshing Roundtable*, 77-87.
- Shewchuk, J. 2002. Constrained Delaunay Tetrahedralizations and Provably Good Boundary Recovery. In *Proceedings of 11th International Meshing Roundtable*, 193-204.
- Staten, M.L., Owen, S.J. and Blacker, T.D. 2005. Unconstrained Paving & Plastering: A New Idea for All Hexahedral Mesh Generation. In *Proceedings of 14th International Meshing Roundtable*, San Diego, 399-416.
- Taghavi, R. 2000. Automatic Block Decomposition Using Fuzzy Logic Analysis. In *Proceedings of 9th International Meshing Roundtable*, 187-192.
- Tautges, T.J., Blacker, T. and Mitchell, S.A. 1996. The Whisker Weaving Algorithm: A Connectivity-Based Method for Constructing All-Hexahedral Finite Element Meshes. *International Journal for Numerical Methods in Engineering* 39, 3327-3349.
- White, D.R. and Tautges, T.J. 2000. Automatic Scheme Selection for Toolkit Hex Meshing. *International Journal for Numerical Methods in Engineering* 49, 127-144.
- Yamakawa, S. and Shimada, K. 2002. Hex-Dominant Mesh Generation with Directionality Control via Packing Rectangular Solid Cells. In *Proceedings of Geometric Modeling and Processing 2002*.
- Yamakawa, S. and Shimada, K. 2002. HEXHOOP: Modular Templates for Converting a Hex-Dominant Mesh to an ALL-Hex Mesh. *Engineering with Computers* 18, 211-228.
- Zhou, T. and Shimada, K. 2000. An Angle-Based Approach to Two-dimensional Mesh Smoothing. In *Proceedings of 9th International Meshing Roundtable*, 373-384.

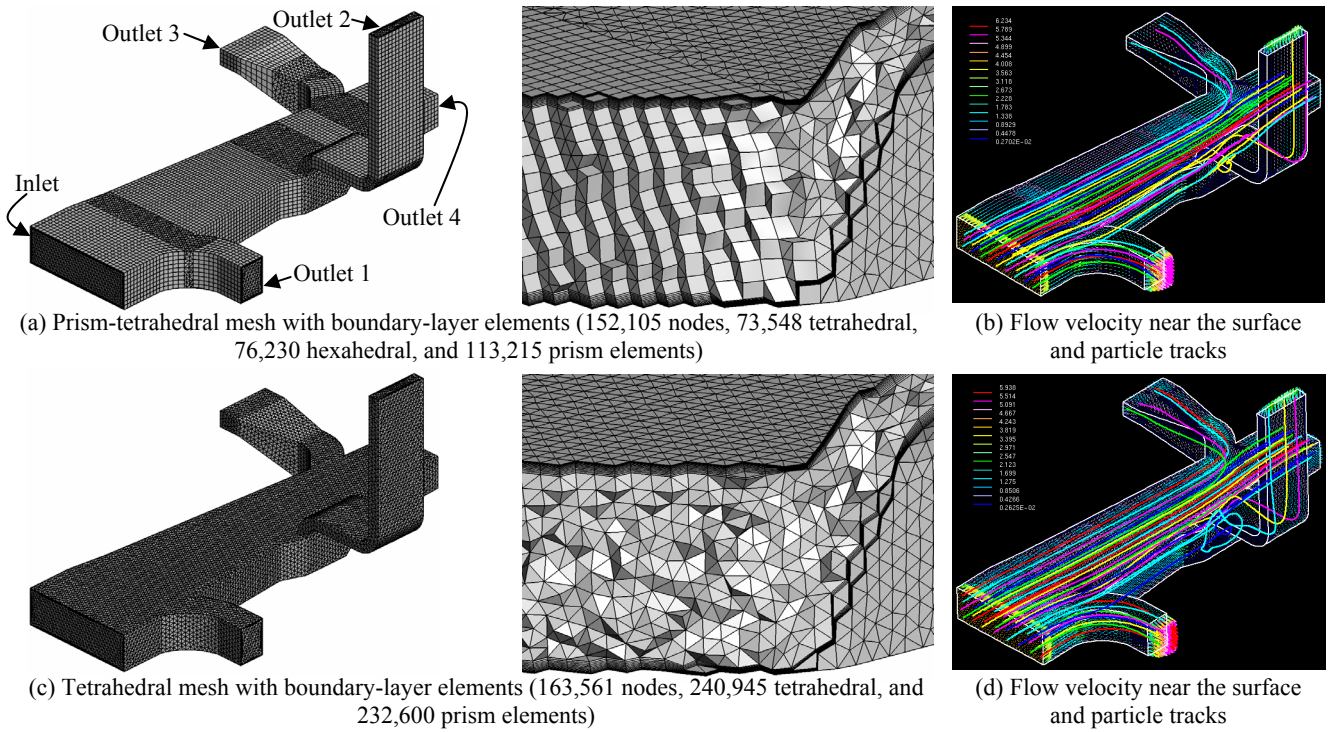


Figure 12 A CFD simulation of an air-conditioner duct

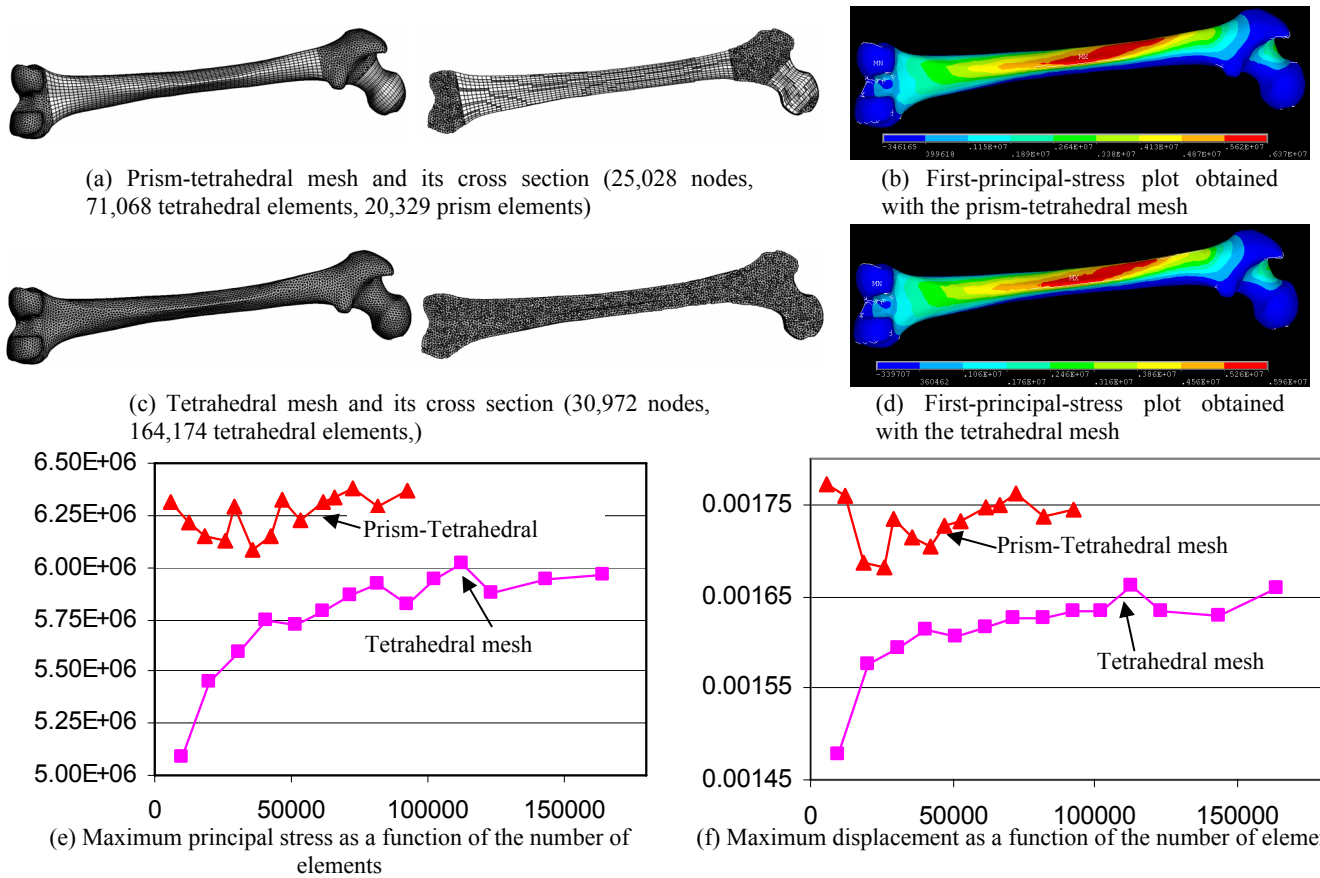


Figure 13 Structural analysis of a leg bone