Numerical Simulation of Liquid Composite Injection/Compression Molding. Part 1: Mesh Generation

K. M. Pillai and C. L. Tucker III*
Department of Mechanical & Industrial Engineering
University of Illinois
1206 W. Green St.
Urbana, IL 61801

F. R. Phelan Jr.
Polymer Composites Group
Polymers Division
Building 224, Room A209
National Institute of Standards and Technology
Gaithersburg, MD 20899

Submitted to Polymer Composites.

March 29, 1998

Abstract

This paper presents a numerical simulation of the injection/compression type of liquid composite molding (LCI/CM), where the fiber preform is compressed to a desired degree at the same time as the resin is being injected into the mold. Due to the possibility of an initial gap at the top of the preform and out-of-plane heterogeneity in the multi-layered fiber preform, a full three-dimensional (3D) flow simulation is essential. We propose an algorithm to generate a suitable 3D finite element mesh, starting from a two-dimensional shell mesh representating the geometry of the mold

*The author to whom the correspondence should be addressed

cavity. Since different layers of the preform have different compressibilities, and since properties such as permeability are a strong function of the degree of compression, a simultaneous prediction of preform compression along with the resin flow is necessary for accurate mold-filling simulation. The algorithm creates a coarser mechanical mesh to simulate compression of the preform, and a finer flow mesh to simulate motion of resin in the preform and gap. Lines connected to the top and bottom plates of the mold, called spines, are used as conduits for the nodes; this reduces the number of unknowns at a node point from the usual three components of displacement to a single scalar parameter marking position of the node along the spine. A method to generate a surface parallel to a given surface, thereby maintaining the thickness of the intermediate space, is used to construct the layers of the preform in the mechanical mesh. The mechanical mesh is further subdivided along the spines to create the flow mesh. The algorithm is robust and can generate mechanical and flow meshes from the shell mesh of any arbitrary three-dimensional mold shape. Since the algorithm assumes that gap height varies continuously with position in the shell mesh, it has the drawback of requiring a finer shell mesh near step changes in the mold thickness. Examples of the three-dimensional meshes generated by the algorithm are presented.

1 Introduction

Liquid composite molding (LCM) is emerging as an important technology to to make net-shape parts from polymer-matrix composites [1]. In any LCM process, a preform of reinforcing fibers is placed in a closed mold, then a liquid polymer resin is injected into the mold to infiltrate the preform. When the mold is full, the polymer is cured by a crosslinking reaction to become a rigid solid. Then the mold is opened to remove the part. LCM processes offer a way to produce high-performance composite parts using a rapid process with low labor content.

This paper deals with a particular type of LCM process called liquid composite injection/compression molding (LCI/CM). In LCI/CM, unlike other types of LCM processes, the mold is only partially closed when resin injection begins. This increases the cross-sectional area available for resin flow, and decreases flow resistance by providing a high porosity in the reinforcement. The fiber preform is slightly compressible, and normally fills the entire mold during this stage. After most of the resin has been injected, the mold is slowly closed to its final height, causing additional resin flow and saturating all portions of the preform. The LCI/CM process fills the mold more rapidly than LCM processes that use injection alone.

Complete filling of the mold with adequate wetting of the fibers is the primary objective of any LCM mold designer; incomplete filling in the mold leads to production of defective parts with dry spots. There are many factors which affect the filling of the mold: permeability of the preform, presence of gaps in the mold to facilitate resin flow, arrangement of inlet and outlet gates, injection rates of resin from different inlet ports, etc. [2, 3, 4]. Often it is not possible for the mold designer to visualize and design an adequate system for resin infusion by intuition alone, and mold filling simulations [4, 5, 6] are used to optimize mold performance. Situation in LCI/CM is more complex than ordinary LCM because of compression of the mold during the filling operation. As a result, numerical simulation of the mold filling process in LCI/CM becomes all the more important.

LCI/CM fiber preform frequently comprise layers of different reinforcing materials such as biaxial woven fabrics, stitch-bonded uniaxial fibers, random fibers, etc. Each type of material has a unique behavior as it is compressed in the mold. When such different materials are layered to form the preform, each of them will compress by different amounts as the mold is closed. This behavior is illustrated in Fig. 1, which shows a small piece of a mold. Here the lighter center layer deforms much more than the darker outer layer as the mold is closed.

Capturing this deformation behavior during compression is critical to accuracy of any LCI/CM process model. Resin flows through the preform at all stages of compression, and the porosity and permeability of the preform are critical in determining the resin flow. The ratio of deformed volume to initial volume determines the porosity of each preform layer, and from this one can determine the layer's permeability, either from a theoretical prediction or a correlation of experimental data [7, 8]. Because of this strong coupling between the

state of compression in a preform layer and its permeability, computations for fluid flow and preform compression have to be done simultaneously for mold filling simulations in LCI/CM.

Significant steps have already been taken to computationally model the mold filling in the LCI/CM process. A computer program called CRIMSON [9], is capable of isothermal mold filling simulation which involves simultaneous fluid flow and preform compression computations in the flow domain. But CRIMSON's initial capability is limited to two-dimensional (2D) planar geometries where prediction of preform compression is straightforward. Deformation of the preform is modeled using the incremental linearized theory of elasticity; the mathematics simplifies due to reduction in the number of degrees of freedom (DOF) associated with displacement from the usual three to one along the thickness direction. However typical parts made by the LCI/CM process usually have complicated three-dimensional shapes and this reduction of the mathematical complexity is no longer possible. The present paper describes our effort to expand CRIMSON's capability by enabling it to tackle any arbitrary non-planar three dimensional (3D) mold geometry.

Most injection molding simulation programs read for the mold geometry in the form of a shell mesh [5, 6, 10]. Even if it were possible to transmit the full geometrical information about the mold through a 3D mesh, it still is difficult to incorporate all the information of relevance to the process engineer. The latter needs to know the thicknesses of various layers of fiber mats and their corrosponding porosities at each time step. As a result, it is very important that elements representing different layers of preform in the 3D finite element mesh fall within separate layered regions. Overlap of an element onto more than one region is not acceptable as the element has to carry the material properties, such as porosity, permeability etc., of only one fiber mat. Mesh-generators in state-of-the-art commercial software such as Patran [11] are not designed to generate such a 3D mesh. Consequently, we decided to create a preprocessor suitable for LCI/CM mold filling simulation.

The objectives of this paper are to introduce basic ideas about modeling mold filling in three-dimensional LCI/CM parts, and to introduce an algorithm to generate a 3D finite element mesh from a given 2D shell mesh for preform and flow computations. In subsequent papers [12, 13], we will model finite deformation of preform using the non-linear theory of elasticity, and use this information to model resin flow in an LCI/CM mold.

2 Generating a 3D mesh from the given 2D shell mesh

Our aim is to develop a preprocessor that can generate 3D finite element meshes for flow computations starting from a 2D shell mesh. We wish to allow the LCI/CM process engineer to include all relevant information such as thicknesses of layers of the preform, thickness of the gap, etc. into the mesh.

Fig. 2 describes the three possible starting mold configurations (A, B and C) for a typical angular part geometry. Case A represents the starting configuration for the open mold injection/compression (I/C) molding, with ample gap between the top plate and preform.

Cases B and C occur when the gap is partly or completely eliminated before the start of the injection process. In the former, the preform is completely uncompressed with gaps at a few places. In the latter, the gap is removed at the cost of partial compression of the preform in certain regions. In the present paper, mesh generation for configuration A only will be addressed to. Once this mesh is created, cases B and C can be generated by solving for the mechanical compression of the preform [12].

As we shall see in the subsequent papers, six-noded wedge elements and eight noded brick elements are adequate for the modeling both the resin flow and preform compression. Our mesh generation algorithm is designed to generate such elements from the three- and four-noded triangular and quadrilateral elements of the shell mesh.

2.0.1 Mechanical and flow meshes

Development of the 3-D mesh for flow computations from a given 2-D shell mesh, representing the part geometry, is divided into two stages. In the first stage, an intermediate *mechanical* mesh is created, where the number of layers of elements equals the number of fiber mats in the lay-up, with the thickness of the mats equal to the height of those elements. Such a coarse mesh is adequate to track deformation of the mats during compression of the mold. In the second stage, the mechanical mesh is further subdivided along the thickness direction to create a more refined mesh, called the *flow* mesh, which is used for flow calculations.

2.1 Basic concepts of mesh generation algorithm

We first introduce two basic ideas which form the backbone of our mesh generation algorithm: spines and parallel surfaces.

2.1.1 Use of spines

One of the salient features of our mesh generation technique is the use of *spines* to track the nodes of the 3D mechanical mesh. This is similar to the use of spines in the free boundary problems [14, 5]. These spines are lines connecting node points of the top mold surface to their counterparts of the bottom mold surface.

In our scheme, nodes representing the preform between the two surfaces are constrained to move along these lines. Fig. 3 demonstrates the use of a spine to track the position of a point P of the mechanical mesh. The position of point P can be expressed as:

$$\vec{x_P} = \vec{X_i} + \lambda_P \hat{s}_i \tag{1}$$

where λ_P, $\hat{s}_i$ and $\vec{X_i}$ are the spine parameter, a unit vector representing the spine direction, and the position vector of node i of the shell mesh respectively. $\vec{x_P}$ is the position

vector of Point P. Since $\vec{X}_i$ is given and $\hat{s}_i$ can be computed, use of spines leads to a reduction in the nodal degree of freedom for mesh generation from the typical three position vector components to a single scalar variable λ (the spine parameter), representing the distance from the bottom surface along the spine.

Since the spines are connected to the upper and lower plates of the mold, they may rotate as the LCI/CM mold is compressed, as indicated in Fig. 5. This is relevant in modeling preform compression which will be considered in our next paper [12].

2.1.2 Generating a parallel surface

A basic action of the mesh generation algorithm is to generate a shell mesh for a surface which is parallel to the shell mesh of another surface. The generated surfaces represent either fiber-mat interfaces in the preform, or the moving upper plate of the LCI/CM mold. As shown in Fig. 4, it is not possible to generate the parallel surface by simple translation if the given surface is non-planar. In the following section, we propose a method to generate the parallel surface, based on the principle of minimizing error between the given and projected heights. This method yields a simple analytical solution in terms of spine parameters.

Using linear interpolation, the position of any point inside a triangular or quadrilateral shell element can be represented by:

$$\vec{X} = \sum_{i}^{n_{nie}} N_i \vec{X}_i \tag{2}$$

where $\vec{X}$ is the position vector of a point inside the element, N_i $(= N_i(\xi, \eta))$ is the shape function, $\vec{X}_i$ is the position vector of node i and n_{nie} is the total number of nodes in the element. ξ and η are the local coordinates within the element.

Now at any point in the element a vector $\vec{\mathcal{N}}$ normal to the surface is given by:

$$\vec{\mathcal{N}} = \left(\sum_{i}^{n_{nie}} \frac{\partial N_i}{\partial \xi} \cdot \vec{X}_i \right) \times \left(\sum_{i}^{n_{nie}} \frac{\partial N_i}{\partial \eta} \cdot \vec{X}_i \right) \tag{3}$$

The corresponding unit normal $\hat{n}$ can be computed by dividing $\vec{\mathcal{N}}$ by its magnitude.

If $\hat{n}_i^j$ denotes the unit normal at node i of element j, then the direction $\vec{S}_i$ of the spine passing through the node is computed as the average of normals of the surrounding elements:

$$\vec{S}_i = \frac{1}{n_{surr}} \sum_{j}^{n_{surr}} \hat{n}_i^j \tag{4}$$

where n_{surr} is the number of elements surrounding node i. Once again, the corresponding unit spine direction $\vec{s_i}$ is computed by dividing $\vec{S}_i$ by its magnitude.

The shell mesh of the second surface is generated by minimizing the net deviation of projected heights from the given height as follows,

$$I = \sum_{i}^{n_{nodes}} \left(\sum_{j}^{n_{surr}} \left(h_i^j - h_o^j \right)^2 A^j \right) \tag{5}$$

where I is the error function which is to be minimized, A^j is a weighting factor which is equal to angle subtended by element j at node i, h_i^j is the height of the top surface with respect to the bottom surface, and h_o^j is the specified thickness of the final part at element j.

Let $\vec{r_i} = \lambda_i \hat{s}_i$, where $\vec{r_i}$ is the displacement vector shown in Fig. 6. By projecting $\vec{r_i}$ onto $\hat{n}_i^j$, the heights can be computed as follows,

$$h_i^j = (\hat{n}_i^j \cdot \hat{s}_i)\lambda_i \tag{6}$$

On substituting Eqn. 6 into Eqn. 5, simplifying, and setting $\frac{\partial I}{\partial \lambda_i}$ equal to zero (see [15] for details), one gets a closed-form expression for the spine parameter,

$$\lambda_i = \frac{\hat{s}_i \cdot \left(\sum_j^{n_{surr}} h_o^j \hat{n}_i^j A^j \right)}{\sum_j^{n_{surr}} \left(\hat{n}_i^j \cdot \hat{s}_i \right)^2 A^j} \tag{7}$$

Since $\frac{\partial^2 I}{\partial {\lambda_i}^2} > 0$, the set of spine parameters thus obtained ensures a minimum of I. Substituting the λ_i's into Eqn. 1 gives the nodal coordinates of the desired parallel surface.

2.2 Algorithm

The main actions carried out in our mesh generation algorithm are as follows,

1. **Read data describing the 2-D shell mesh.**

   The mesh data is read, along with the information important for process modeling such as direction of clamping, properties of fiber mats, initial gap provided at the top of the preform, etc.

2. **Construct the upper surface of the final part.**

   The upper surface is generated by using the method described in Section Section 2.1.2. The input 2D shell mesh represents the bottom, immovable surface of the mold. The input thicknesses between the given and upper surfaces are taken to be the final thickness of the LCI/CM mold (equal to the desired part thickness).

3. **Compute the magnitude of the clamping vector.**

   Movement of the top plate with respect to the fixed bottom plate[1] is responsible for compression in the LCI/CM mold and distortion in the computational mesh. Motion

[1] This is a convention that we are adopting; it is equally possible for the top plate to remain stationary.

of the top plate is prescribed by a vector called the clamping vector, whose direction and magnitude are equal to the direction and extent of this motion (see Fig. 2).

The magnitude of the clamping vector is computed by translating the top mold plate (or surface) from its final position along the direction antiparallel to the clamping vector, until the height between the two mold plates equals the desired initial height between them (see Fig. 5). This initial height is equal to the sum of thicknesses of all fiber mats in the preform stack in the uncompressed state, plus the height of any initial gap provided above the preform.

$\vec{C}$ $(= |\vec{C}|\cdot\hat{c})$ is the negative of the clamping vector[2], where $|\vec{C}|$ is its unknown magnitude and $\hat{c}$ is its direction, given as a unit vector. Using the idea of Eqn. 6, the height of the translated top surface $\bar{h}_i^j$ is computed as follows,

$$\bar{h}_i^j = \hat{n}_i^j \cdot \left((\vec{X}_i^t + \vec{C}) - \vec{X}_i \right) \tag{8}$$

where $\vec{X}_i^t$ is the position vector of node i of the top shell mesh, corresponding to the final compressed position as obtained in the last step of the algorithm.

Here the error function to be minimized is

$$\bar{I} = \sum_{j=1}^{n_{elm}} \left(\bar{h}^j - \bar{h}_o^j \right)^2 B^j \tag{9}$$

such that $\bar{h}_o^j$ is the final uncompressed height of the preform at element j (equal to the sum of uncompressed widths of fiber-mats constituting the preform, plus a predetermined minimum gap), $\bar{h}^j$ the average of heights $\bar{h}_i^j$ over element j and B^j is the area of the element. The error function $\bar{I}$ is the sum of errors over all the *elements* of the shell mesh, and is different from the error function of Eqn. 5 which is computed over all the *nodes* of the shell mesh.

Using Eqn. 8, one can show that $\bar{h}^j$ is linear in $|\vec{C}|$ as follows,

$$\bar{h}^j = D_j + |\vec{C}| E_j \tag{10}$$

where

$$D_j = \frac{1}{n_{nie}} \sum_{i}^{n_{nie}} \hat{n}_i^j \cdot \left(\vec{X}_i^t - \vec{X}_i \right) \tag{11}$$

$$E_j = \frac{\hat{c}}{n_{nie}} \cdot \left(\sum_{i}^{n_{nie}} \hat{n}_i^j \right) \tag{12}$$

On substituting Eqn. 10 into Eqn. 9 and minimizing $\bar{I}$ with respect to $|\vec{C}|$, one gets,

$$|\vec{C}| = -\frac{\sum_{j=1}^{n_{elm}} E_j \left(D_j - \bar{h}_o^j \right)}{\sum_{j=1}^{n_{elm}} E_j^2} \tag{13}$$

[2]Since we are reversing the direction of the upper mold to recreate the initial top surface

Once again, the expression for $|\vec{C}|$ ensures a minimum for $\bar{I}$ since $\frac{\partial^2 \bar{I}}{\partial |\vec{C}|^2}$ is always positive.

4. **Solve for the rotated spines**

   When the upper mold surface is translated to its initial position, the position of a translated top surface point $\vec{\bar{X}}^t{}_i$ is given by,

$$\vec{\bar{X}}^t{}_i = \vec{X}^t_i + |\vec{C}|\hat{c} \tag{14}$$

   Once the coordinates of nodes of the translated top surface are known, the directions of new rotated spines $\hat{s}_i$ can be computed as:

$$\hat{s}_i = \frac{\vec{\bar{X}}^t{}_i - \vec{X}_i}{|\vec{\bar{X}}^t{}_i - \vec{X}_i|} \tag{15}$$

5. **Create the mechanical mesh.**

   The mechanical mesh is generated by using the method presented in Section 2.1.2. Increments in the spine parameters $\Delta\bar{\lambda}_{i,k}$ are calculated for the new rotated spines, where λ_i of Eqn. 7 now represents such increments. Nodes of each interface layers k are created using the Eqn. 1 as follows:

$$\vec{\bar{X}}_{i,k} = \vec{\bar{X}}_{i,k-1} + \Delta\bar{\lambda}_{i,k}\hat{s}_i \tag{16}$$

   where $\vec{\bar{X}}_{i,k}$ and $\vec{\bar{X}}_{i,k-1}$ are position vectors of node i of shell meshes of layers k and $k-1$ respectively. Now 3D elements of the mechanical mesh are created on the top of each 2D shell element by joining corresponding nodes of any two neighboring fiber mat interfaces to create the interfaces of the fiber mats in the preform. The spacing between any two such interfaces is equal to the uncompressed thickness of the corresponding fiber mat. The gap at the top of the preform is treated as an extra layer to generate the gap elements.

6. **Create the flow mesh.**

   The flow mesh is used to compute fluid flow through the multilayered preform, and may be much finer than the mechanical mesh. Therefore, each layer of the mechanical mesh is further subdivided into a pre-specified number of sublayers to create the flow mesh. Either Eqn. 1 or Eqn. 16 can be used to generate the nodes of intermediate layers of the flow mesh.

3 Examples and Discussion

A computer program has been developed to implement the mesh generation algorithm, and tested for its efficacy and robustness. In the following sections, examples of the creation of

three-dimensional computational meshes from two-dimensional shell meshes are presented. Since the thicknesses in LCI/CM parts are much smaller than their other dimensions, realistic meshes are relatively thin. To highlight important features of the algorithm, the thicknesses of the meshes are scaled up in the following examples. In each example, a gap which is a certain fraction of the total thickness of the uncompressed preform is provided between the upper surface of the preform and the top mold plate.

3D mesh from a planar rectangular shell mesh

Fig. 8 shows a mechanical mesh created from a rectangular shell mesh lying in the x-y plane as shown in Fig. 7. The shell mesh contains the two types of elements, quadrilaterals and triangles, which are used most often in LCM simulations [6]. In this example, each type of element corresponds to a separate material domain, and we assign different thicknesses to the layers of the preform in each domain. Due to averaging, the height of the topmost surface varies linearly across the boundary between the two domains. As a result of this averaging, a finer mesh is needed near the boundary for accurate representation of the step change in the mold height. A clamping vector along the $< 1, 1, 1 >$ direction is chosen to create a more complicated motion of the top plate.

At this point, the importance of weighting factors while computing direction of spines in Eqn. 7 should be mentioned. We discovered that using the areas of elements surrounding a node as weights in these equations leads to formation of "bumps" along an interface where elements of different types (triangles and quadrilaterals) touch (see Fig. 9). This unevenness of the top surface resulted from the difference in the number of elements surrounding alternating nodes along the interface, thereby changing the value of the spine parameter λ. The anomaly was corrected by using the included angle of each surrounding element at a node as the weighting factor A^j in Eqns. 5 and 7.

The mechanical mesh of Fig. 8 is further subdivided along the spine direction to create the flow mesh shown in Fig. 10.

3D mesh from an arbitrary shell mesh in three dimensions

Fig. 11 shows a shell mesh created from a surface patch that is curved and undulates in three dimensions. The quadrilateral elements making up the shell mesh are no longer co-planar, and the normals of the elements surrounding each node (used to compute the spine parameter in Eqn. 7) are no longer parallel. As a result, this provides a good test of the robustness and general applicability of our method. Fig. 12 shows the mechanical mesh created by stacking three layers of reinforcement, with one thin gap layer at the top. The negative of the clamping vector is once again $< 1, 1, 1 >$. As one can see, our method is successful in creating an adequate 3D mesh. This mesh can be further subdivided along the spines to create a flow mesh. The aspect ratio of the elements can be improved by refining the mesh.

4 Summary and Conclusions

In this paper, we present a methodology to create three-dimensional finite element meshes for modeling mold filling in LCI/CM. We propose the concept of predicting preform compression using the coarse mechanical mesh, and predicting fluid flow using the finer flow mesh. A mesh-generating algorithm, to create the mechanical and flow meshes from a given shell mesh, is presented. This algorithm incorporates information about the position of fiber mat interfaces in a multi-layered preform, which is crucial for accurate modeling of the filling process. A technique to create surfaces parallel to any arbitrary shell mesh surface enables us to represent the interfaces accurately. Further, the use of spines in mesh generation reduces the number of unknowns at each node from three to one. The algorithm is used successfully to create the mechanical and flow meshes from two different shell meshes; its robustness is demonstrated by creating a 3D mesh from a shell mesh for an arbitrary mold shape. The need to refine the shell mesh in the region of a step change in the thickness of the mold is the main limitation of the algorithm. In subsequent papers, we will use the mechanical and flow meshes to simulate preform compression and resin flow during mold filling in LCI/CM.

Acknowledgements

Financial support for this work was provided by the National Institute of Standards and Technology.

References

[1] C. D. Rudd, A. C. Long, K. N. Kendall, and C. G. E. Mangin. *Liquid Molding Technologies*. Woodhead Publishing Ltd., 1997.

[2] K. M. Pillai, T. L. Luce, M. V. Bruschke, R. S. Parnas, and S. G. Advani. Modeling the Heterogeneities present in Preforms during Mold Filling in RTM. In *Advanced Materials: Expanding the Horizons*, volume 25, 1993. 25th International SAMPE Technical Conference.

[3] K. M. Pillai and S. G. Advani. A Model for Unsaturated Flow in Woven Fiber Preforms during Mold Filling in Resin Transfer Molding. *Journal of Composite Materials*, to appear.

[4] S. G. Advani, editor. *Flow and Rheology in Polymer Composites Manufacturing*. Elsevier, 1994.

[5] C. L. Tucker III, editor. *Fundamentals of Computer Modeling for Polymer Processing*. Hanser Publishers, Munich, 1989.

[6] M. V. Bruschke and S. G. Advani. RTM. Filling Simulation of Complex Three Dimensional Shell-like Structures. *SAMPE Q.*, 23(1):2–11, October 1991.

[7] C. Lekakou, M. A. K. B. Johari, and M. G. Bader. Compressibility and flow permeability of two-dimensional woven reinforcements in the processing of composites. *Polymer Composites*, 17, 1996.

[8] N. Pearce and J. Summerscales. The compressibility of a reinforcement fabric. *Composites Manufacturing*, 6, 1995.

[9] F. R. Phelan Jr. Analysis of Injection/Compression Liquid Composite Molding Process Variants. In *ASME Proceedings*, November 1996. Presented at the 1996 ASME International Mechanical Engineering Congress and Exhibition Atlanta,Ga.

[10] F. R. Phelan Jr. Simulation of the injection process in resin transfer molding. *Polymer Composites*, 18(4):460–476, August 1997.

[11] PDA Engineering, 2975 Redhill Avenue, Costa Mesa, California 92626. *Patran 3: User Manual*, 1994.

[12] K. M. Pillai, C. L. Tucker III, and F. R. Phelan Jr. Numerical simulation of liquid composite injection/compression molding. part 2: Modeling preform compression. Manuscript in preparation.

[13] F. R. Phelan Jr., K. M. Pillai, and C. L. Tucker III. Numerical simulation of liquid composite injection/compression molding. part 3: Modeling mold-filling process. Manuscript in preparation.

[14] S. F. Kistler and L. E. Scriven. Coating flows. In J. R. A. Pearson and S. M. Richardson, editors, *Computational Analysis of Polymer Processing*. Applied Science Publishers, 1983.

[15] K. M. Pillai and C. L. Tucker III. Progress Report: January 1 to March 31. Technical report, University of Illinois, 1997.

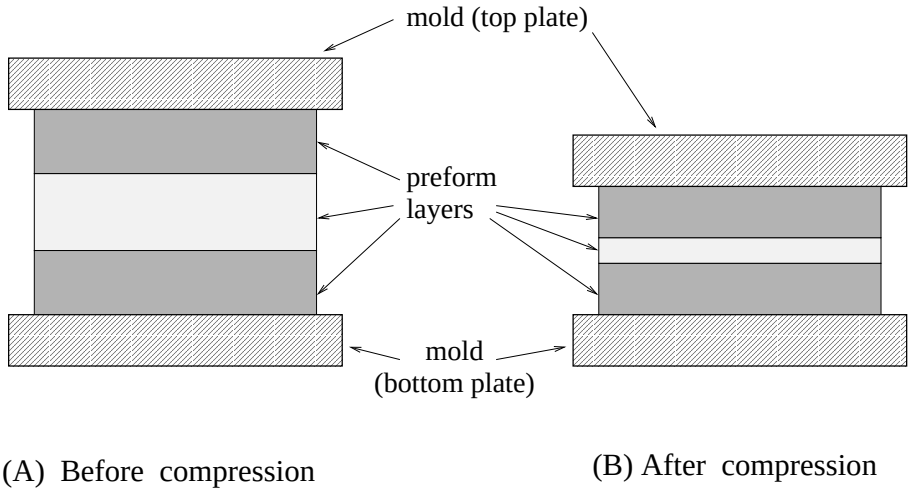

Figure 1: Uneven deformation of preform layers under compression.

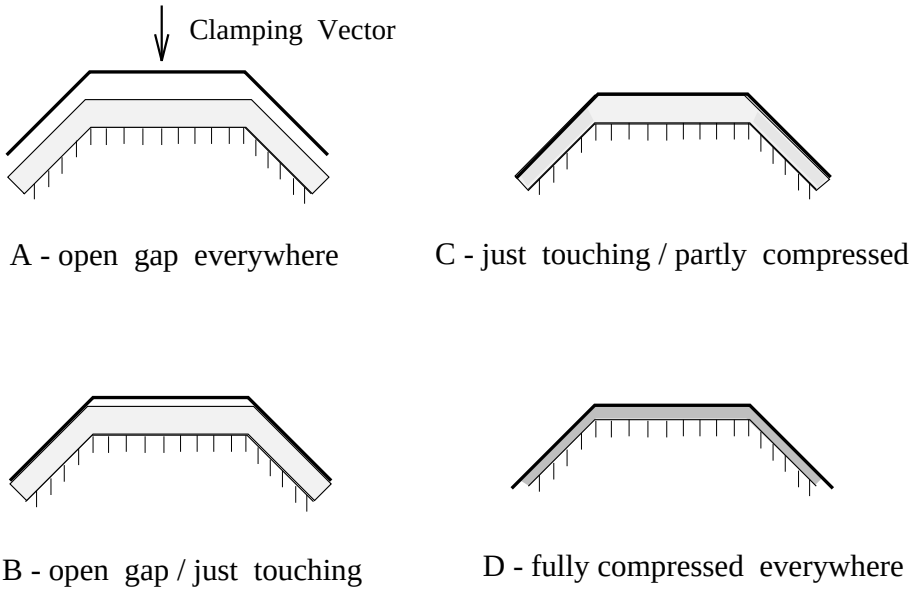

Figure 2: A schematic describing the various stages of the compression/injection molding process. The top plate of the mold moves along the clamping vector, while the bottom plate is stationary. Stages A, B and C are three possible starting positions of the top plate. Stage D shows the final configuration of the mold when it is fully compressed.

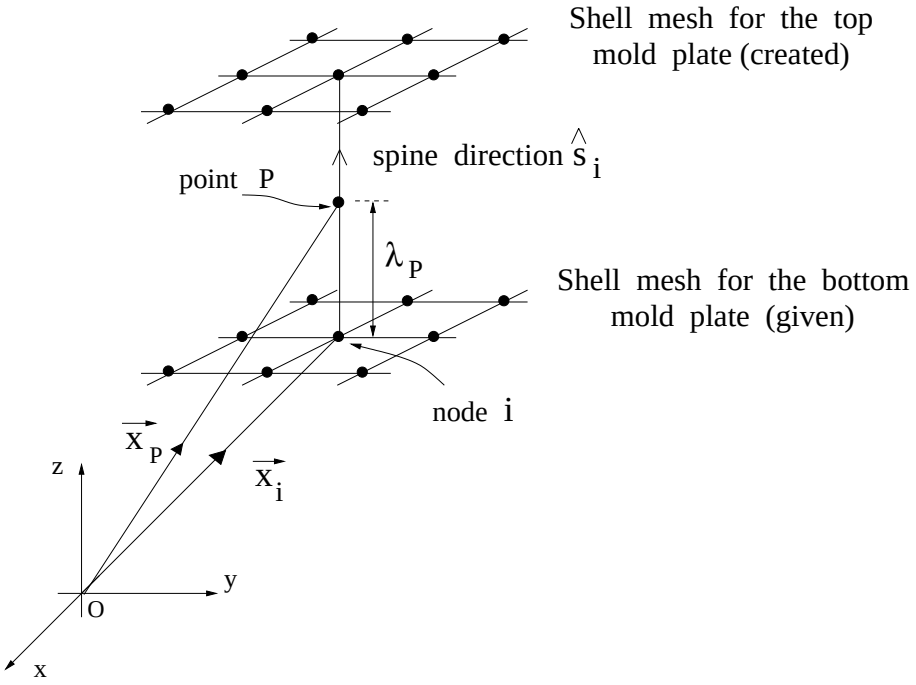

Figure 3: Expressing the position vector $\vec{x_P}$ of a point P of the mechanical mesh in terms of the spine direction $\hat{s}_i$, the node position $\vec{X}_i$, and the spine parameter λ_P.

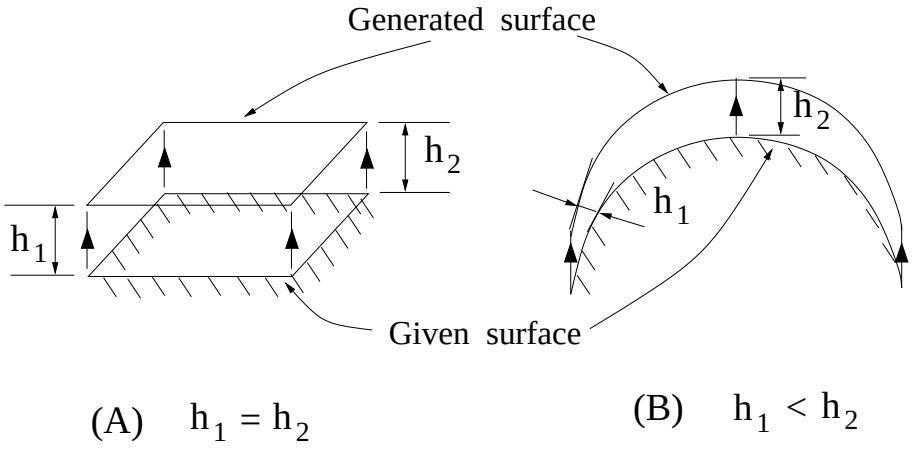

Figure 4: Simple translation of a surface can be used to generate a parallel surface if the given surface is planar (part A). This method fails if the given surface does not lie in a plane (part B).

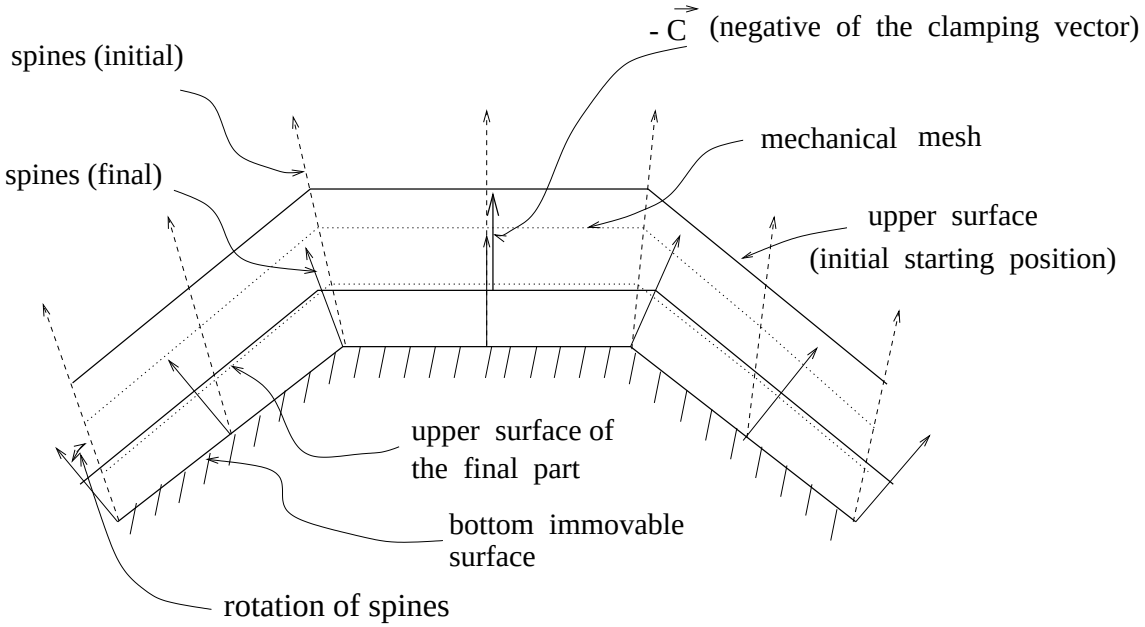

Figure 5: Spines and their role in the creation of meshes.

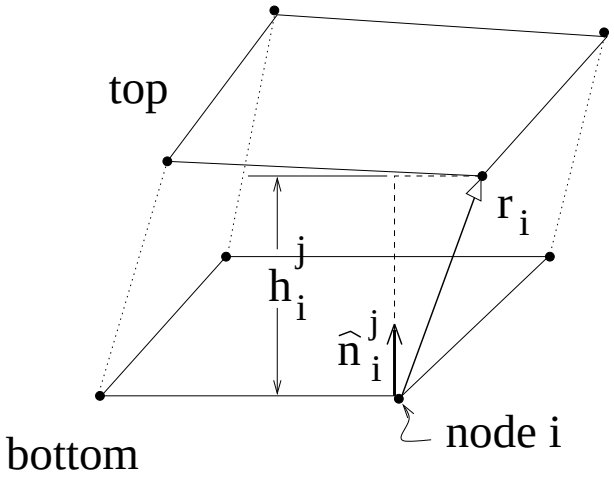

Figure 6: Definition of the height h_i^j between the top surface with respect to the bottom surface expressed in terms of the normal $\hat{n}_i^j$ and the displacement $\vec{r_i}$.

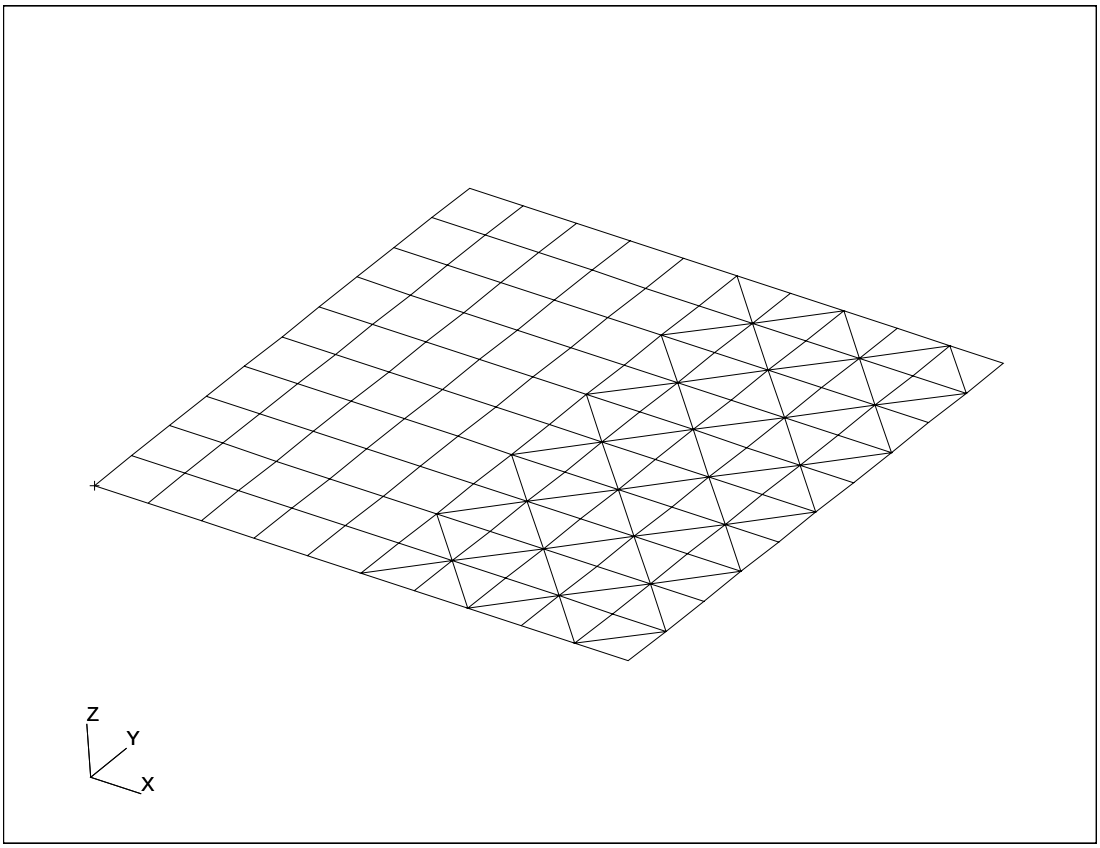

Figure 7: A shell mesh consisting of a rectangular planar patch of quadrilateral and triangular shell elements.

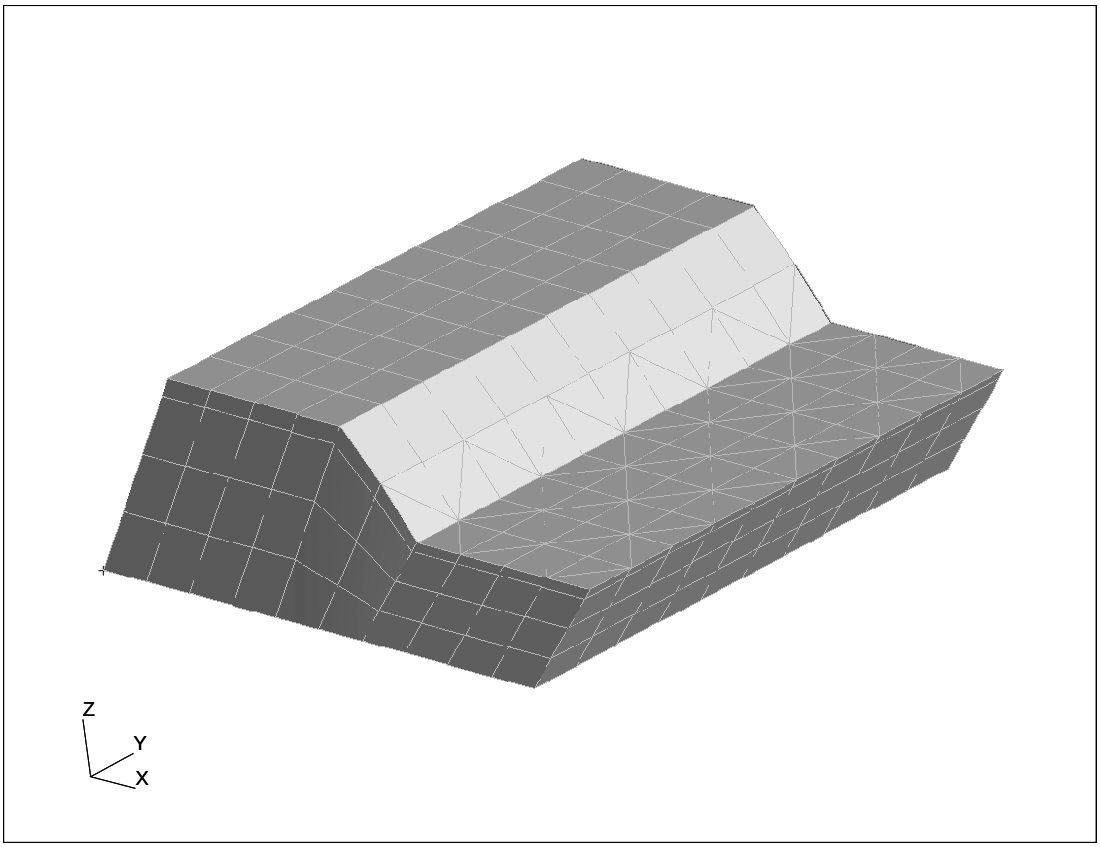

Figure 8: The mechanical mesh created from the shell mesh shown in Fig. 7. There are three layers of reinforcement with one thin gap layer at the top; the reinforcement layers are of equal thicknesses of 0.1 and 0.05 units in the two material domains, corresponding to the rectangular and triangular elements respectively. The negative of the clamping vector $\vec{C}$ corresponds to the direction $< 1, 1, 1 >$.

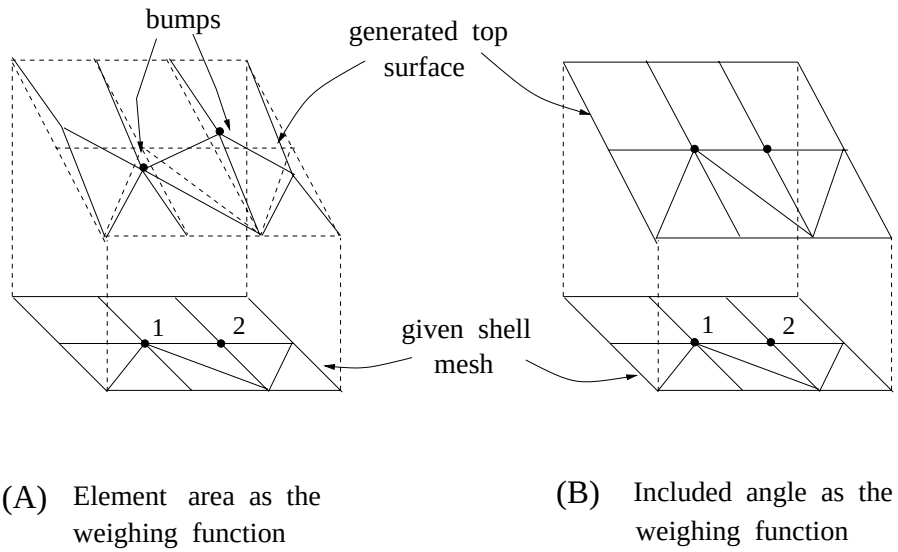

Figure 9: Effect of weighting factor A^j in Eqn. 7 while generating the top surface at the boundary of triangular and rectangular element regions. Part A : creation of "bumps" on the surface when A^j is equal to area of the surrounding element j. Part B : bumps disappear when A^j is taken to be the included angle of element j at nodes such as 1 or 2.

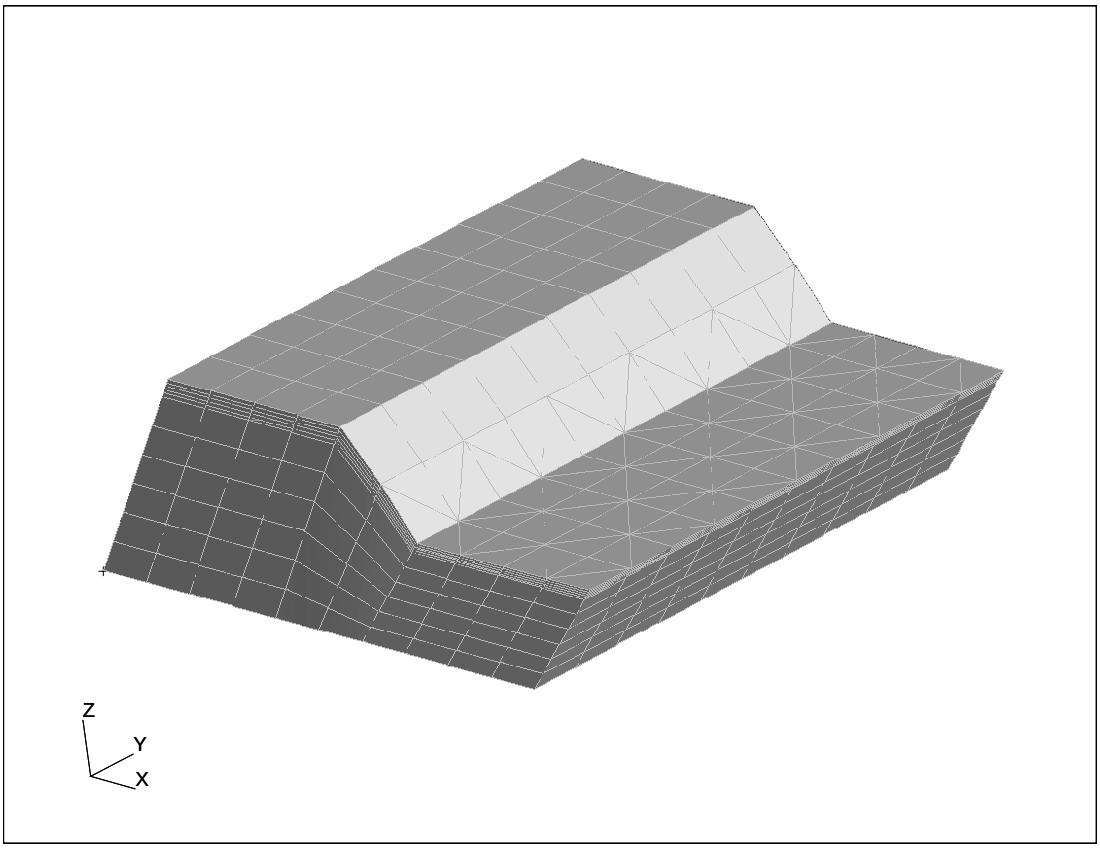

Figure 10: The flow mesh created by further subdividing the mechanical mesh of Fig. 8 along the spines. The gap and preform layers are divided into five and two sublayers respectively.

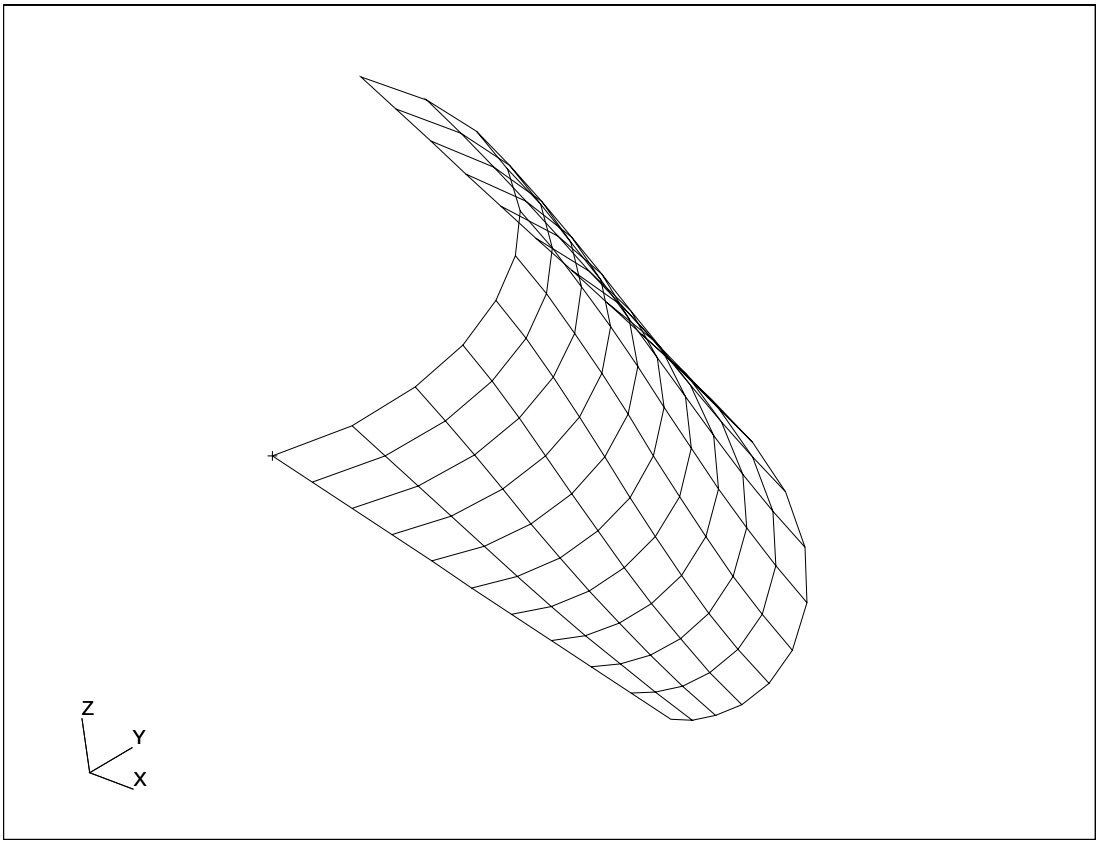

Figure 11: A representative shell mesh created out of an arbitrary surface patch. Notice that no two elements are coplanar, and consequently the normals of the elements surrounding any node are all dissimilar.

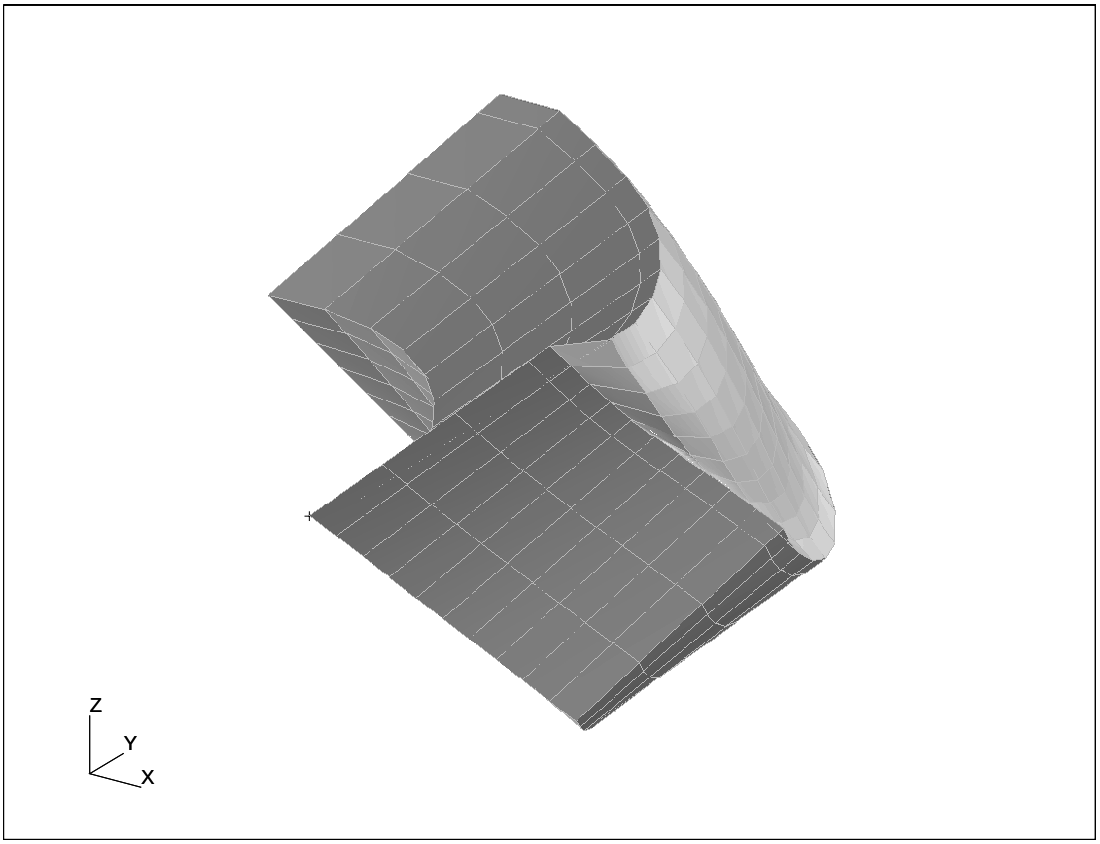

Figure 12: A mechanical mesh created from the shell mesh of Fig. 11. There are three equally thick reinforcement layers of thicknesses 0.1 units each, with the "gap" layer at the top.