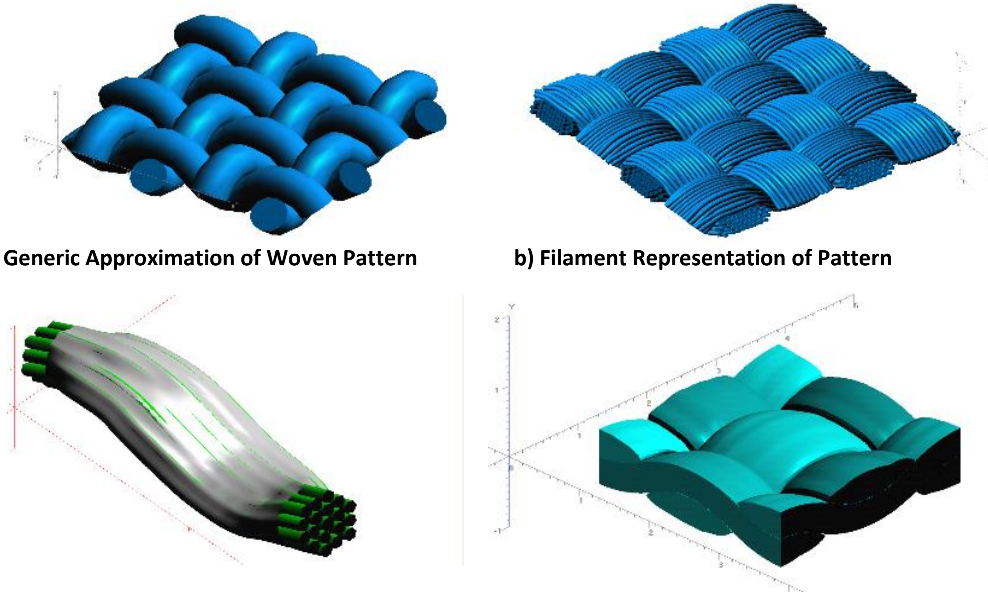
1. View Tow Data from VTMS
    a. Can view in VTMS viewer, verification that we have data using Paraview <u>Here</u>
        i. Need to import tow surface data (.stw) into working format
        ii. Develop way to translate VTMS data to Paraview data

2. Detect Penetrations of Composite Tows
    a. Method 1: Polygon Penetration (From AFRL) Here
        i. Method heavily exercised but found inadequate
    b. Method 2: NURBS
        i. Find suitable library to detect penetrations between two NURBS surfaces
        ii. Learn library interface and functions
        iii. Interface library with BetaMesh
        iv. Develop method to translate VTMS surface data to NURBS surface
        v. Execute penetration detection algorithms
        vi. Translate penetration data into format for visualization (Paraview chosen)
        Began Summer vii. Translate NURBS surface into Paraview format
        i. Convert penetration curves and surfaces to BetaMesh meshes for mesh manipulation
        Here -----------> ii. Determine which surface elements are intersected by the penetration curves (Could also be included in solving penetration regions
            1. Use KDTree algorithm to reduce number of elements checked (Keith)
            2. Develop intersection algorithm to account for all types of penetrations possible by boundary curves (Collin)
                a. Can approximate that curves are on same plane as element
                b. Will likely involve either Separating Axis Theorem or multiple "Same side of line" checks

1. Solve Penetration Regions
    a. Re-mesh intersected elements to include boundary curve points
        Began Summer i. Keep track of boundary curves
        ii. Test whether "bad elements" significantly affect results
    Here -----------> b. Remove elements with boundary curves on one tow (will need to determine which systematically)
    c. Insert mesh (within boundary curves) from second into first tow
        i. Ensures compatible mesh surface

The main purpose of the research being conducted is to resolve surface interpenetrations between composite fiber bundles (commonly referred to as tows) that have been converted into surfaces. Many woven textile finite element models use idealized tow surfaces, which means they are perfectly compatible where the surfaces interact. However, there is a need for a more realistic modeling of woven composites for analyzing the surface shapes that induce damage between fiber bundles in the matrix. To create this surface, a software called VTMS is used. It uses a simulation with woven fiber bundles to create a tightly woven composite. Then the bundles are bounded by a surface to significantly reduce the computation power needed to conduct an analysis. The surface us used to form a volume representation of the tow bundles and has material properties that mirror the strength of a fiber in the longitudinal direction. The figure below shows the process of VTMS is visual form.



a) Generic Approximation of Woven Pattern b) Filament Representation of Pattern c) Surface Approximation of Filament Bundles d) Volume Model Derived from Surfaces

Figure 1: Evolution of Weave Tow Geometry

Figure 1.d is where the main formation of surface interpenetrations occurs. Figure 2 shows these interpenetrations between two tow surfaces.

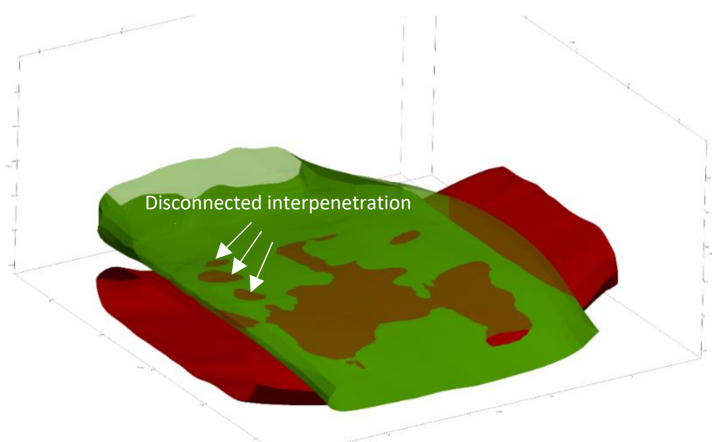


Figure 2: Tow surface interpenetrations with emphasis on disconnected regions

For traditional finite element software, the two domains cannot occupy the same region. Therefore, they need to be removed in a way that is made to be directly used with traditional finite elements. This is so it can be used the finite element software used by the research group. The first step is to correctly identify the interpenetrating elements and nodes of the surface meshes. Originally the interpenetrations were going to be identified with the default format of the surfaces, which is a discretized mesh of the surface that bounds the fiber bundles. However, none of the methods using this data format were successful. Therefore, a new surface format was chosen to represent the surfaces.

Non-Uniform Rational B-Spline Surfaces are parametric, equation-based surfaces that can be evaluated anywhere on the surface with two parametric variables. Because they are equation based, the two surfaces equations can be equated to determine where the two surfaces are equal, indicating that they are crossing through each other. This is a very helpful when determining if and where the two surfaces intersect. Developing a library that will accomplish this task is a large amount of work on its own, so a third-party library was found to calculate the intersections. However, the surfaces needed to be converted into a format that could be used with the library so a method was created to take the VTMS surface data and convert it to the input format for the library. The library then takes the surfaces and converts them into NURBS surfaces and calculates intersection curves between the two surfaces. Figure 3 shows the result of this step. These curves defined the boundary of each interpenetration region between the two surfaces.

Figure 3: Boundary curves in two viewing formats

Figure 3 also helps to show that the data from the third-party NURBS library needed to be converted into another format to be shown in Paraview, the preferred viewing software for this research. Paraview offers an extensive amount of functionality above the including viewing suite provided by the library. Both the surfaces and curves required individual translator methods so that they may be viewed in Paraview.

Once the intersection curves were returned, a post processing method was developed to ensure that all the curves are closed and unique, which ensures that they define an enclosed region of interpenetrations. This is vital because the surfaces must exit each other to form the regions that are enclosed. Regions of interpenetrations that are not enclosed cannot be solved. Once the boundary curves have been found, the NURBS surface is converted into a discrete mesh by sampling the surface and creating nodes that define a mesh. The mesh is the made by connecting all the nodes with triangular surface elements. This mesh is used as a baseline mesh for the tow surface and is the starting point for the mesh that will eventually be included in an analysis. It is important to note that this surface mesh is slightly smaller in volume than the original surface mesh used to create the NURBS surface due to how the NURBS surface is defined. A NURBS surface is not required to go through all the points that define the surface but rather is a closely fitted approximation that, by definition, is a very small error under approximation. However, it is important to sample the surface for the new mesh because that is where the boundary curves are derived from.

Once a mesh was created, a method was developed to find where the boundary curves intersect surface mesh elements. All of the intersected elements need to be divided where the boundary curve intersect any of the surface elements. Where the curve crosses the surface elements, an intersection point is added to the boundary curve. This algorithm calculates all of these points for both of the surfaces meshes. The idea is that if all the intersection points are kept, compatibility between the two surfaces can be kept. This is the main contribution of this research is the ability to maintain compatibility of the surface meshes where they intersect each other. The boundary curves now become the boundary of compatibility between the surface because it contains points that lie on both of the surface meshes. These curves are used to subdivide the elements on both surfaces. The result is shown in figure 4. However, the relative refinement of the newly subdivided elements was much higher than anticipated

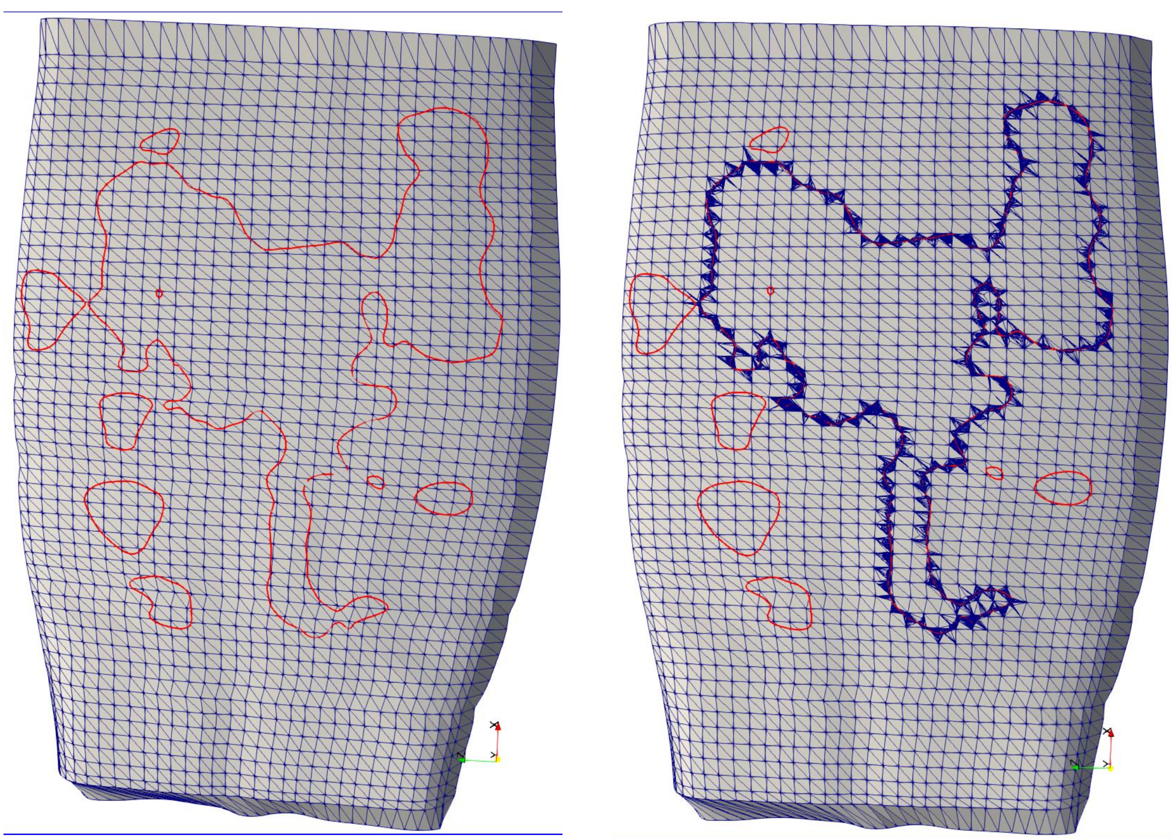


Figure 4: Original vs Subdivided surface mesh

due to the high refinement of the boundary curves. Therefore, the curves are edited to only keep the intersection points between the curve and the surface elements, effectively reducing the refinement of the boundary curves to only the essential data. The result is a much more acceptably refined surface mesh showing in figure 5.

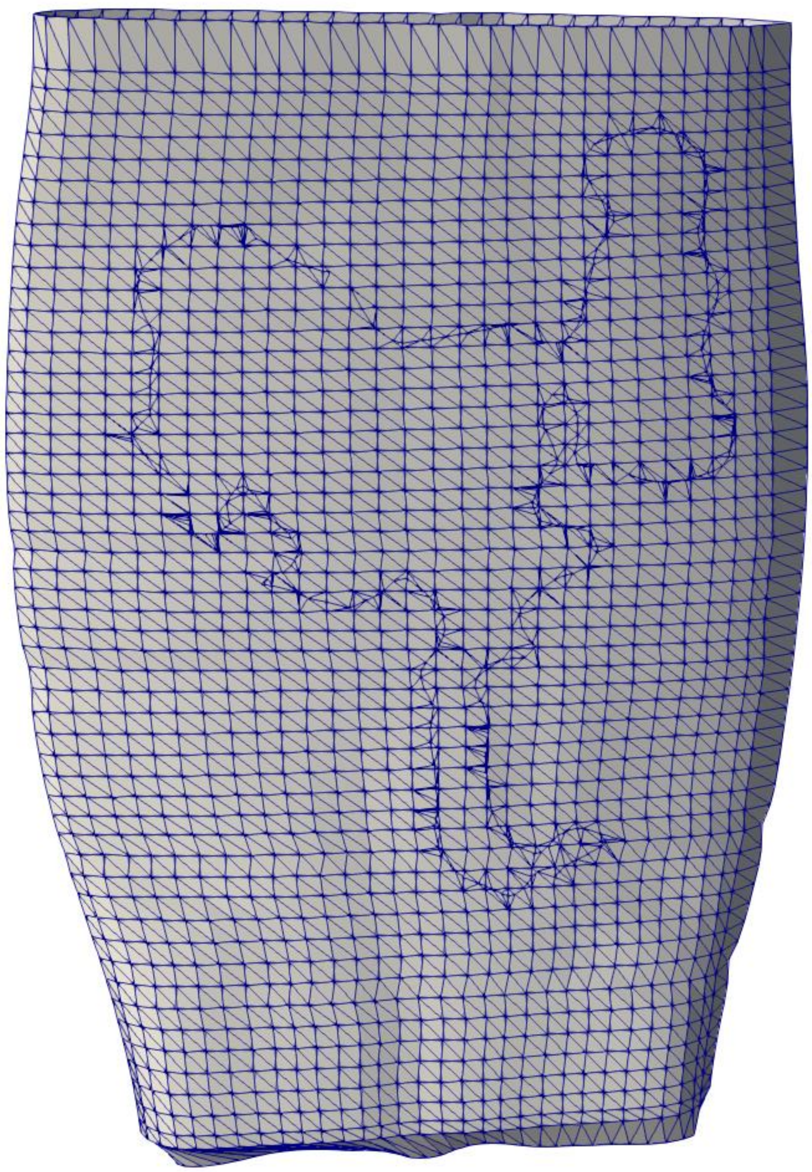


Figure5: Reduced refinement mesh of a tow surface

This is currently where the research is at. The last thing to be accomplished is to capture all the surface elements enclosed by the boundary curves and is currently under development. A method has been conceptually formulated and is currently being implemented. Once implemented, the algorithm will export the element groups so that they can be removed and a master mesh can be substituted in. The elements could also be used to perform a contact analysis and determine some form of collision surface between the two surfaces. The method to return these elements simply complies two lists that are formed by iterating over all of the elements of the tow surface and conducting edge comparisons. The elements that share an edge with the boundary curve will first be collected and then will find the other elements connected to it. By checking the connected elements as well, two lists are formed and which ever list has less elements is the list that contains all of the interpenetrating elements. It is a less than elegant solution but will accomplish this task. Once the elements are known, it is a simple matter of deciding what to do to with the elements themselves.

I have also included a picture of a more complex woven fabric that the algorithm will be applied to.

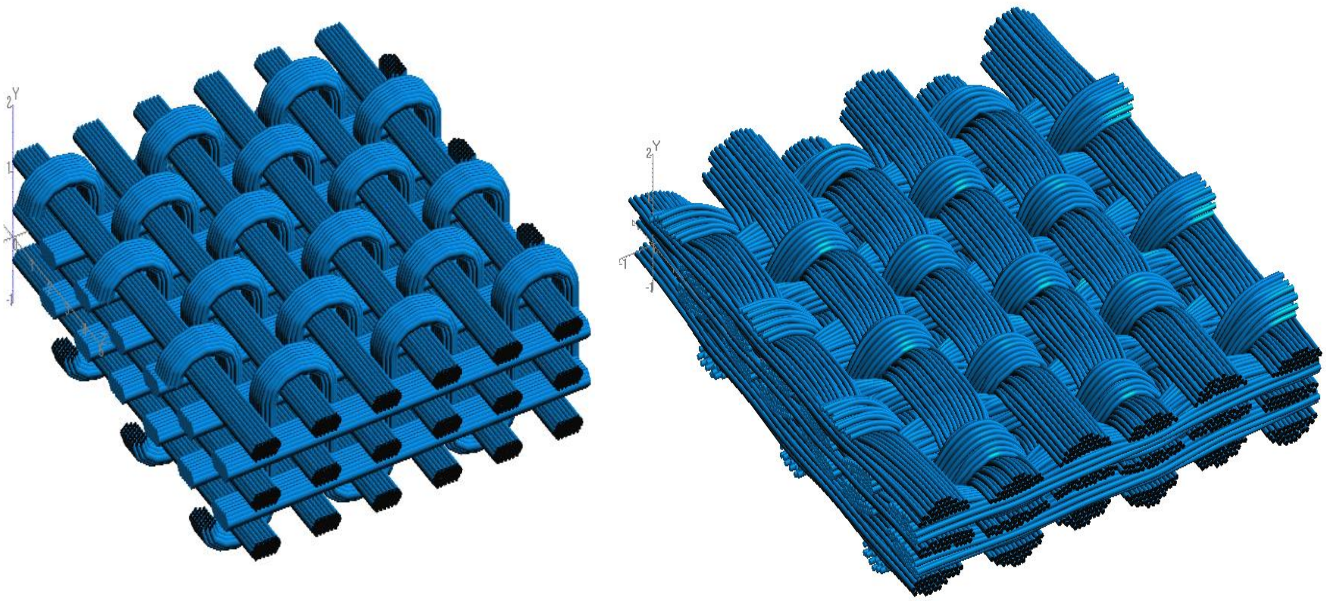


Figure 6: Preliminary-to-Post Relaxation of Complex Fabric Geometry

Tentative Schedule:

August:

- Finish development of code and write up last bit of thesis that pertains to developed methods.
- Continue editing process with professor and organize defense dates.

September:

- Finish editing process of thesis.
- Create presentation and prepare for defense.
- Development of any additional figures for defense
- Defend (planning to be at end of September)