AutoMesher for LS-DYNA® Vehicle Modelling

Ryan Alberson[1], David Stevens[2], James D. Walker[3], Tom Moore[3]

[1] Presenting Author, Protection Engineering Consultants, 14144 Trautwein Road, Austin, TX 78737, ph: 512.380.1988 ext. 307, ralberson@protection-consultants.com

[2] Protection Engineering Consultants, 14144 Trautwein Road, Austin, TX 78737

[3]Southwest Research Institute, 6220 Culebra Road, San Antonio, Texas 78238 USA

Abstract

Software has been developed to automatically mesh CAD files in support of expedient modeling of armored vehicles and similar structures. The AutoMesher software is written in Python as well as LSTC's Script Command Language (SCL). The SCL syntax is similar to C programming but runs as a script within LSTC's LS-PrePost® (LSPP) software application. A Python module is used as the interface and a wrapper for LSPP. By leveraging the functions in LSPP through the SCL, nine different algorithms were written to mesh I beams, T beams, angles, rods, plates, tubes, and surface-meshed formed shapes. Logic is used in these algorithms to identify the shape characteristics needed to define an equivalent FEA mesh of the CAD geometry. These algorithms are the heart of the AutoMesher and can be used to generate more intelligent meshing solutions. The algorithms and software are described in this paper.

The AutoMesher software was developed by Protection Engineering Consultants (PEC) in support of the Defense Advanced Research Projects Agency (DARPA) Adaptive Vehicle Make (AVM) program, under subcontract to Southwest Research Institute (SwRI). AVM is an ambitious program to reduce the time required for the design, development, and production of complex defense cyber-mechanical systems, such as military ground vehicles, by a factor of five.

Introduction

The purpose of the AutoMesher software is to generate a Lagrangian Finite Element (FE) mesh from 3D Computer Aided Drawings (CAD). The AutoMesher opens a batch of STEP files in LS-PrePost (LSPP) and executes shape specific algorithms on each CAD file to modify the geometry and prepare it for meshing operations. The generated mesh of each part is then included in a master input file that groups all parts into one LS-DYNA formatted input deck.

With an automated meshing process, engineers will be able to focus energy on the other challenges that must be overcome to produce accurate and precise simulations. As an example, the sample vehicle hull shown in Figure 1 required less than 5 minutes to generate a mesh with corresponding *PART and *SECTION cards. Performing the same work manually may take an engineer several days, if not weeks, depending on the complexity and number of parts.

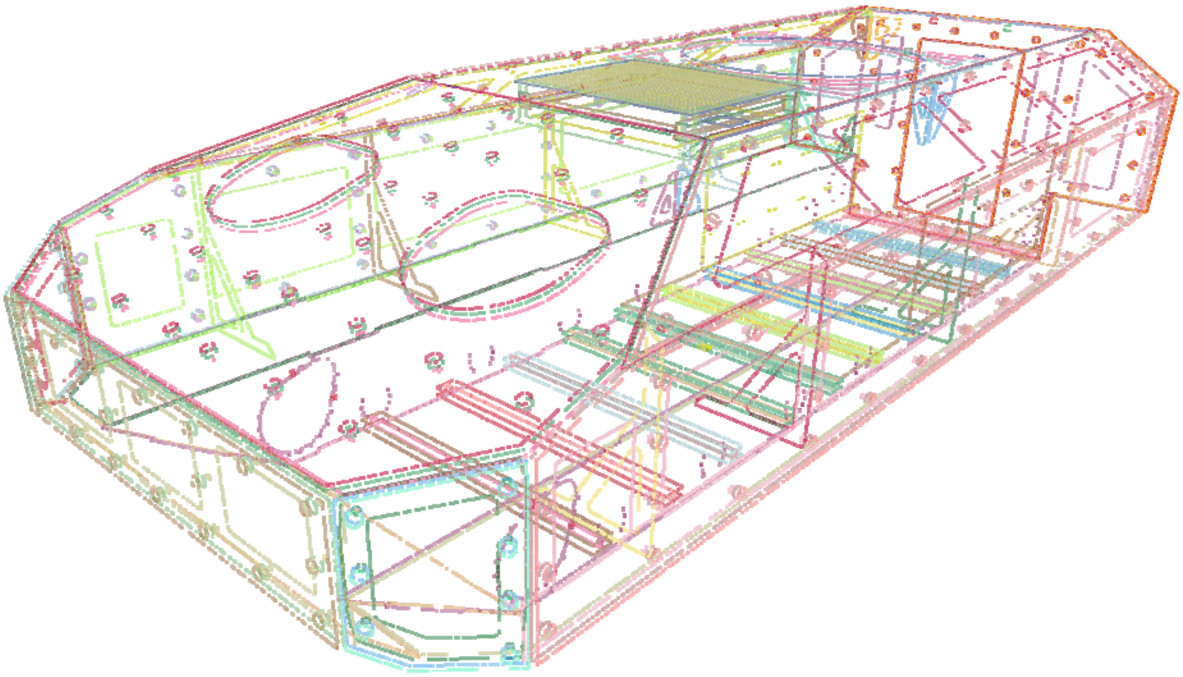


Figure 1. Example Vehicle Hull

Figure 2 shows the time required to mesh a variety of parts. Notice that most parts take less than 10 seconds to mesh if the part has less than 50,000 elements. A linear increase in meshing time can be observed in parts that have more than 5,000 elements. This is due to the efficiencies within the LSPP meshing algorithm.

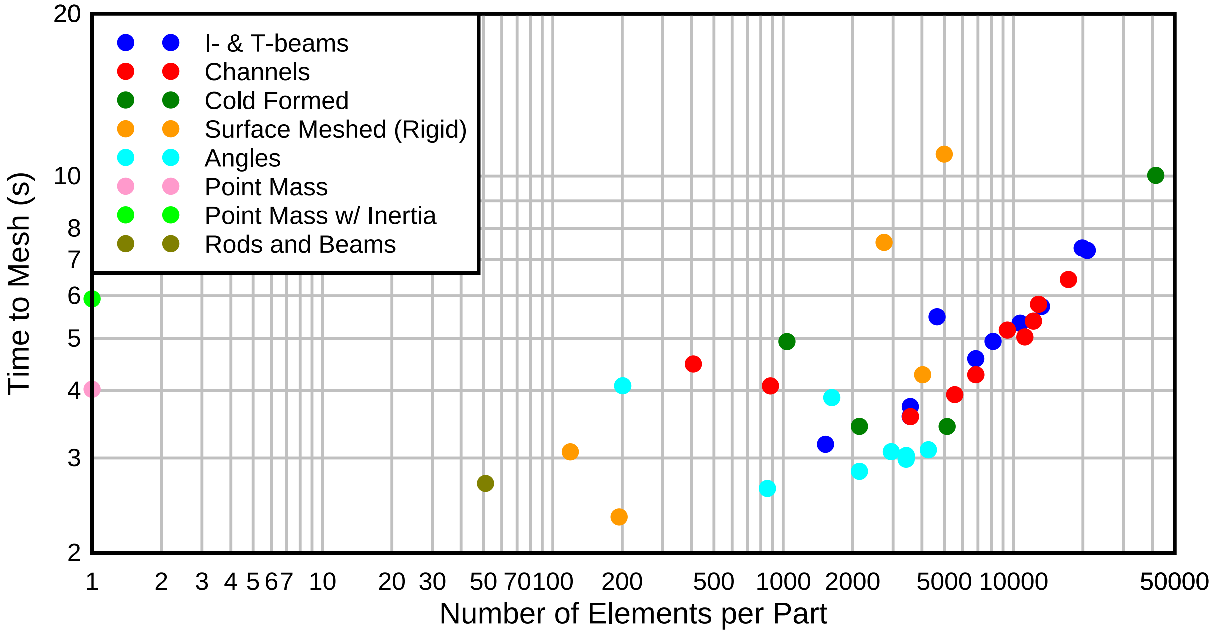


Figure 2. Time to Mesh

Development

PEC developed an algorithm that automatically generates shell and beam element meshes from arbitrary 3D CAD geometry. To accomplish this, the CAD geometry was divided into several categories based on the shape of the geometry. These shape categories include I- & T-beams, channels, angles, cold formed shapes, HSS beams, rods, and complex molded parts. These shape categories can be specified by the user or the AutoMesher will identify the correct shape category for each part. However, the computational time to identify the shape category can increase the processing time required for each part by a factor of 2 to 8 times. Therefore, it is recommended that the shape be identified prior to running the AutoMesher.

In addition to the required CAD geometry for mesh generation, there are several optional parameters that can be specified to tune the generated mesh. These parameters are used to specify element size, minimum and maximum hole diameters, toggles for hole processing algorithms, and mesh quality controls. If material information is supplied with the CAD file, the AutoMesher will use that information to check the time-step of the generated elements and use these criteria to judge mesh quality.

The AutoMesher leverages the capabilities found in LSPP through the use of the Script Command Language (SCL). SCL scripts are executed by LSPP similar to the way macros are executed, but SCL scripts are written in a C-based syntax, which are compiled by LSPP at runtime. They are capable of more advanced logic operations compared to conventional LSPP macros. In the AutoMesher, LSPP and the SCL script are wrapped by Python scripts to provide better system level control and exception handling.

Capabilities

For each shape category, a separate algorithm was developed to interrogate the CAD geometry and identify the common features of the shape category.

I-beams and T-beams

I beams (W shapes) and T-beams (WT shapes) are similar and are more complicated than other shapes due to their “built up” nature. That is, these shapes have both a web and flange section that must be handled separately but in conjunction with each other. In addition to identifying which surfaces represent the flange or web, any discontinuities (cutouts, bolt holes, bends, etc.) in the surface must be tracked and incorporated into the mesh as well (Figure 3). The flange(s) are joined to the web by a merge operation which requires precise node alignment between each surface. Also, during the meshing process, the thickness of the flange and web sections are parsed from the CAD file and *SECTION_SHELL cards are automatically generated. As shown in Figure 4, the web is meshed on the centerline while the flange is meshed on the interior face to better represent the shape with shell elements. The *SECTION_SHELL cards are used to define the thickness and the location of the reference surface for the shell thickness.

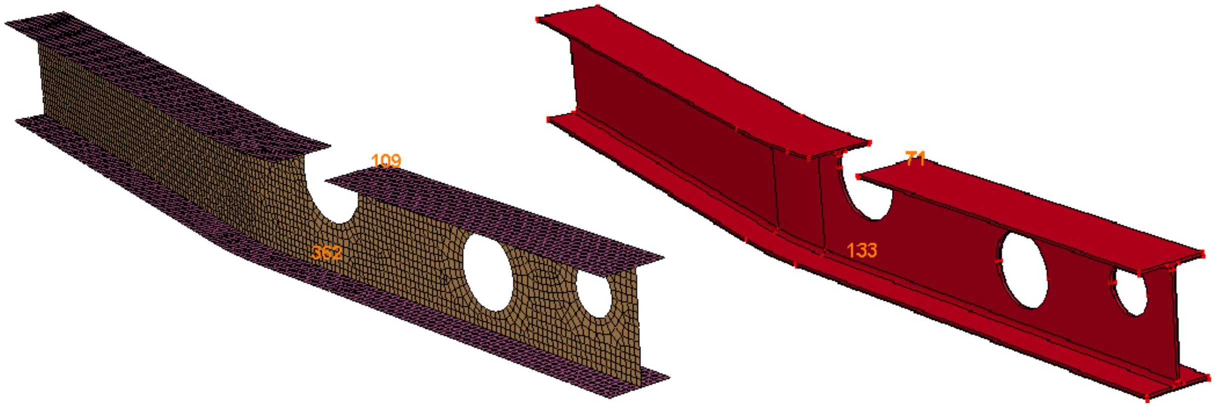


Figure 3. Complex I-beam used for Validation

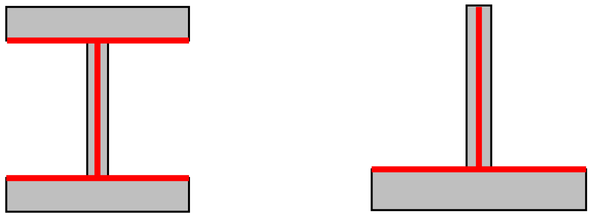
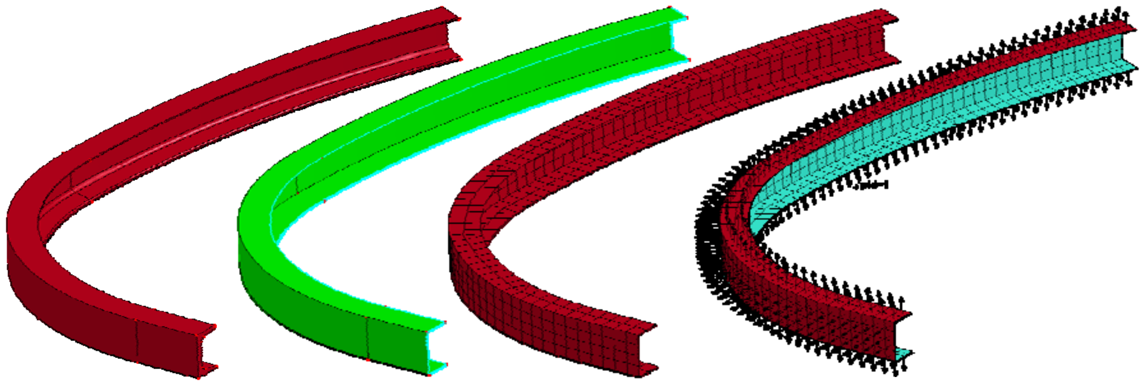


Figure 4. I-beam and T-beam Cross Section with Shell Locations shown in Red

Channels

Channels (C-Shapes) are similar to I- and T-beams in that they also have web and flange surfaces. For many structural channels/C-shapes, the flange is also tapered, which can make it hard to identify the actual mid-plane of the flange. Therefore, the algorithm creates surfaces along the outside of the C-shape and meshes those surfaces directly. The corresponding section cards for the flange and web assign the material to the inside of the shell. When the flange thickness is tapered, the average thickness of the flange is used as a uniform thickness. In Figure 5, the algorithmic process is shown for a bent channel, starting with the CAD Geometry and progressing to a representative shell mesh with section cards defined.



Original Solid

Surface Representation

Meshed Surfaces

Consistent Normals

Figure 5. CAD to Mesh Progression

Cold Formed, HSS, Sheet Metal

This shape category is very common in automotive applications and accommodates any continuous part where all sections of the part are of uniform thickness and connected tangentially to each other. This algorithm can be applied to cold formed steel, Hollow Structural Section (HSS), plate steel, and sheet metal. Parts that meet this criteria are meshed along the centerline and the *SECTION_SHELL cards are automatically defined based on the interrogations of the CAD geometry. Figure 6 shows an example of a uniform thickness part with the CAD geometry and the shell representation.

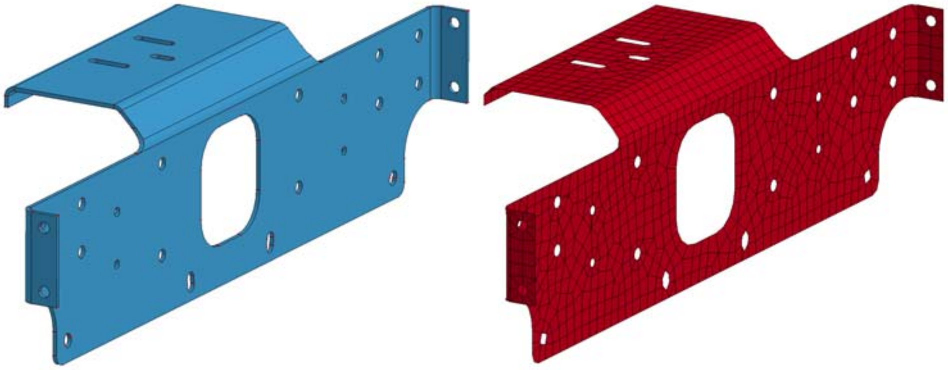


Figure 6. Uniform Thickness, Formed Part with CAD Geometry (left) and Shell Representation (right)

Surface Meshed

Surface meshed parts are “shrink-wrapped” to the CAD geometry. They are useful for representing rigid parts that contribute to the mass and inertia of the system but their deformation isn’t important to the overall simulation. For vehicle models this is often true of parts that belong to the drivetrain. The mass and inertia are important for analyzing the total response of the system, but deformation in the parts is not needed. The mesh is only used for volumetric representation (for contact purposes) and the inertial properties of the part are assigned to a point mass at the center of gravity (CG) of the CAD geometry.

This shape category is also useful for molded parts that should eventually be meshed as solid parts but can be meshed as rigid parts during the initial debugging of the simulation. Figure 7 shows a surface meshed assembly.

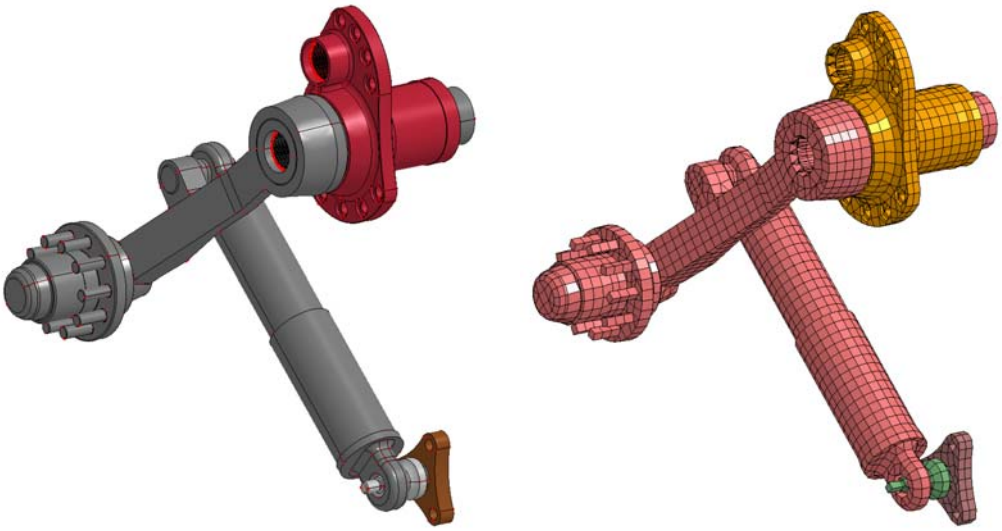


Figure 7. Surface Mesh (Rigid Parts) with CAD Geometry (left) and the Mesh Representation (right)

Angles and Square Tubes

Angles and square tubes are very similar to channels in that these shapes have 90-degree angles at the corners; however, angles and square tubes have a uniform thickness (as opposed to unique flange and web thicknesses). These shapes also differ from cold-formed and HSS parts because of the 90-degree angle that joins the different sections of the shape. The most common use for this shape is in connections where very short angles are used to bolt adjoining parts. These often lead to very small angles with several bolt holes, which can be challenging to represent well with a mesh. This algorithm has extra logic built into it for handling these challenging scenarios. Figure 8 shows the CAD geometry (left), shell representation (middle), and shell representation with thickness displayed and overlaid with the CAD geometry (right). Notice the good correlation between the shell representation and the CAD geometry; the only discrepcancies exist at the fillets near the bend and at the tapered ends.

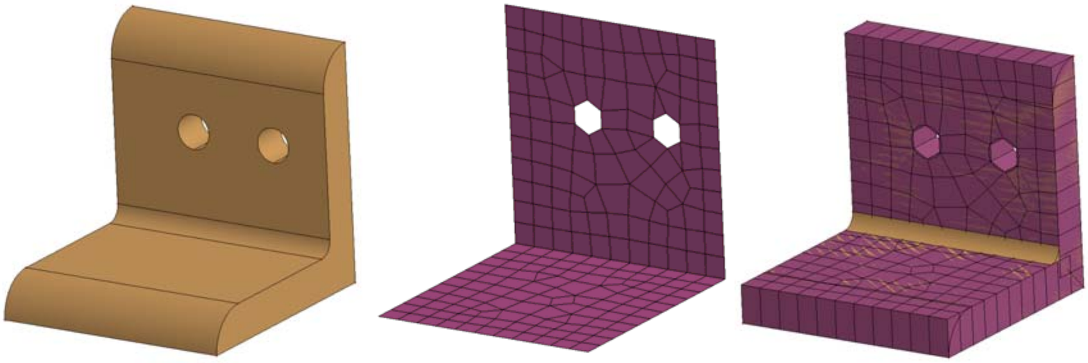


Figure 8. Angle Connection CAD geometry (right), shell presentation (middle), and overlaid mesh and CAD geometry (right)

Rods and Other Beam Elements

The AutoMesher can also handle complex parts that can be represented as beam elements in the simulation. Currently, round, square, and hexagonal rods can be identified from CAD geometry and the corresponding *SECTION_BEAM cards are created automatically using the available beam cross sections available in the *INTEGRATION_BEAM card. The beam elements are created along the axis of the part and have the ability to "heal" over small cutouts or notches in the CAD geometry. As shown in Figure 9, the complex CAD geometry on the left was modeled and the representative beam elements are shown in purple. The "Beam Prism" option is turned on for the right image to show the correct section card applied to the beam elements. Also notice that the cutouts at the end of the section did not cause mesh generation issues

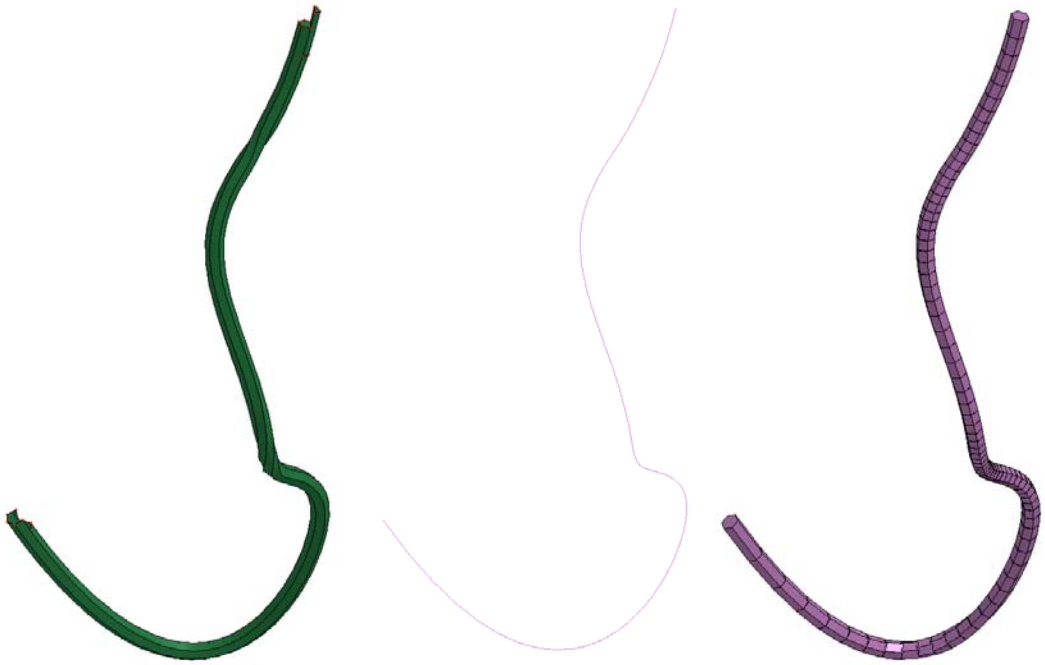


Figure 9. Complex Hexagonal Rod with CAD Geometry (left), Beam Bepresentation (middle), and Beams with Thickness Displayed (right)

Point Mass (With Inertia)

Some parts do not need a mesh to represent the volume for contact purposes but the mass and inertial properties must be included. If the material density or mass of the part is supplied by the user, the AutoMesher will create an *ELEMENT_MASS or *ELEMENT_INERTIA to represent the part.

Validation

Automatic mesh generation has the potential to not only save time but improve mesh quality with strong element quality checks that are executed systematically. To examine the mesh properties, several checks were performed to assess mesh accuracy, precision, and quality. A suite of parts were assembled to use as unit tests for each of the shape categories as shown in Figure 10. More shapes are added to the unit tests as features are added.

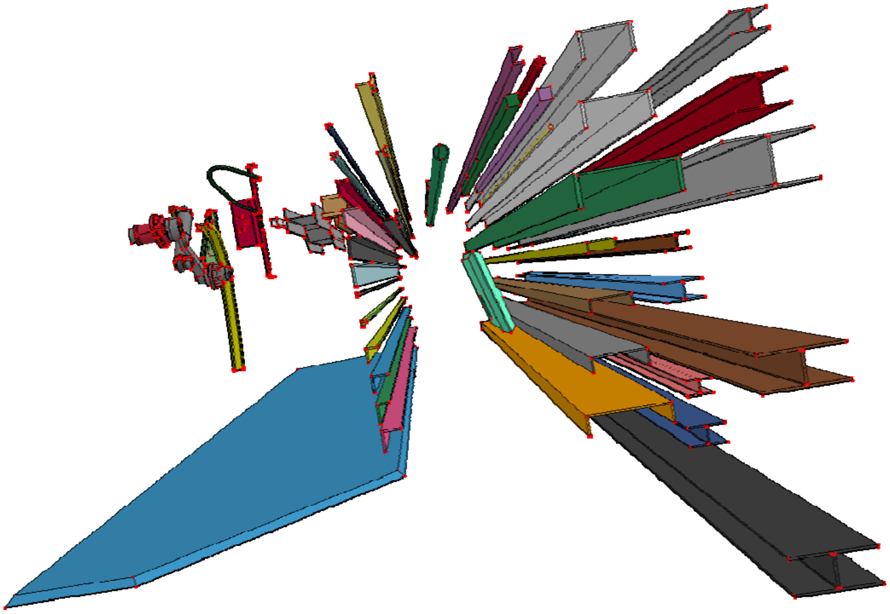


Figure 10. Unit Testing Parts

Accuracy

To verify mesh accuracy, the AutoMesher checks the volumetric ratio between the CAD geometry and the meshed part. A perfect match will yield a ratio of 1.0. This check is run on every part and is used as an initial filter to judge whether or not a part should be re-meshed. During the Results from the unit tests are displayed in Figure 11. This plot reveals that larger parts will yield better representations, which is due to fewer features where the mesh density may not represent the CAD geometry ideally. However, the results are still very good, and the error is in the noise.

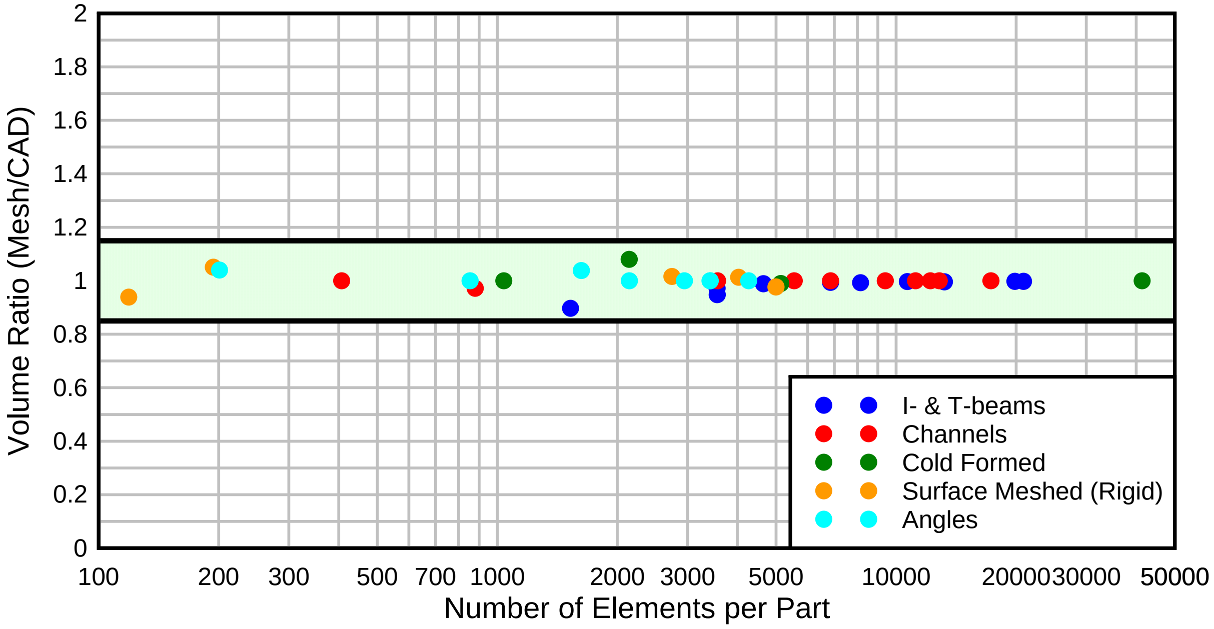


Figure 11. Volume Ratio

The threshold value can be specified by the user, but care has to be used to ensure that the value is not too large or small. Larger thresholds result in a higher chance that the mesh will not represent the CAD geometry, and lower values may be too strict, resulting in unintentional discarding of adequate mesh representations. A value of +/-15% seems to be a reasonable value for most parts and yields good results.

Figure 10 shows the current set of parts that are used for unit testing the AutoMesher. There are several parts for each shape category with several different sizes therein. Most of these parts were part of the original project requirements, but the shapes were fairly basic. More complex parts were generated to ensure robust operation for modified shapes. Some of the more complex modifications that were made to I- and T-beams are shown in Figure 13.

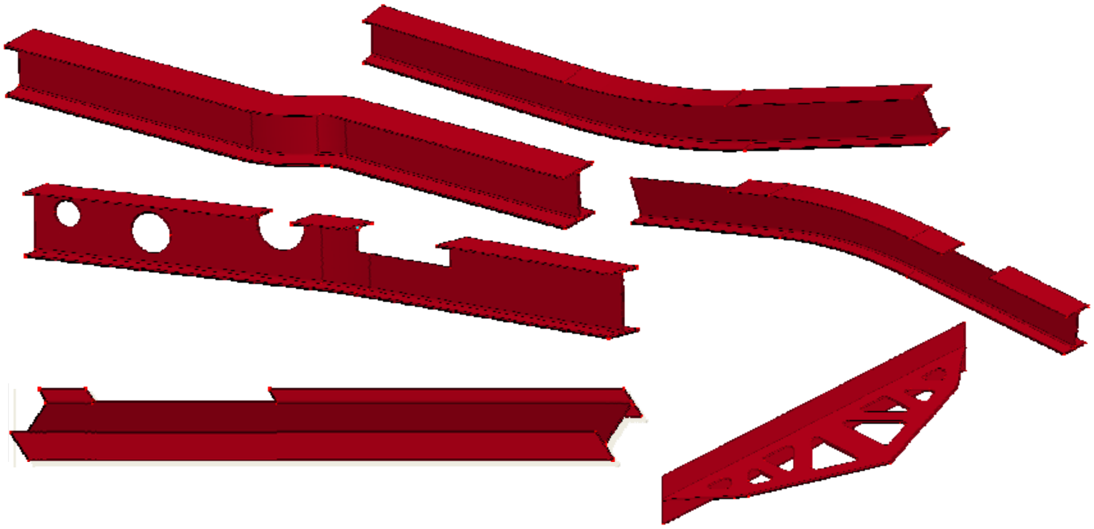


Figure 12. I- and T-Beam Parts with Cutouts in the Web and Flange

Quality

As with any automated task, quality control is of concern. The AutoMesher is able to run the same quality checks that LSPP can perform for shell elements: minimum side length, maximum side length, aspect ratio, warpage, minimum angle, maximum angle, taper, skew, jacobian, characteristic length, element area, feature angle, and timestep. If material properties have been supplied for the part, the timestep can be computed, and it is the ultimate quality check prior to simulation execution. The calculated timesteps for each part in the unit test are shown in Figure 14. With a 15-mm mesh, all parts are more than an order of magnitude above the desired threshold. The Jacobian is also a good measure of element quality and the minimum jacobian value for each part in the unit test is shown in Figure 15. If any given element does not meet these requirements, the part will be remeshed either locally or as a whole to resolve the mesh quality issues.

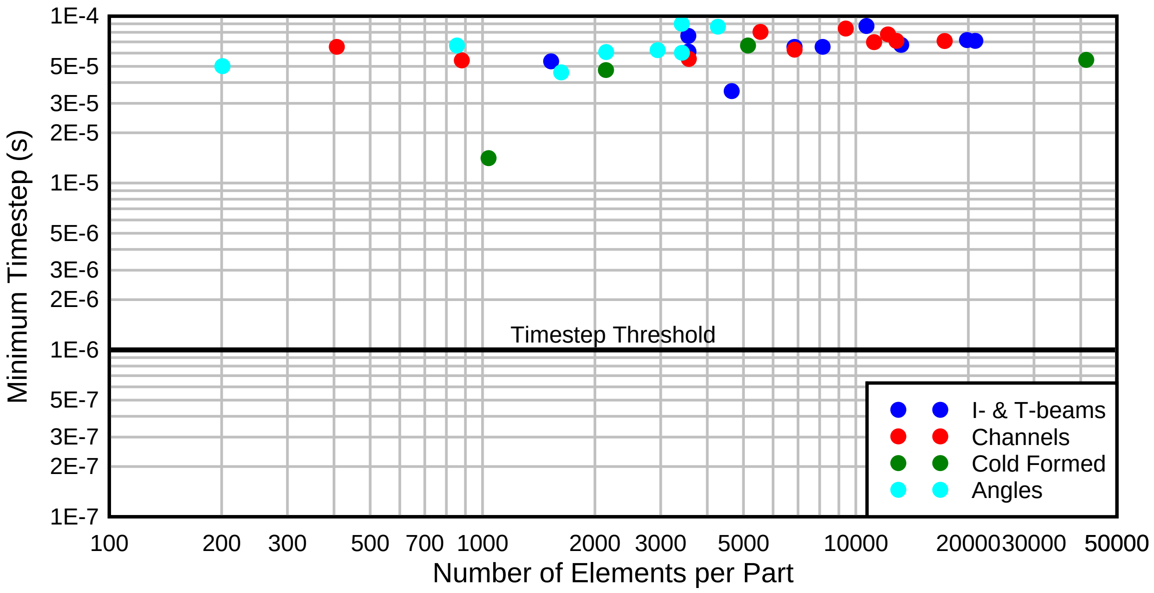


Figure 13. Minimum Time-step of Automeshed Parts

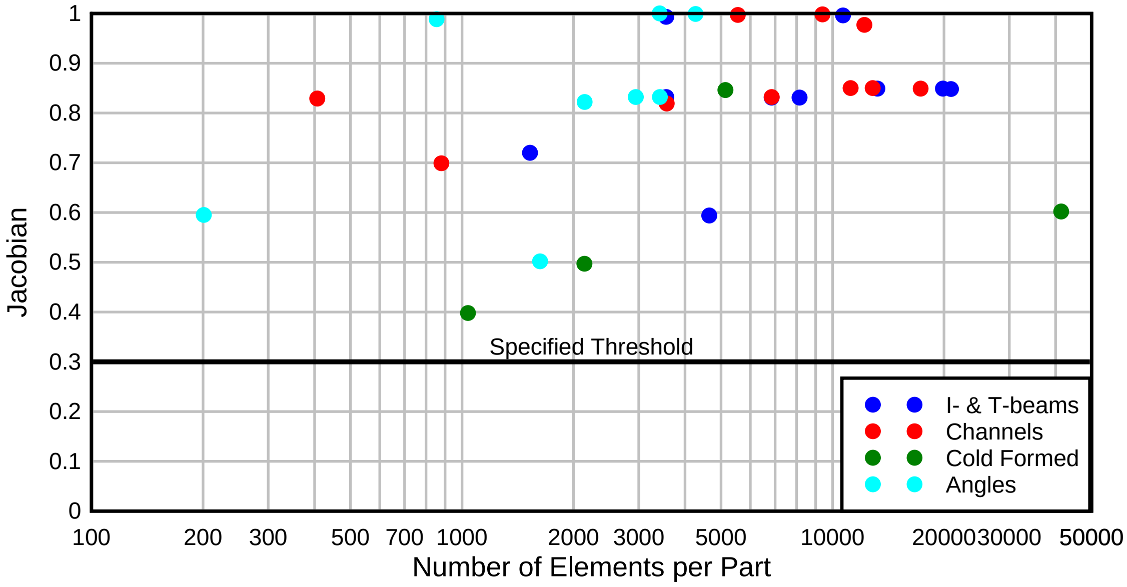


Figure 14. Jacobians of Automeshed Parts

Conclusions

The AutoMesher has a broad range of capabilities applicable to industries where meshing is a significant portion of the simulation development process. In industries where LS-DYNA is used as a design tool and parts are meshed several times across multiple iterations, the AutoMesher can provide engineers with a fast, accurate, and precise method to reproduce high-quality meshes. The time savings from using the AutoMesher will allow engineers to focus on solving the problem, rather than meshing and re-meshing parts.

Distribution Statement "A" (Approved for Public Release, Distribution Unlimited).

Disclaimer: The views expressed are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S. Government.