Isogeometric analysis using the *IGA_INCLUDE_BEZIER keyword in LS-DYNA

Christopher Whetten[2], Matthew Sederberg[1], Michael Scott[2],

[1]Coreform LLC
[2]Coreform LLC, Brigham Young University

1 Introduction

In contrast to the laborious and error-prone process of translating computer-aided design (CAD) into computer-aided engineering (CAE) models, isogeometric analysis (IGA) performs the finite element analysis (FEA) simulation directly on CAD geometry, using smooth spline basis functions. LS-DYNA is a leader in the industrial adoption of IGA, and has recently made a significant enhancement to broaden the possible use of IGA within LS-DYNA.

`*IGA_INCLUDE_BEZIER` is a new keyword that was recently implemented in LS-DYNA to enable the use of unstructured spline models such as U-splines and T-splines. This is significant because it allows for more complex models to be used for IGA in LS-DYNA.

The `*IGA_INCLUDE_BEZIER` definition is open source, and Coreform has also implemented this in its Coreform Analyze IGA solver and its Coreform Process preprocessor. This provides a tight connection between Coreform and LS-DYNA, and enables several unique workflows. For instance, LS-DYNA linear meshes can be fully or partially converted to smooth U-spline models in Coreform Process, then exported back to DYNA using the `*IGA_INCLUDE_BEZIER` keyword as unstructured splines, to take advantage of the LS-DYNA IGA capabilities.

A brief summary of IGA and its benefits over traditional FEA will be given. The basic concepts of Bézier extraction and the new `*IGA_INCLUDE_BEZIER` file format will also be presented. Examples of the workflow described above for converting existing CAE models will be highlighted, including simulation results. Motivating examples of complex geometry that can be brought into LS-DYNA through this workflow will also be shown. The full description of `*IGA_INCLUDE_BEZIER` will be included as an appendix.

The new `*IGA_INCLUDE_BEZIER` keyword will accelerate the adoption of IGA and bring the benefits of higher-order geometry to the simulation and design industry. The integration with the Coreform toolset and, in particular, U-splines can provide a practical and advantageous use for IGA in LS-DYNA simulations, especially in problems where contact, curved geometry, or dynamics are required.

2 Smooth spline based simulation (IGA)

Isogeometric analysis is an approach to FEA that performs simulation directly on smooth splines; the same fundamental basis used by almost all current CAD software. IGA was originally introduced in 2005 by Hughes et. al as an alternative method to traditional FEA. Conventional analysis techniques require a time-consuming process of converting smooth CAD geometry to linear faceted representations required by CAE software. As the complexity of the simulation increases, this conversion process becomes increasingly time consuming and requires additional expertise to be done correctly. Because IGA can operate on splines, the same fundamental basis of geometry as CAD, there is potential to reduce or eliminate this lengthy conversion process.

IGA has also been shown to be capable of solving problems that are difficult or time consuming to solve using linear finite elements. Large deformation simulations that have required explicit approaches with small time steps and many elements can be solved with implicit methods with significantly larger timesteps and fewer elements. Contact problems that are either highly dynamic or involve complex curved geometry lend themselves to a higher-order spline representation that is more capable of representing contact surfaces as they meet and deform.

New spline technologies have been developed to increase the robustness of IGA and allow for a wider application of analysis. T-Splines, an unstructured spline technology adopted by Autodesk, incorporates the use of T-Junctions, significantly increasing the complexity of topology and geometry available for smooth spline simulation. This also introduced a limited amount of local *h*- refinement for smooth splines; an important characteristic for analysis. More recently U-splines have been developed to eliminate certain restrictions inherent in T-Splines, and allow for the construction of meshes with mixed element types and varying degree and continuity. This enables increasingly complex geometry to be used in IGA as well as coarser meshes that produce accurate results.

U-splines are a novel development in the IGA community as they allow local *h-, p-, and k-* refinement. This freedom allows designers and analysts to tailor a basis specifically to the needs of their geometry or simulation. As will be shown in example 4.2.3, one of the most significant benefits of this functionality is the ability to create higher-order meshes with properties that produce larger explicit timesteps than a linear representation of the same mesh. Another significant property of U-Splines is the ability to translate traditional linear finite element meshes and perform degree and continuity elevation. These meshes can include extraordinary points (vertices with valence other than 4), triangle elements, exotic T-Junction configurations, and other mesh elements that would otherwise be impossible to represent as a single geometric object with higher order degree *and* continuity. The U-spline technology enables IGA to fit into pre-existing workflows where higher order geometry is desired.

3 *IGA_INCLUDE_BEZIER

LSTC had previously implemented a format called BEXT (for Bezier Extraction) that allowed for certain kinds of splines to be used in IGA. Recently, LSTC has developed a new format which expands the types of splines that can be used in LS-DYNA. The new format `*IGA_INCLUDE_BEZIER` was developed through consultation with academic and industry leaders in unstructured spline technologies, including Coreform LLC, and enables the use of T-splines, U-splines, and potentially other types of unstructured splines in LS-DYNA to represent complex, watertight models.

3.1 The Bezier Extraction Operator

Spline functions (B-splines, NURBS, T-splines, U-splines, etc.) can be constructed by multiplying a set of Bernstein polynomials with a Bézier extraction operator. The Bézier extraction operator is a matrix of coefficients that determines the contribution of each Bernstein polynomial to a given smooth spline basis function. In FEA, this construction is generally localized to a specific element, creating a single extraction operator per element. The A-th basis function (N) with support on element e is defined as

$$N_A^e = \sum_j C_{Aj}^e B_j^e \qquad (1)$$

where B_j^e is the j-th Berstein basis function defined on element e, and C_{Aj}^e is the j-th Bezier extraction coefficient for the function.

By carefully choosing extraction coefficients, one can construct splines that span multiple elements with parametrically smooth transitions between elements and the properties necessary for analysis. Traditional untrimmed B-splines and NURBS restrict the construction of these splines to rectangular topologies and require all elements of the model to be the same degree.

U-splines were specifically invented by Coreform to represent arbitrary topologies in an analysis-suitable fashion. U-splines reduce the restrictions inherent in NURBS, and can support more complex topological features such as T-junctions (or “hanging nodes”), extraordinary points, and mixed element types (i.e., both triangle and quad elements). See Table 1 for more details. An example of a U-spline model is shown in Figure 1.

Property	U-splines	BREPs	T-Splines	FEA
NURBS compatibility	**Yes**	Yes	Yes	No
Exact Geometry	**Yes**	Yes	Yes	No
Smooth basis	**Yes**	Yes	Yes	No
Arbitrary degree	**Yes**	Yes	No	Yes
Local exact h, p, k adaptivity	**Yes**	No	No	No
Linear independence	**Yes**	Yes	No	Yes
Optimal approximation	**Yes**	No	No	Yes
Volumes	**Yes**	No	No	Yes

Table 1: A comparison of the analysis-suitability of U-splines, BREPs, T-splines (patent owned by Autodesk), and FEA meshes. While many of these conjectures about U-splines are still being proven, U-splines have been constructed explicitly to address significant limitations in these older technologies.

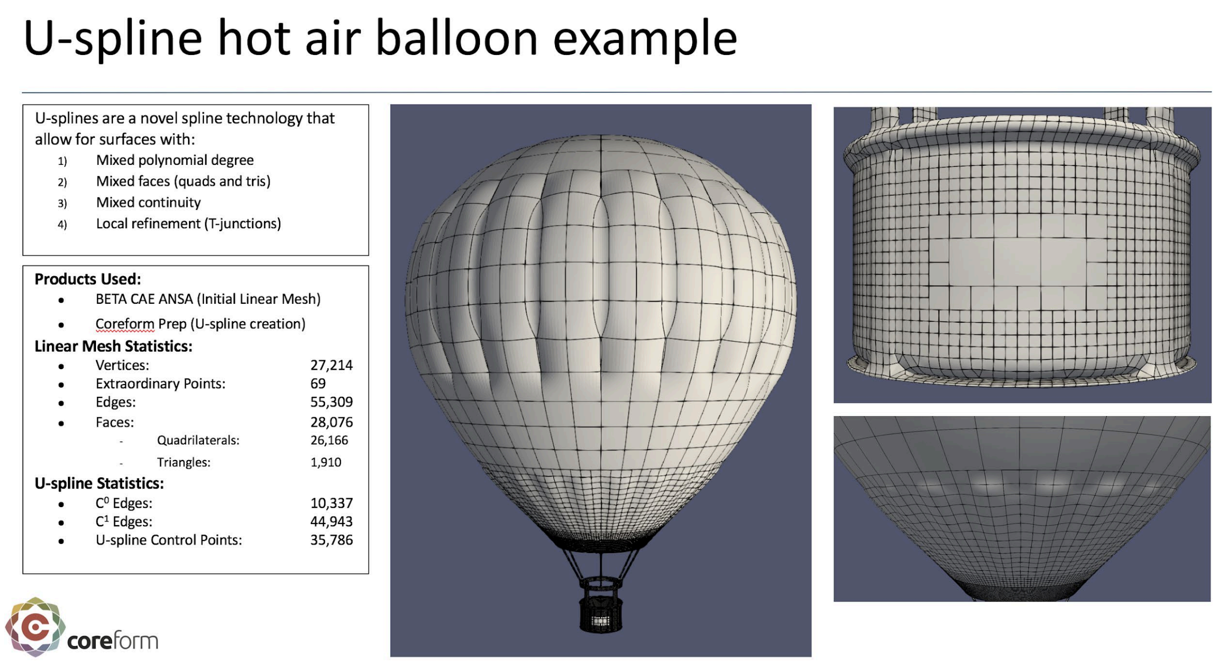

Fig.1: Example of a U-spline model, showcasing the flexible U-spline smoothness and refinement properties critical for success in simulations.

Fundamentally, all spline types, including NURBS and U-splines, can be constructed from the form of equation 1. The number of Bernstein polynomials (indicated by the index j) is determined uniquely by a description of the element type (i.e. cube, simplex, etc.) and a vector defining the degree of the basis in each parametric direction. The Bezier extraction operator then defines the number of functions with support on that element and the Bernstein coefficients for each of those functions. Regardless of the complexity of the spline, this information allows LS-DYNA to create the basis and associated connectivity required for analysis.

3.2 Description of *IGA_INCLUDE_BEZIER

The new **`*IGA_INCLUDE_BEZIER`** keyword adopts a general approach for input and storage of geometric and extraction data that allows for many kinds of splines to be used in LS-DYNA. A “patch” as used in this keyword denotes a single connected spline object. Note that although multiple patches can be present in a single file, each must share the same dimension and part ID. The general format of the input for a single patch is as follows:

1. Patch data: Basic data pertaining to the entire patch. This includes the patch ID, weight flag, and the number of nodes, elements, and unique coefficient vectors.
2. Geometry: Nodal positions and weighting.
3. Elements: Element descriptions such as type (cube, simplex, etc.), degrees, and pointers to the rows of the extraction operator allowing for a reconstruction of the basis.
4. Coefficient vectors: Reusable storage for each unique element extraction row in the patch model using either dense or sparse representations.

This new format is designed to maximize efficiency and storage in the input. All element types are sorted into blocks of similar elements to compress the data. Coefficient vectors are also sorted, and can be represented by either a dense or sparse representation, depending on which compresses the data more for that specific extraction row.

LSTC advises that **`*IGA_INCLUDE_BEZIER`** is currently scheduled to be publicly available in Revision 12 of the LS-DYNA Keyword Manual. Though no exact date for the release of this version has been set, it is anticipated that this keyword will be available in a beta version during the summer of 2019. A description of this keyword is included in the appendix.

4 Workflow and examples

To help analysts take advantage of unstructured splines and provide a pipeline for users to bring these new spline types into LS-DYNA, Coreform LLC is developing Coreform Process as a preprocessor for unstructured smooth splines. Coreform has implemented an integration with LS-DYNA that gives users the ability to both read in LS-DYNA meshes and write out smooth splines using the **`*IGA_INCLUDE_BEZIER`** keyword. This tool allows IGA to fit into existing workflows by taking meshes already created and converting them to unstructured higher-order spline representations. These higher-order U-Spline meshes can then be modified to the needs of the simulation and fed back to LS-DYNA for analysis or used in Coreform’s dedicated spline solver, Coreform Analyze. Tools are also available to generate models natively in Coreform Process for problems that lend themselves well to IGA constructions. Both Coreform Process and Coreform Analyze are prerelease software not yet publicly available; interested parties may request access to these tools by contacting Coreform LLC.

It should be noted that although U-Splines theory is being extended to define unstructured solid meshes, this technology is still under development. The examples below focus on surface geometry using shell formulations and structured solid meshes.

4.1 Using U-splines to increase time step size in LS-DYNA via *IGA_INCLUDE_BEZIER

A promising potential benefit of U-spline technology is the ability to modify mesh properties to produce superior explicit dynamic simulation times. When running assembly simulations using explicit dynamic formulations, the stable time step size is controlled by the maximum discrete frequency in the model. This has previously limited the use of IGA, as time steps for higher-order meshes have been observed to be significantly smaller than those of linear elements, leading to inferior performance compared to FEA. U-splines provide a unique opportunity to locally modify degree and continuity of elements to increase the overall timestep of the mesh. This technique involves reducing the degree locally for elements on the boundary and nearby other features that require only positional continuity. Experimental results have shown that timesteps of meshes constructed this way can be up to 60% larger than linear elements and nearly 75% larger than uniform degree splines. This technology has the potential to enable analysts to take advantage of higher-order smoothness without sacrificing solution speed.

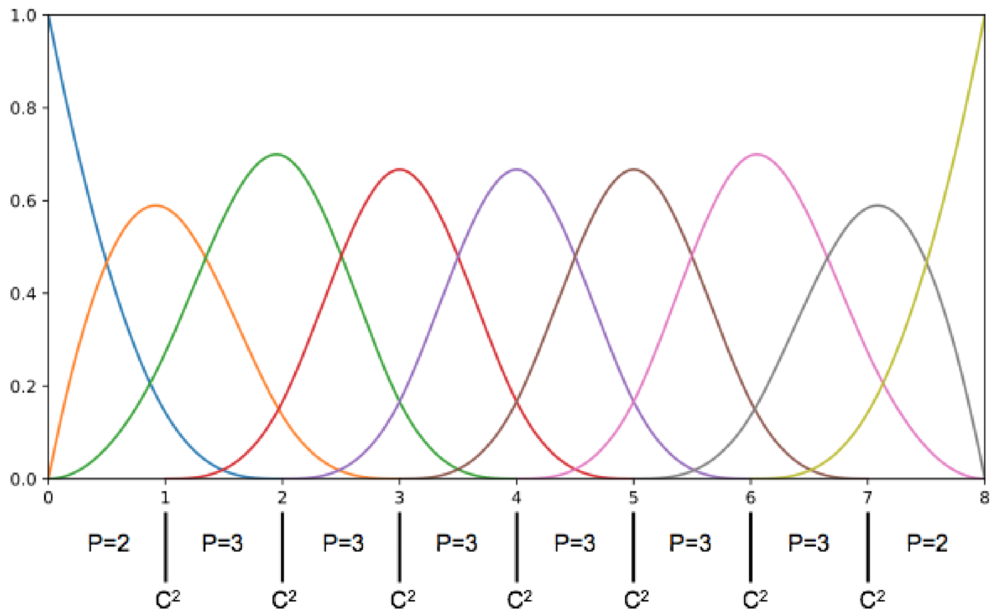

Fig.2: Non-uniform degree U-Spline in one dimension. Novel spline technology permits the construction of bases with varying degree while maintaining properties of partition of unity, linear independence, and positive local support. The U-Spline shown in this figure has been modified at the ends to reduce degree from P=3 to P=2 while maintaining C^2 continuity.

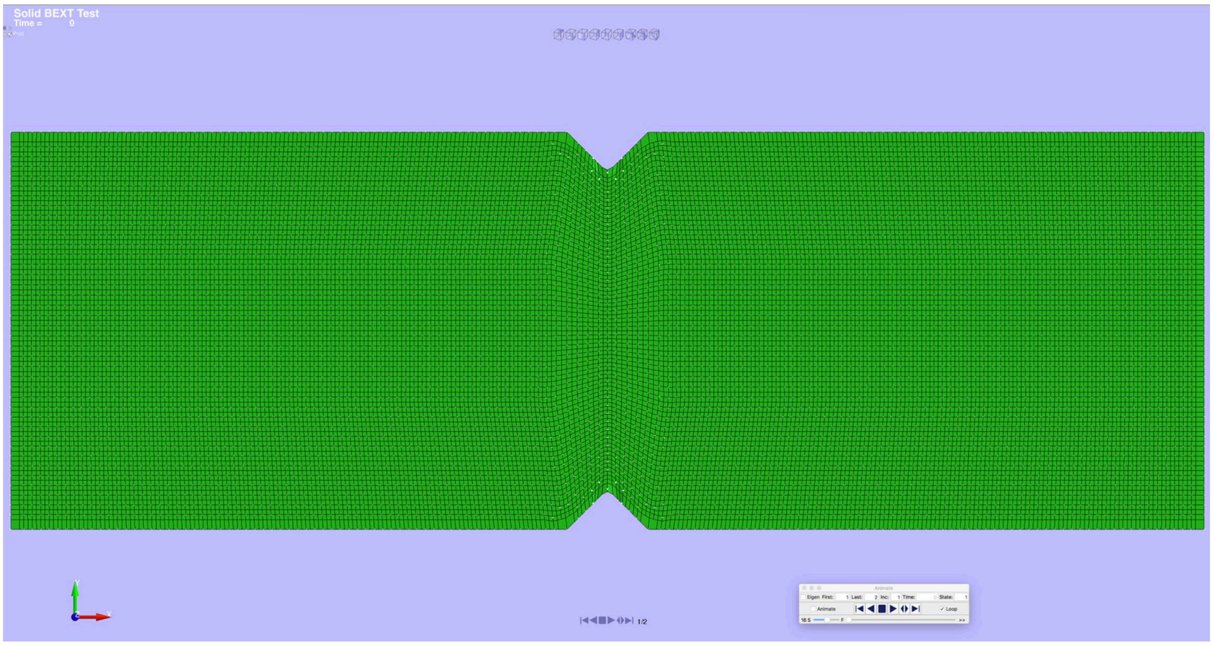

Fig.3: V-notch geometry as a mixed-degree, mixed-continuity U-Spline in LS-PrePost

In collaboration with Honda, the geometry for a fracture test (shown in figure X) was generated in Coreform Process and then read into LS-DYNA. A study was then done on the explicit dynamics timestep for different meshes created this way. Three variations of this model were created using (a) linear finite elements, (b) uniform degree U-splines, and (c) U-splines with modified degree near the boundary (shown in Figure 3), all with the same element layout. These models were then imported into LS-DYNA using the **`*IGA_INCLUDE_BEZIER`** keyword to create shell models. The linear elements went through this process as well so that the only difference between the models is the basis itself.

The stable time step was then estimated using LS-DYNA's largest eigenvalue estimate and a scale factor of 0.9 was used.

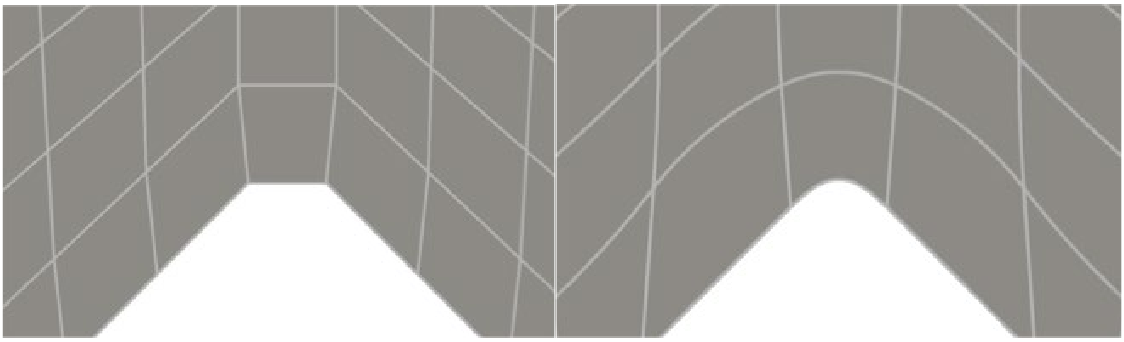

Fig.4: Notch geometry comparison, left: linear finite elements (P=1), right: U-splines with modified degree near the boundary (P=2).

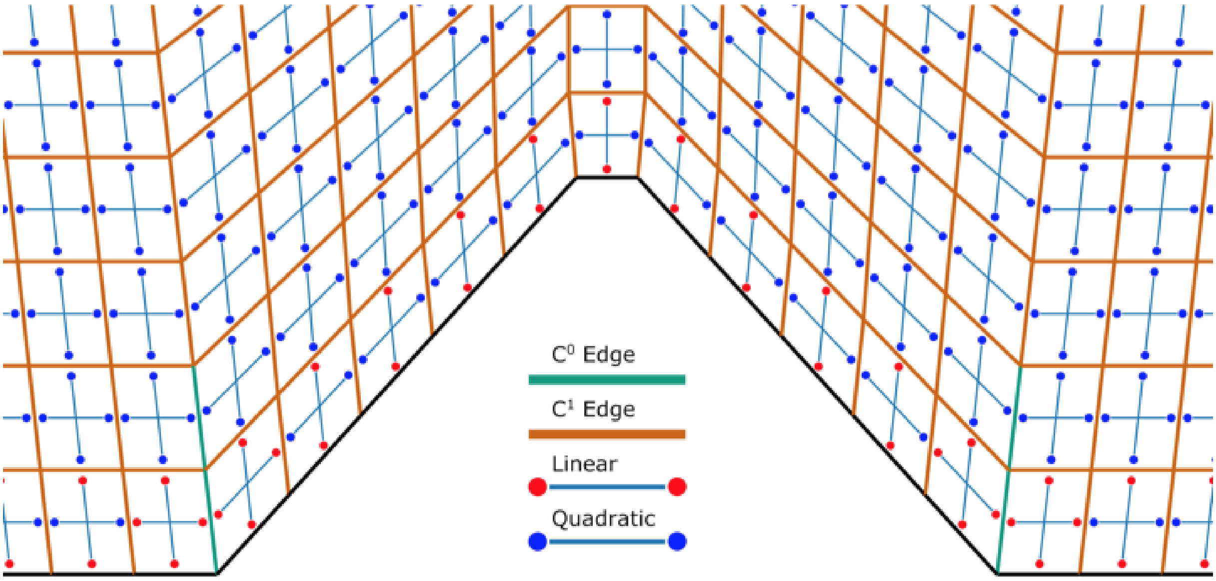

Fig.5: Modified quadratic U-Spline layout. Elements near boundaries (in black) or near C^0 edges (in green) were modified to decrease the degree and increase the characteristic length near the edges. Edges near corners were crease to C^0 to maintain geometric exactness.

The results of this study are shown in Table 2. It can be seen that unmodified splines can indeed be a limiting factor on simulation speed. However, U-splines that have been properly adjusted to fix the characteristic length produce faster explicit simulations without even changing element size. Additional benefit can be obtained through mesh coarsening to reduce the overall number of elements required and thus increase speed even further.

Basis Type	Time Step (as calculated in LS-DYNA)
Linear	1.35×10^{-7}

Quadratic unmodified U-spline	1.25 x 10^{-7}
Quadratic modified U-spline	2.18 x 10^{-7}

Table 2: Explicit time step size for different mesh types

4.2 Import of solid tensor-product spline model into LS-DYNA

The following is a spring contact buckling simulation provided by Honeywell. The solid geometry for this problem was created natively in Coreform Process and compared with traditional linear methods. As seen in Figure 6, a successful contact simulation was run with 300 solid spline elements as compared with 225,000 linear elements required to perform an identical simulation.

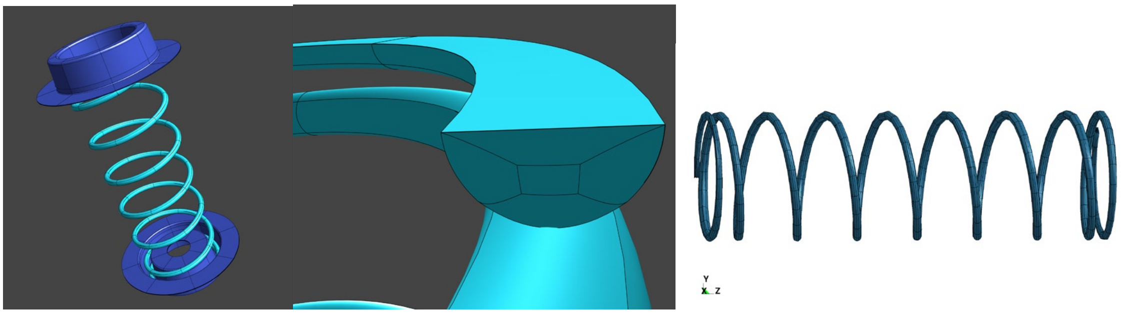

Fig.6: Left to right: Spring contact assembly in Coreform Process, Spring modeled with U-Spline cross section, solid spring geometry in LS-PrePost after export via ***IGA_INCLUDE_BEZIER**.

This solid geometry was exported from Coreform Process to LS-DYNA with the use of the new ***IGA_INCLUDE_BEZIER** keyword. This model represents complex geometry modeled with an extremely coarse mesh, now available to users in LS-DYNA.

4.3 Import of T-spline CAD model to LS-DYNA IGA model

An open source automotive console model shown in Figure 7 consists of 128 individual NURBS patches sewn together to create a single model. This model was generated as a single watertight T-spline model and then converted to a single U-Spline model. This model is ready for analysis and can be modified or refined in Coreform Process before exporting to LS-DYNA.

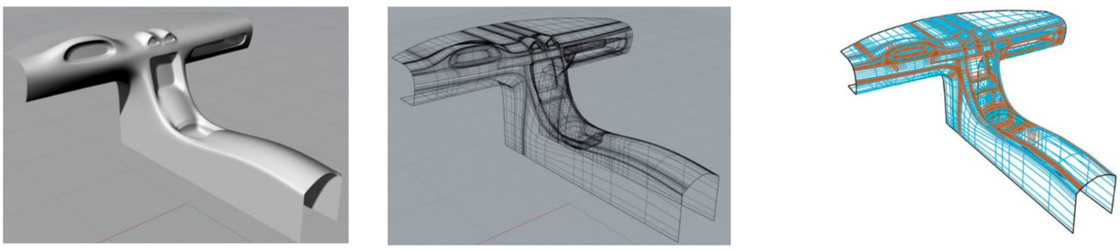

Fig.7: Left to right: Open source automotive console CAD model, model converted to T-Splines, model converted to U-Splines

4.4 Smoothing existing unstructured FEA mesh via U-splines to create LS-DYNA IGA model

The following hood surface geometry provided by Ford also demonstrates the power of this mesh conversion workflow. The LS-DYNA linear surface mesh was read into Coreform Process and converted to a U-Spline as shown in Figure 8. This basis was then elevated to increase the overall smoothness of the geometry shown in Figure 9.

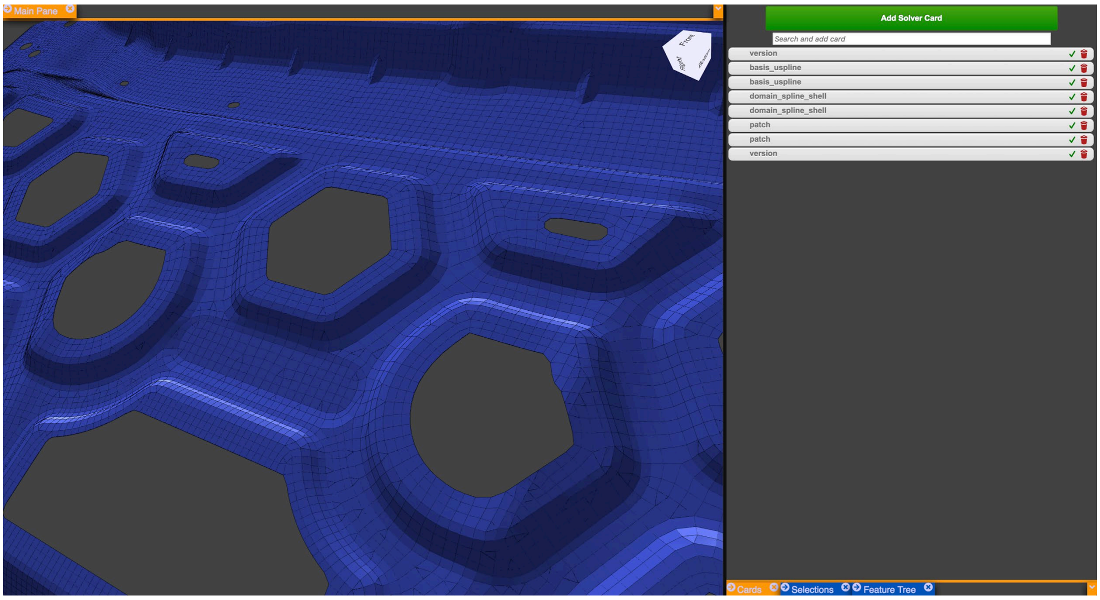

Fig.8: Ford hood geometry as a linear U-Spline in Coreform Process

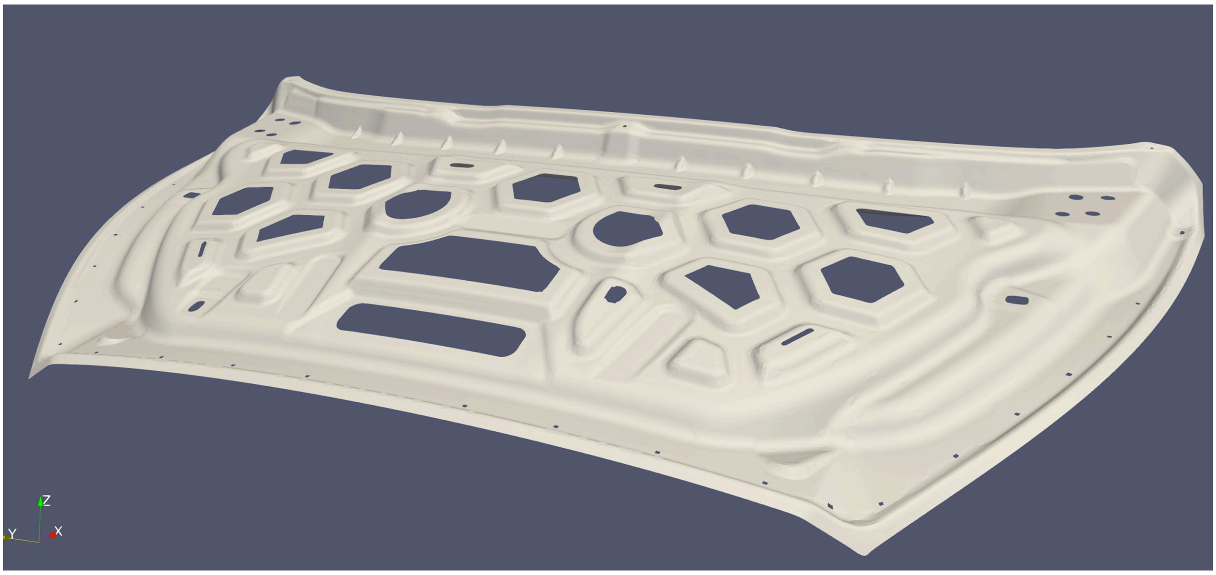

Fig.9: Ford Hood constructed as a P=2, C^1 dominant U-Spline

It should be noted that limitations still exist in the smoothness possible for these translations. Though U-Splines can incorporate triangles and extraordinary points, these features currently must be creased to C^0 smoothness. Since the linear mesh is then projected back on this basis to create the geometry, creases near such topological elements reflect the linear mesh. Coreform is currently developing technology that will allow U-Splines built from linear meshes to use the CAD geometry directly for projection, which will improve these geometric projections.

4.5 Future possibilities: IGA assembly models in LS-DYNA

Coreform Process and Analyze now support basic assembly simulations, and anticipates working with LSTC in the future to ensure these will convert for IGA assemblies in LS-DYNA.

One example of an IGA assembly run in Coreform Analyze, shown in Figure 10, is the conversion of an LS-DYNA assembly with over 20 individual linear mesh parts to a U-Spline model. *p*- and *k*-refinement were then performed on the mesh to increase the degree and increase the smoothness where possible.

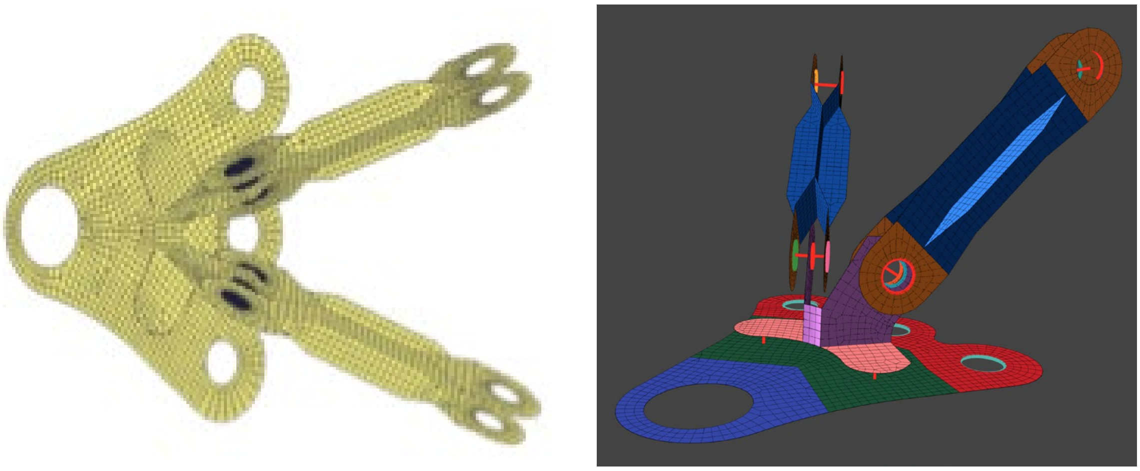

Fig.10: Left: Linear model in LS-DYNA, Right: P2 U-Spline model in Coreform Process

This example marks one of the most complex IGA assembly simulations ever completed and demonstrates the ability to automatically generate smooth spline models for analysis. This simulation was run in Coreform Analyze and compared with an analogous linear mesh simulation in LS-DYNA. These results, shown in Figure 11, demonstrate that full IGA simulations compare well with traditional methods. The individual parts of this smooth U-spline assembly have been successfully imported to LS-DYNA via **`*IGA_INCLUDE_BEZIER`**, and when IGA support in LS-DYNA is extended to support assemblies, Coreform looks forward to converting the entire assembly model back to a .k file.

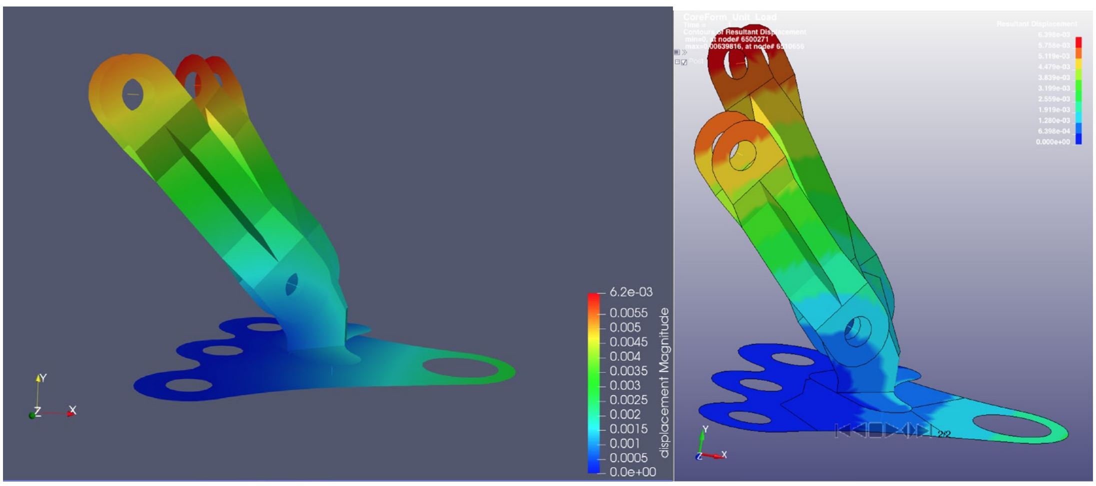

Fig.11: Left: Simulation results of full IGA assembly in Coreform Process, Right: Simulation results of an analogous linear analysis in LS-DYNA

4.6 Test suite examples

Coreform's test suite to verify the workflow between geometry created in Coreform Process and LS-DYNA includes examples such as those in Figure 12 that show simple demonstrations of the novel mesh constructions now accessible in LS-DYNA through Coreform Process.

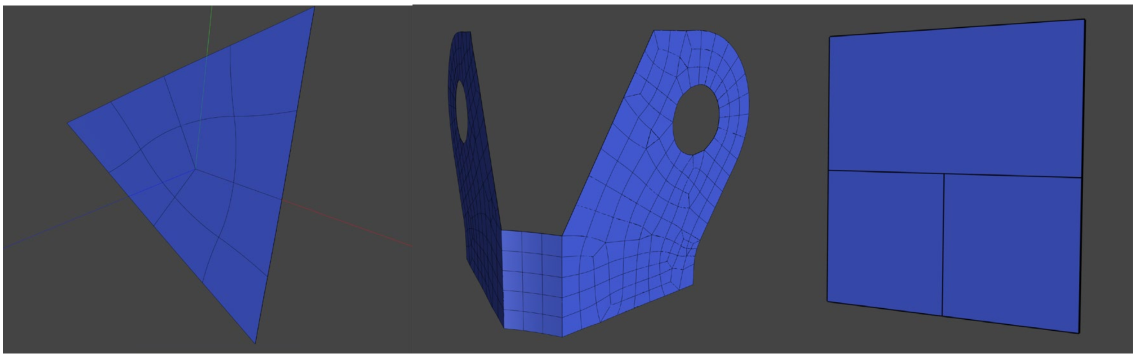

Fig.12: Unstructured meshes read into LS-DYNA. From left to right: P=2 mesh with an extraordinary point, P=2 mixed element types and mixed continuity, P=3 C2 mesh with T-junction.

5 Summary

The **`*IGA_INCLUDE_BEZIER`** keyword opens the door to novel workflows that allow unstructured splinesk, including U-splines and T-splines, to be used in LS-DYNA. A significant benefit that this makes available is the ability to increase time step size in LS-DYNA through the use of modified U-splines. This potentially overcomes a prior barrier to the use of IGA in LS-DYNA, which is that time steps for higher-order meshes had been observed to be significantly smaller than those of linear elements.

Additionally, we showed several workflows now possible, including importing solid tensor-product spline models into LS-DYNA, importing T-splines CAD models into LS-DYNA, and smoothing existing unstructured FEA meshes via U-splines to create LS-DYNA IGA models. Finally, we introduce the future possibility of leveraging the **`*IGA_INCLUDE_BEZIER`** keyword to import IGA assembly models into LS-DYNA.

The authors would like to thank the IGA team at LSTC and acknowledge their assistance in the preparation of this article. They would also like to thank Honeywell, Honda, and Ford for support and test cases.

6 Literature

[1] J. A. Cottrell, T. J. R. Hughes, and Y. Bazilevs, Isogeometric Analysis: Toward Integration of CAD and FEA. 2009.

[2] T. J. R. Hughes, J. A. Cottrell, and Y. Bazilevs, “Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement,” Computer Methods in Applied Mechanics and Engineering, vol. 194, no. 39, pp. 4135–4195, Oct. 2005.

[3] M. A. Scott, M. J. Borden, C. V. Verhoosel, T. W. Sederberg, and T. J. R. Hughes, “Isogeometric finite element data structures based on Bézier extraction of T-splines,” Int. J. Numer. Meth. Engng., vol. 88, no. 2, pp. 126–156, Oct. 2011.

[4] D. Thomas, L. Engvall, S. Schmidt, K. Tew, and M. Scott, “U-splines: splines over unstructured meshes,” p. 48.