A Roadmap to Linear and Nonlinear Implicit Analysis in LS-DYNA

George Laird[1], Satish Pathy[2]

[1]Predictive Engineering, Munich, Germany

[2]LSTC, Livermore, USA

1 Abstract

The default LS-DYNA settings are tailored for running large explicit analyses. For new and even experienced users, it can be challenging setting up an implicit LS-DYNA analysis to match analytical solutions or other standard implicit FEA codes. For example, the default element formulations are based on single-point integration whereas implicit analyses benefits from full-integration. A series of example problems are provided that will allow the simulation engineer to exactly match industry standard implicit codes (complete keyword decks can be found at DYNAsupport.com). Along with these example decks, CPU-scaling results will be presented for each implicit analysis type from linear to nonlinear.

2 Introduction

Basic linear implicit analysis (including eigenvalue analysis) might represent 80 or more percent of all the analysis work done in the world and most likely nonlinear implicit analyses from mild to severe constitute another 10 percent. For users of LS-DYNA and their business organizations, there are many advantages to the adoption of one FEA code to solve the complete range of problems from the most simple (linear elastic static) to the most complex (nonlinear transient). The literature is rich in studies on the accuracy of LS-DYNA toward solving a wide-range of explicit problems (see DYNAlook.com). On the implicit side, only a few publications have been generated [1-5] that provide direct guidance to the simulation engineer. As users of LS-DYNA are well familiar with, the code's default settings are focused on the efficient solution of large, transient, nonlinear finite element (FE) models. Given this background, the default element settings, control settings and post-processing data sets are not applicable for an implicit analysis and nor should they be. This can cause problems for someone coming from an implicit analysis code where the default settings are for static linear elastic analysis. For example, to run an explicit analysis in LS-DYNA, one need not touch any of the default settings, merely set the *CONTROL_TERMINATION time and the problem will run. For an implicit analysis, there are lots of options and some you want and some you don't. As such, in this work we attempt to present concise guidelines for running classical linear implicit problems and also those for nonlinear analysis, we leverage prior work by DYNAmore Nordic [1]. We also show how commercial-sized implicit problems can scale using multiple CPU-cores on a PC-workstation (3.1 GHz dual-socket (20 true CPU-cores (hyper-threading turned off)) with 256 GB of RAM and a 2 TB PCI-SSD).

2.1 What Types of Problems Can Implicit LS-DYNA Analysis Solve?

Generally, we look for problems that are not overly nonlinear and are better suited to be solved statically rather than dynamically. Fig. 1 shows a few of the implicit models that we have solved at Predictive. Other implicit examples can be found at the www.DYNAexamples.com website.

Drop, Rail Impact and PSD Analysis
of Composite Container (MAT_54 Failure)
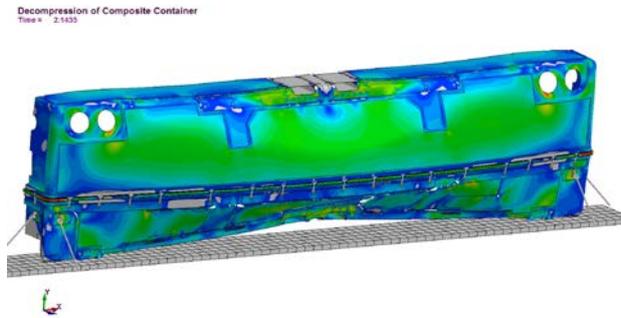
Axisymmetric Rubber Seal Analysis
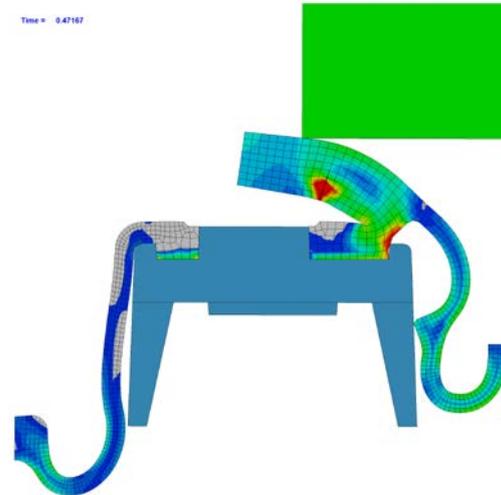
9g Crash Analysis of Jet Engine Stand
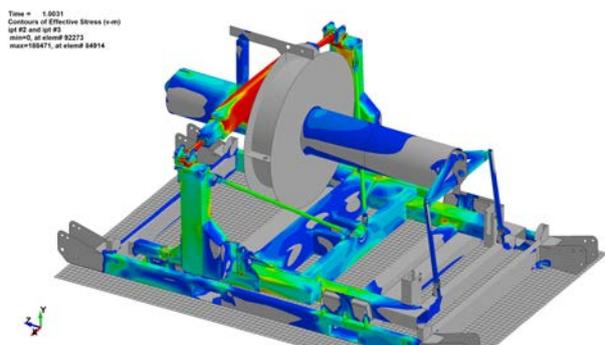
Braze Process Simulation (MAT_188)
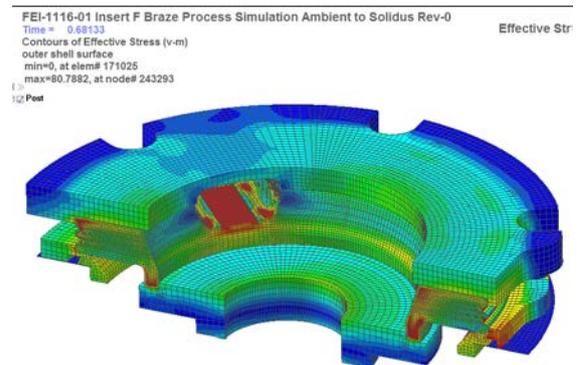
Stress and PSD Analysis
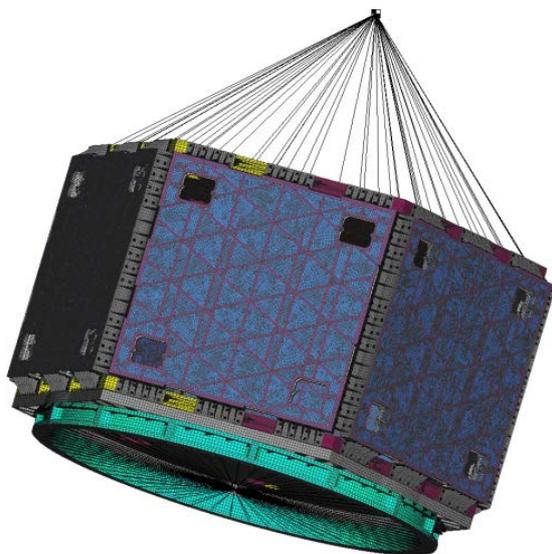
Torque Analysis of Endoscopic Medical Device
(Beam-on-Beam Contact)
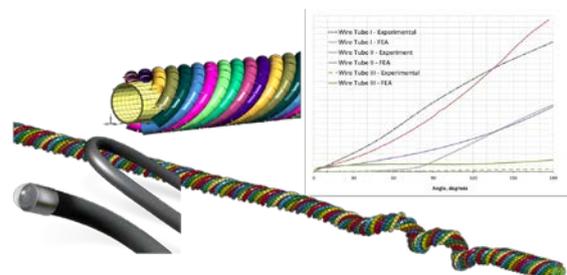
Fuel Tank Impact Analysis	Plastic HDPE Acid Storage Tank
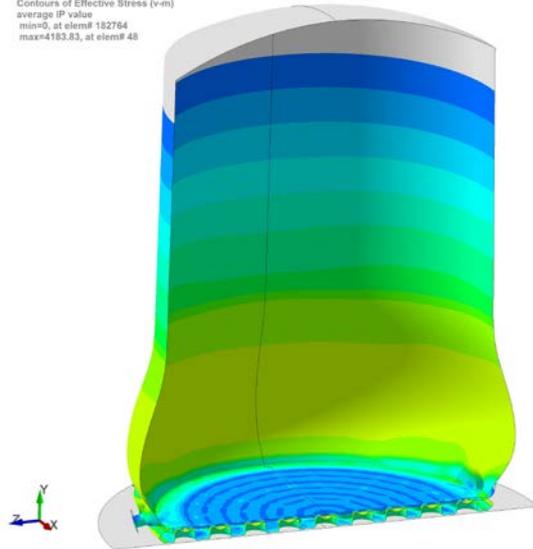
| Cargo Net Analysis – 9g Crash Landing Load	Nylon 12 Watch Band
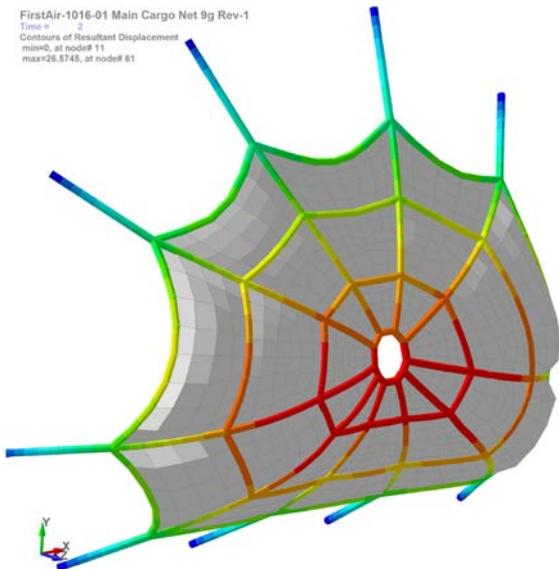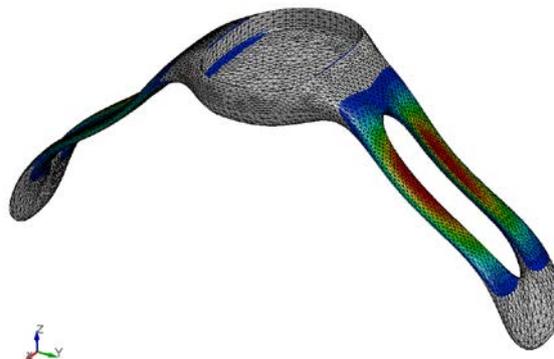
Fig. 1: Examples of implicit LS-DYNA problems (courtesy of Predictive Engineering)

3 Linear Elastic Implicit Analysis

3.1 Simply-Supported Beam

A simply-supported half-symmetric beam is analyzed using beam, shell and solid (8-node brick and 10-node tetrahedral) elements. This basic example is used to demonstrate that LS-DYNA can solve the most basic of linear elastic problems. Fig. 2 provides a side-by-side comparison of results for the beam, shell and solid models using the same mesh. A uniform pressure load is applied to the surface to generate a stress magnitude of 1,000 units at the centerline of the half-symmetric beam. For the beam element, the load was applied on a per-length basis to be equivalent. For the shell elements, the maximum principal stress was contoured to show how the shell stresses are contoured.

Table 1 presents a summary comparison with a % difference against a standard FEA code.
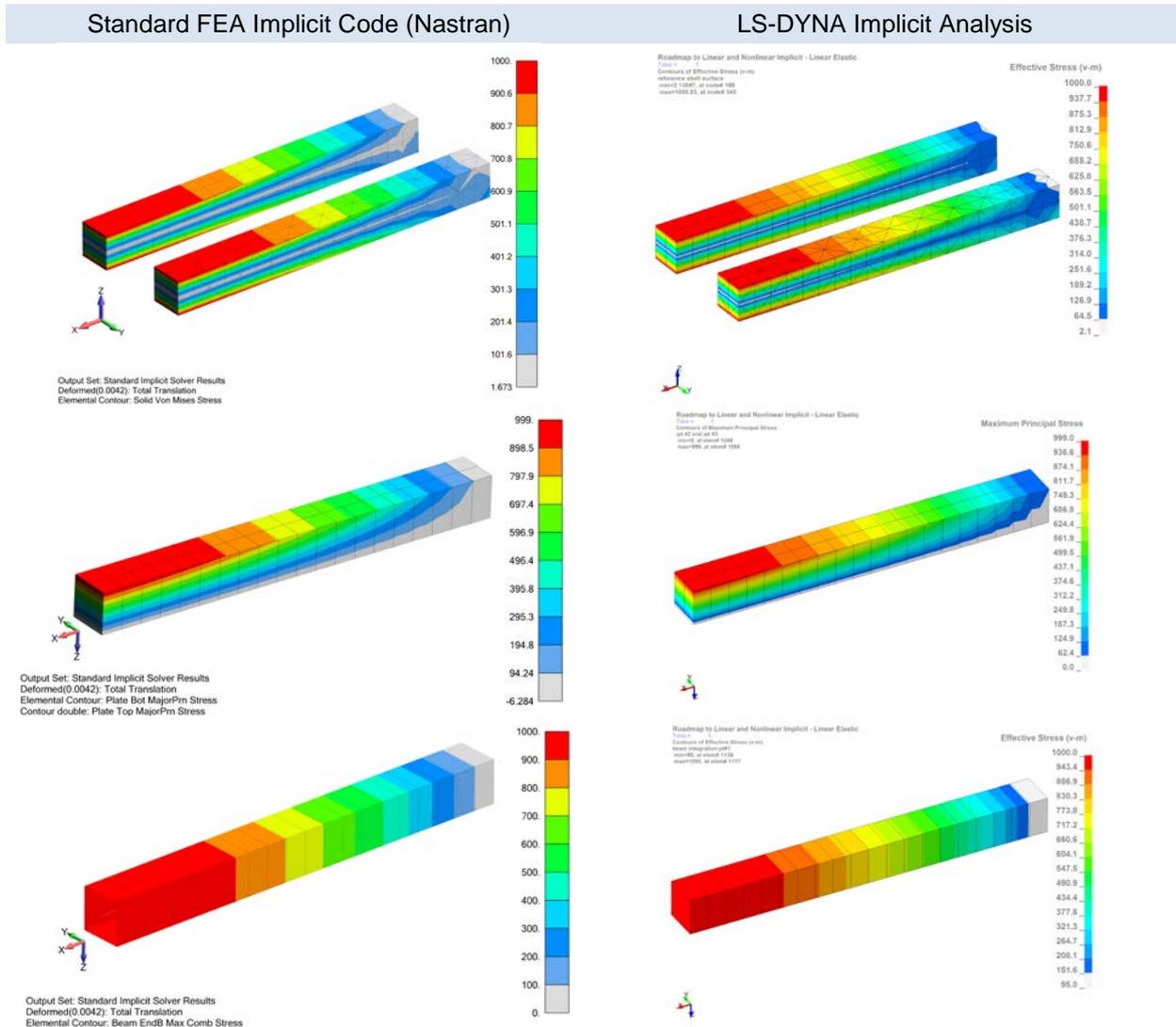
Fig. 2: Implicit LS-DYNA verification against standard implicit FEA code (Nastran)

Table 1: Summary of linear elastic implicit verification results

Model	Hex		10-node Tet		Shell		Beam	
	Stress	Disp	Stress	Disp	Stress	Disp	Stress	Disp
Standard	999.0	4.185e-3	1000.	4.194e-3	999.1	4.192e-3	1000.	4.190e-3
LS-DYNA	999.3	4.184e-3	1000.	4.192e-3	999.0	4.192e-3	1000.	4.192e-3
% Difference	0.03	0.02	0.00	0.05	0.01	0.00	0.00	0.05

3.2   Plate with a Hole

For a uniformly loaded, infinite plate with a hole, the maximum stress concentration is 3x the far field stress. Fig. 3 shows the geometry and loading setup and a sketch indicating the stress mechanics. The utility of this example is that a finite, well-defined stress concentration is created that provides a direct comparison to the stresses calculate by a FE model.

| Plate with Hole Under Uniform Tension	Courtesy of www.fracturemechanics.org
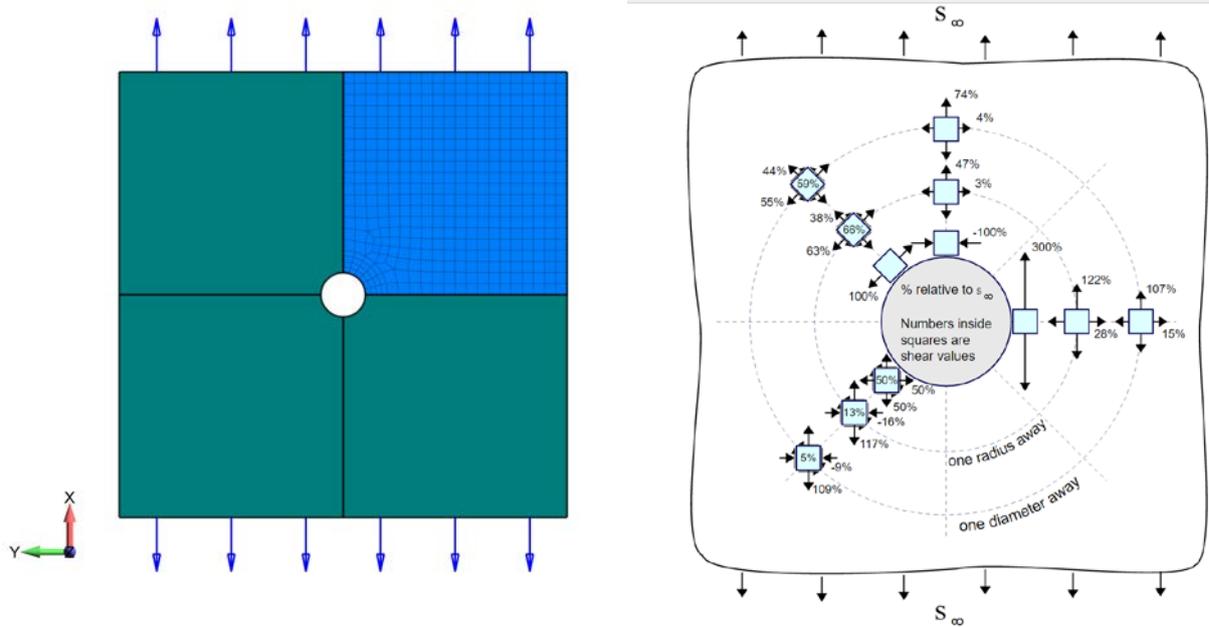
Fig. 3: Plate with hole under uniform tension with schematic of stress distribution

Fig. 4 shows stress results for the solid and shell element formulations. As in the prior example, the same meshes were used between the two programs. In contouring solid element stresses within LS-PrePost, several options are available for averaging nodal stresses: mid, ave and max. The mid option takes a simple average between connected nodes and was used in the hex and tet models. For the shell model, extrapolate 1 was used within LS-PrePost to extrapolate the stresses from the integration points and then averaged using the default setting (mid).

Standard FEA Implicit Code (Nastran)	LS-DYNA Implicit Analysis
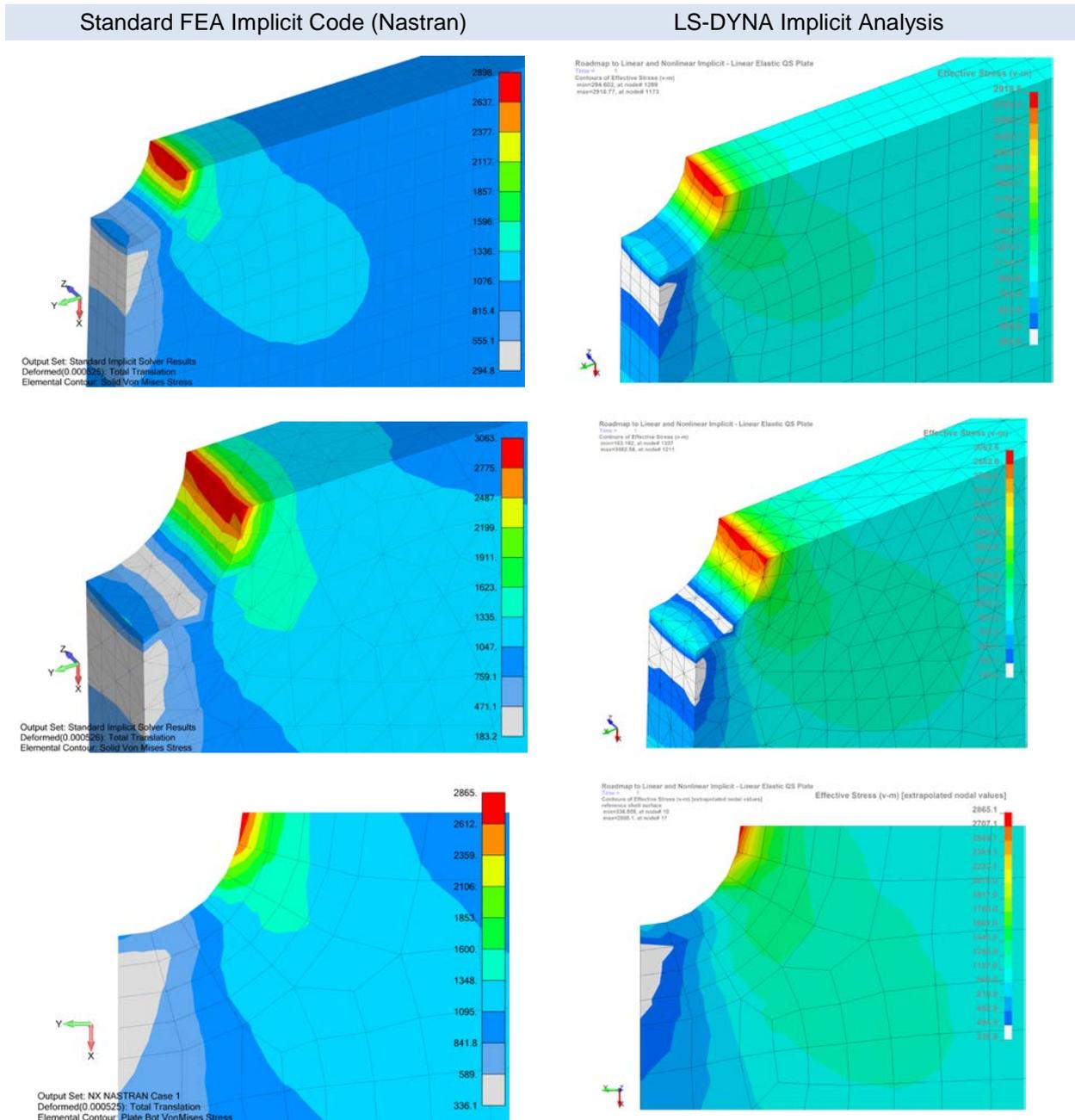
Fig. 4: Comparison of linear elastic implicit results using stress concentration model

Table 2: Summary of linear elastic results for QS plate with hole

	Hex	Tet	Shell
Standard	2898	3063	2865
LS-DYNA	2919	3063	2865
% Difference	0.72	0.00	0.00

3.3   Composite Analysis

Given the importance of linear elastic composite analysis, it merits a discussion on how to set-up LS-DYNA to obtain similar results as that generated by a standard implicit code. We will leverage an example model from prior work on composites [9]. For a discussion on setting up failure criteria, there are several excellent papers by Feraboli et al. [10 & 11] and LSTC's own note set [12]. For a commercial example of using LS-DYNA for progressive failure simulation in composites, one can also look at Jensen et al. [7] and for a brief overview of basic composite analysis in LS-DYNA one can read the newsletter article by Laird [13].

Fig. 5 provides a comparison between the first and fourth plies of an eight ply laminate composite plate with a hole. The analysis is linear elastic. For the LS-DYNA model, the shell formulation is ELFORM=-16 (minus sign 16). To request ply information, use the *DATABASE_EXTENT_BINARY setting maxint=8 to write out integration point data for each ply. The reason for not requesting all integration points on each layer (i.e., ply) using -8 is that Nastran only reports the centroid value as a default and we don't wish to make this comparison more difficult than necessary.

A classic trip up when setting up the LS-DYNA *MAT_54 card is using the correct value for Poisson's ratio (γ). In a Nastran code, one enters γ_{12} whereas in LS-DYNA, one enters γ_{21} or PRBA. If one is converting from Nastran, then $\gamma_{21} = \gamma_{12}(E_2/E_1)$.

| Nastran – Ply 1 & 4	LS-DYNA Ply 1 & 4
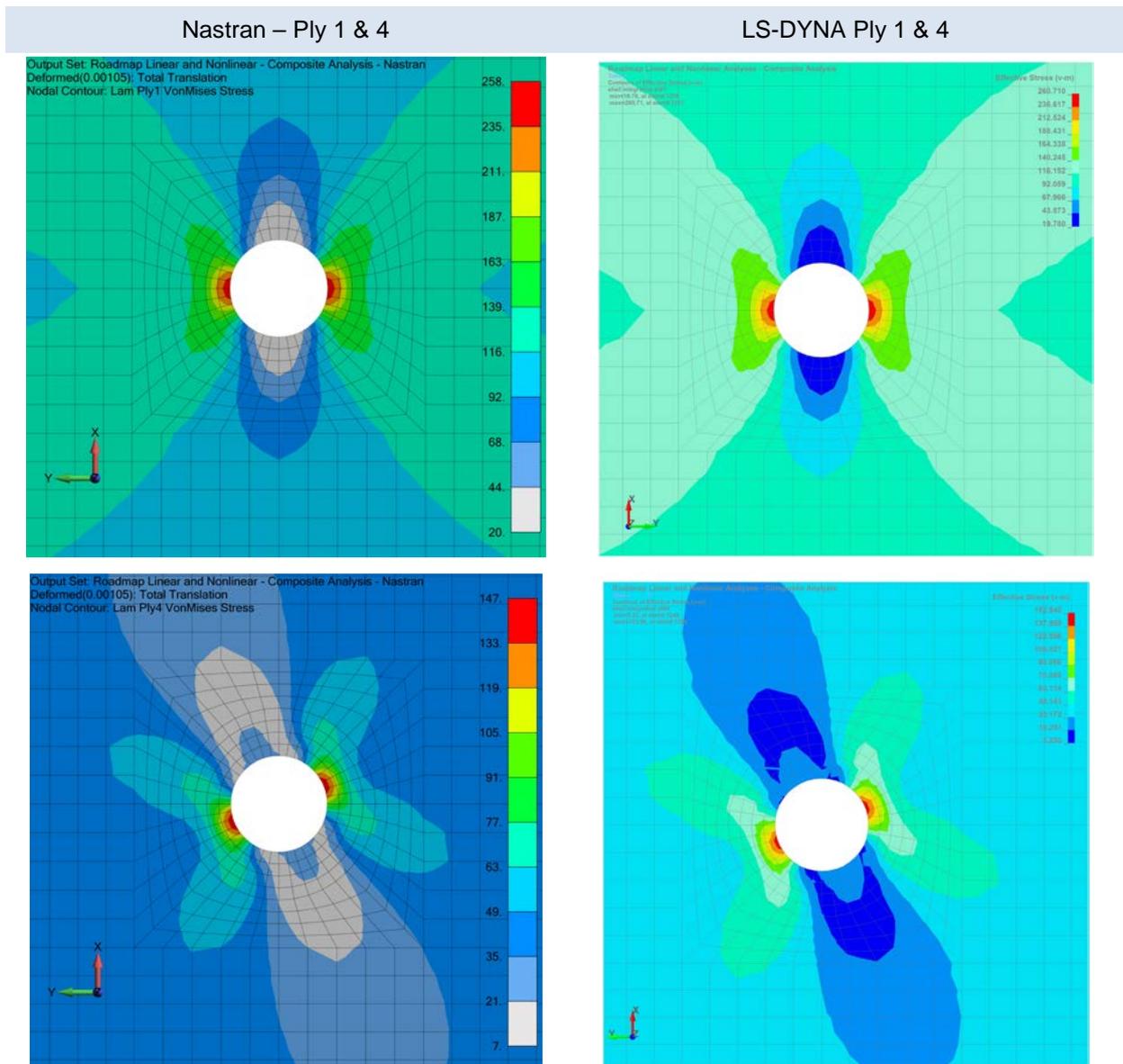
Fig. 5: Comparison of composite linear elastic stress results between Nastran and LS-DYNA

3.4 Linear Connectors (Equivalent Nastran Multi-Point Constraint Elements)

In Nastran implicit analysis, it is quite common to use connectors that are based on constraint relationships between stiffness terms within the stiffness matrix. In Nastran they are termed multi-point constraint elements (MPC's) and depending upon their formulation are also known as rigid elements (e.g., RBE1 and RBE2) or force interpolation elements (e.g., RBE3). In the Nastran analysis sequence, the MPC relationship is created, the matrix decomposed and then forces calculated. Since it is linear, the MPC is defined based on the initial terms of the stiffness matrix. As one can imagine, it is not a useful numerical approach for a nonlinear analysis but for a linear analysis, it has been the standard for thirty plus years. As a comparison, Fig. 6 provides a side-by-side comparison between the two codes. A force is applied to the center of the connector (same force for both connectors) and the plate is pushed downward. The edges of the plate are pinned. For the RBE2 case, the difference is 1.9% while that for the RBE3 example, the difference is 4%. In both cases the stress patterns are nearly identical and although the stress differences are greater than would be ideally desired in a linear elastic analysis, their differences can be explained by the completely different connection formulation between Nastran and LS-DYNA and not the underlying element formulation. (*Note:* When the two models are analyzed without connectors the stress results are numerically identical.)
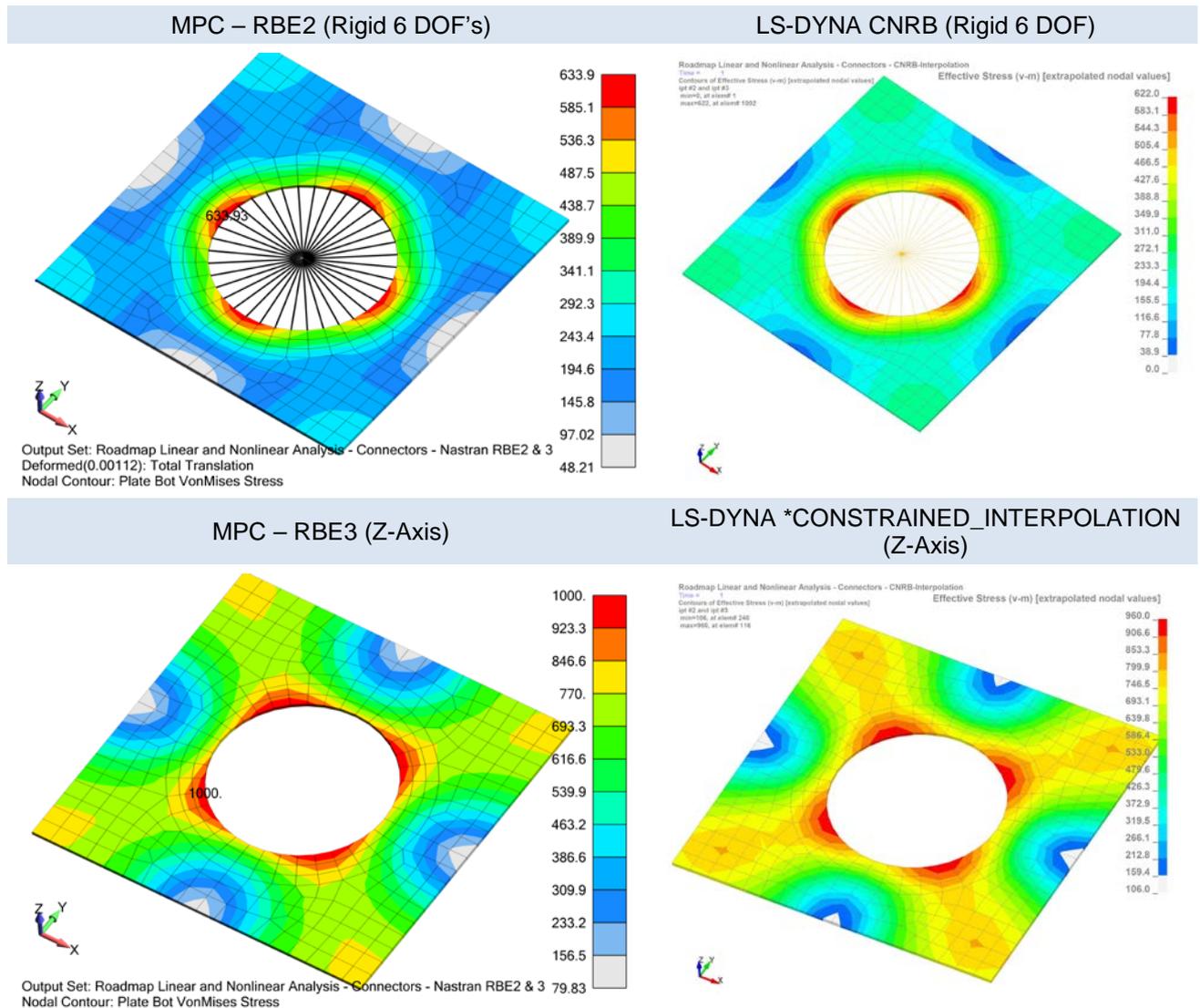
Fig. 6: Comparison between Nastran MPCs and LS-DYNA CNRB and Interpolation Connections

3.4.1 Displacement Comparison

To provide a little background on the mechanics of these two connectors, Fig. 7 shows the displacements between the rigid and interpolation connectors. The rigid connection (RBE2 and CNRB) acts as if one has welded a rigid plug into the hole while the interpolation connection (RBE3 or *CONSTRAINED_INTERPOLATION) only distributes the applied force and adds no stiffness to the structure. This explains why the displacements are very different between the rigid and the interpolation connections. As for a comparison between displacements, the Nastran results are 1% lower than that of LS-DYNA.
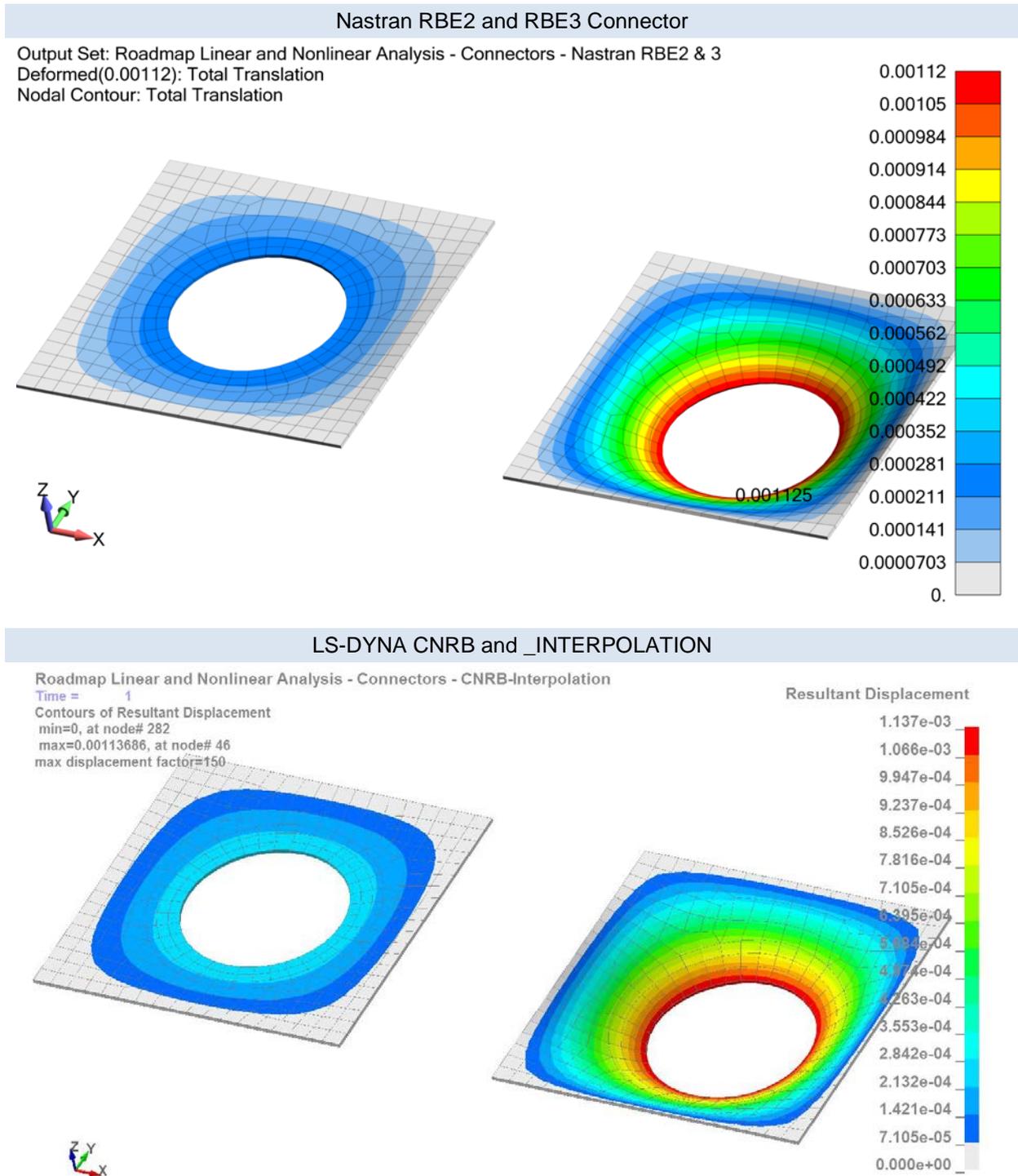
Fig. 7: Comparison of displacements between rigid and interpolation connectors

3.5 Linear Elastic Analysis LS-DYNA Keywords

The complete LS-DYNA keyword decks for these models can be found at www.DYNAexamples.com.

3.5.1 Control Section

*CONTROL_IMPLICIT_ACCURACY with iacc=1 {the iacc parameter activates a number of numerical improvements tailored to improve implicit accuracy. It should always be activated for an implicit analysis. The general intent of this command is to improve implicit accuracy while still allowing the analysis to be switched to an explicit routine if required. It should be mentioned that it is a very powerful option and is often updated with new capabilities.}
*CONTROL_IMPLICIT_GENERAL with imflag=1 and dt0=1.0 {implicit analysis with one time step}
*CONTROL_IMPLICIT_SOLUTION with nsolvr=1.0 {default linear solver}
*CONTROL_OUTPUT with solsig=1 {linear stress extrapolation from the solid element integration points}
*CONTROL_SHELL with intgrid=1 {lobatto integration}

3.5.2 Database

*DATABASE_EXTENT_BINARY with neips=-3, beamip=9 and nintsld=8 {write out all shell, beam and solid element integration points}

3.5.3 Section

*SECTION_SOLID with elform=16 {standard 10-node tetrahedral}
*SECTION_SOLID with elform=18 {8-point enhanced strain solid}
*SECTION_SHELL with elform=21 and nip=3 {full integrated assumed strain C0 shell}
*SECTION_BEAM with elform=4 and qr/irid=4 {Belytschko-Schwer full cross-section integration with 3x3 Lobatto quadrature}

3.5.4 LS-PrePost Commands

For contouring solid element stresses one should note that stress averaging can be done from taking the min, mid, ave or max using the Fringe Component option (see Fig. 8 lower drop-down menus labeled Min and Ave). The averaging method follows standard FE post-processing conventions with Mid being the simple average of all the connected nodes. The Ave method contours the element's average stress value. This value is identical to the element's centroid stress value. One can access the convergence of the mesh by comparing stress plots using the Mid and Ave options and qualitatively access the jump in stress between centroid and grid point. If greater than 20% one might want to refine the mesh.
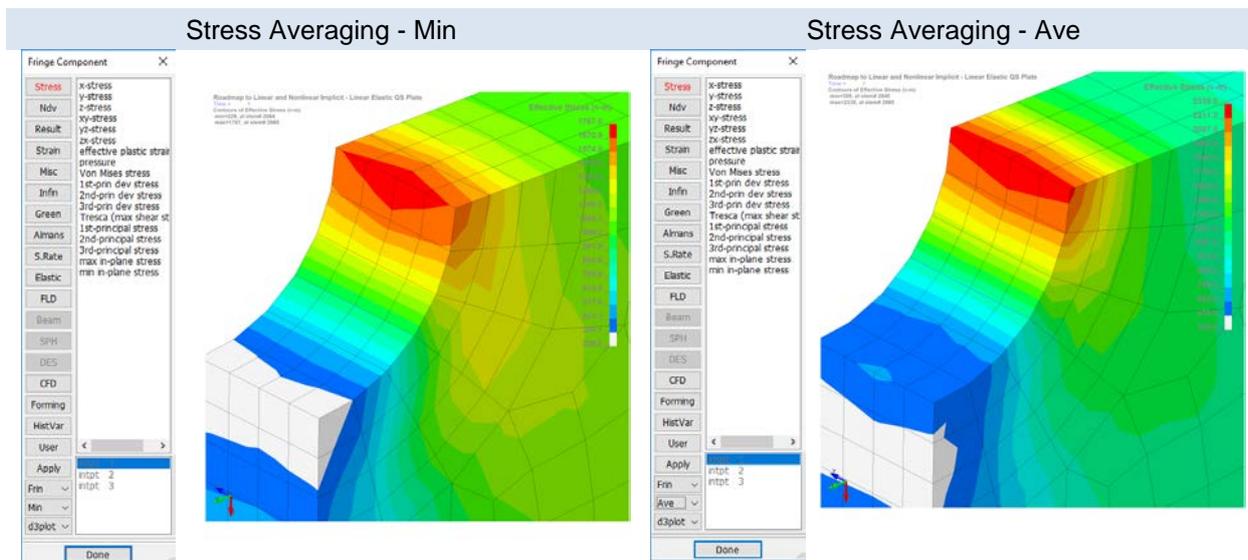
Fig. 8: Stress averaging of solid elements within LS-PrePost

The Extrapolate command within LS-PrePost was developed several years ago prior to the implementation of the solsig command within *CONTROL_OUTPUT. In the solsig command the LS-DYNA solver performs the extrapolation and swaps out the element's integration point stress items for node point stresses. LS-DYNA's method is based on the Superconvergent Patch Recovery by Zienkiewicz and Zhu [8]. This is standard for many linear implicit codes but is only strictly relevant for linear elastic analysis since the extrapolation technique assumes linear behavior. Once plasticity occurs or nonlinear elastic behavior, the linear extrapolation of stress/strain items from integration points is incorrect. Of course, there are extrapolation routines that can be used for elements that have plastically deformed but that is a subject for further research. As for shell elements, a similar command has not been implement within LS-DYNA and to extrapolate shell integration points to grid points, the LS-PrePost interface is used by entering extrapolate 1 in the Command line (lower, left-hand corner of the interface). Fig. 9 provides a comparison of stress results in the default and extrapolated mode.

No Extrapolation (Default)	Extrapolate 1
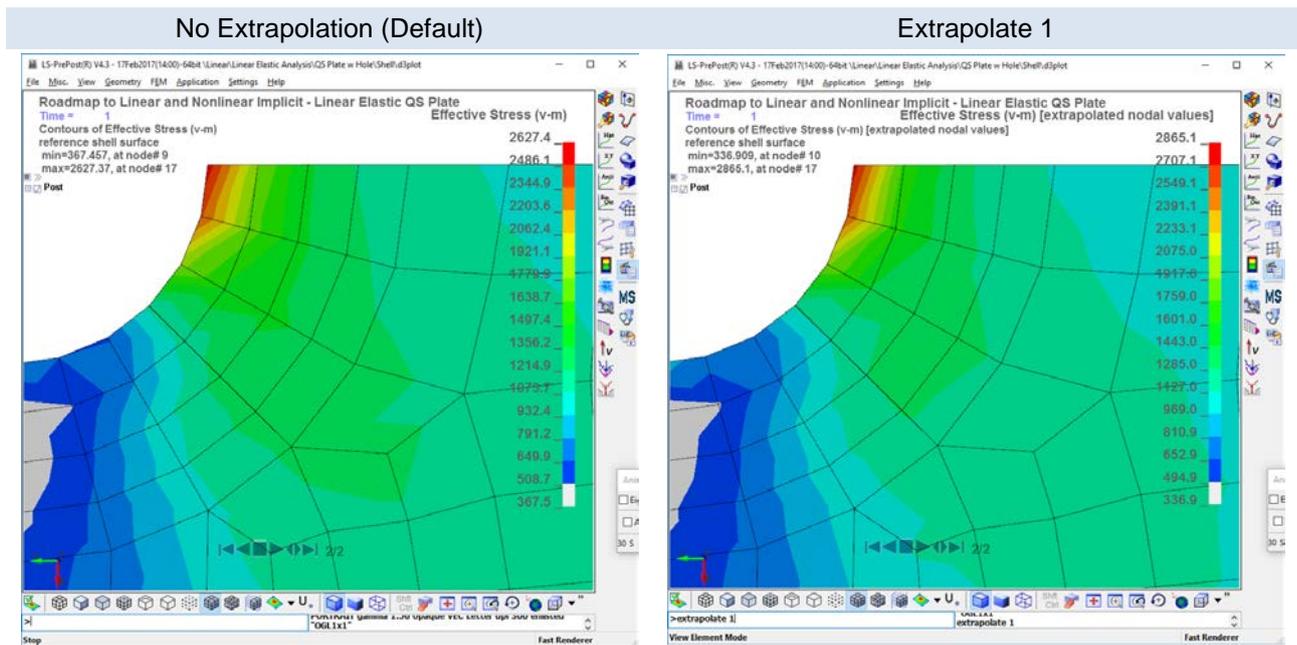
Fig. 9: Extrapolate 1 command in LS-PrePost for linear elastic shell element post-processing

3.5.5 Miscellaneous LS-PrePost Comments

Within LS-PrePost, under the Settings drop-down menu are two useful dialog boxes (see Fig. 10). Given that linear elastic displacements are normally quite small, it is handy to scale the displacements. Plus one can set the Extrapolate option to be a default configuration setting. Controlling the post-process legend is very useful when one notices that your contoured displacements read "zero" due to the default setting rounding off your contoured values.

Displacement Scaling and Extrapolate	Setting the Post-Processing Legend
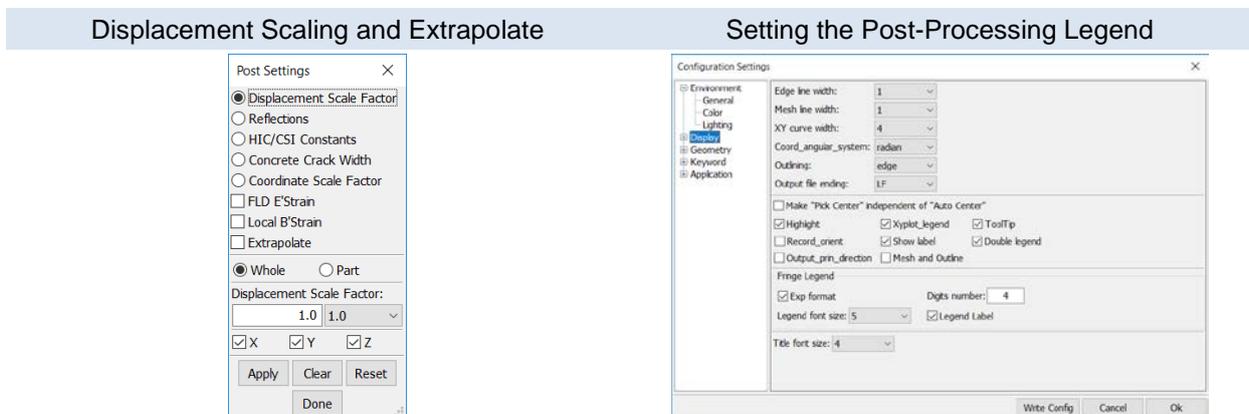
Fig. 10: Useful post-processing commands within LS-PrePost

3.6 Linear Elastic Scaling Results

Scaling results are presented for three commercial examples of linear, elastic FEA models: (i) Top drive casting for the off-shore petroleum industry (10-node tetrahedrals, beam and CNRB elements); (ii) a small commercial satellite (shell, hex and CNRB elements) and (iii) a pressure vessel composed of shell, beam and CNRB elements. Fig. 11 shows a comparison of the results between a Nastran code and LS-DYNA. The scaling results are given in Fig. 12.
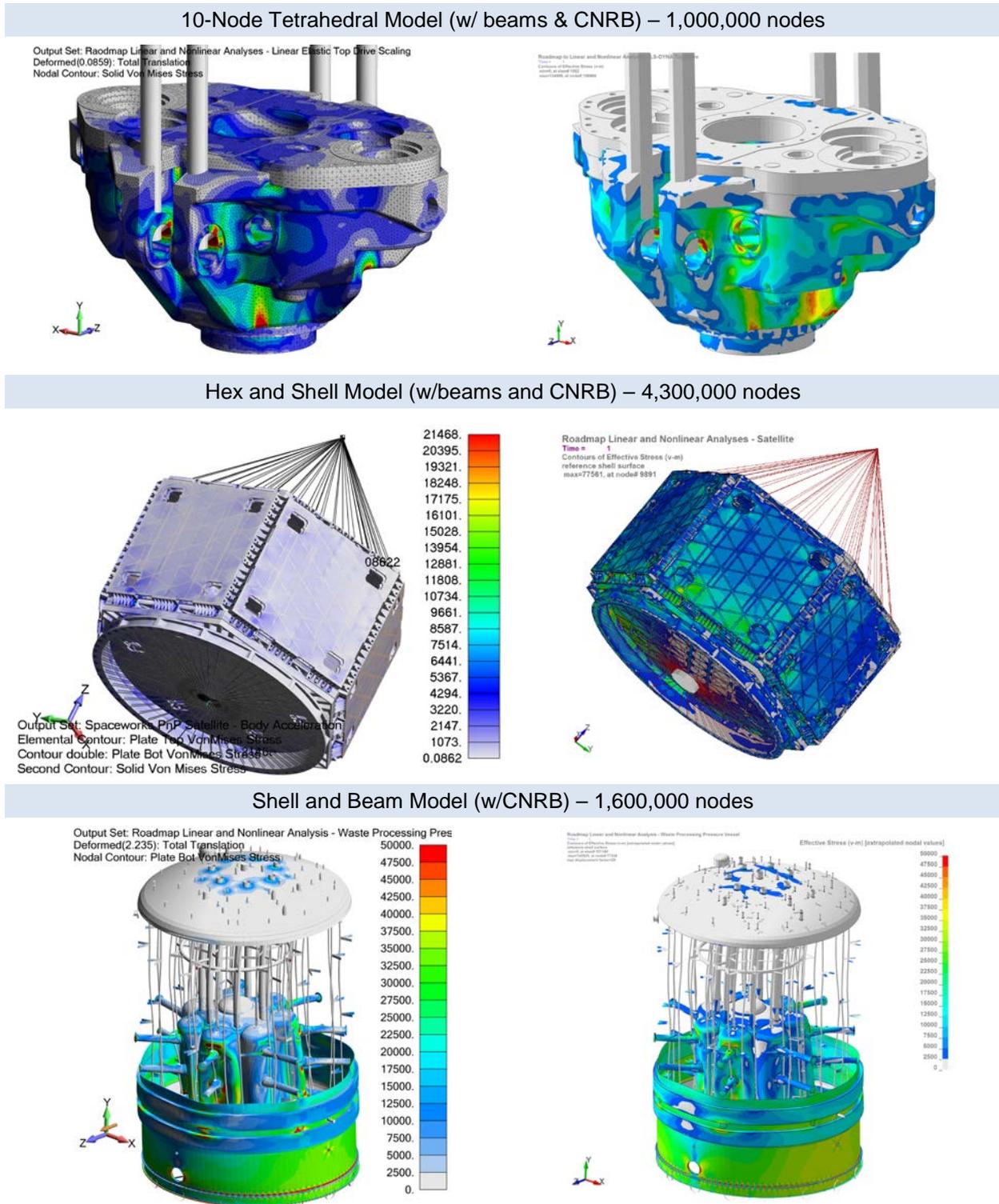
Fig. 11: Linear, elastic models used for scaling study

Although the models presented for the scaling study may not represent models in the multiple millions of nodes, their complexity are representative of standard commercial models that typically have a combination of element types using beam, shell and solid elements and connector elements (e.g., RBE2 and RBE3 types). The reader may note that we have not included a pure 10-node tetrahedral model that is normally solved using an iterative solver (e.g., PGCLSS) but are staying within the sandbox of sparse matrix solvers. This is done for two reasons: (i) we rarely encounter these rare birds and (ii) LS-DYNA does not have a comparative iterative solver within its code.

As shown in Fig. 12, the linear implicit LS-DYNA solver scales nicely to eight CPU-cores and then starts to taper. The default linear solver for MPP (LSOLVR=5) was used for all three models. As an exploratory study, then new solvers in r9.1.0 were evaluated. LSOLVR=22 showed roughly 20% improvement over the default solver while the other new solvers (23, 24, 25 & 26) went the other way and were slower by about 15%. The computer platform was a PC workstation (dual Xeon E5-2687W v3 @ 3.10GHz (20 CPU-cores – Hyper-Threading turned off) with 256 GB RAM and 2 TB PCI-SSD storage).
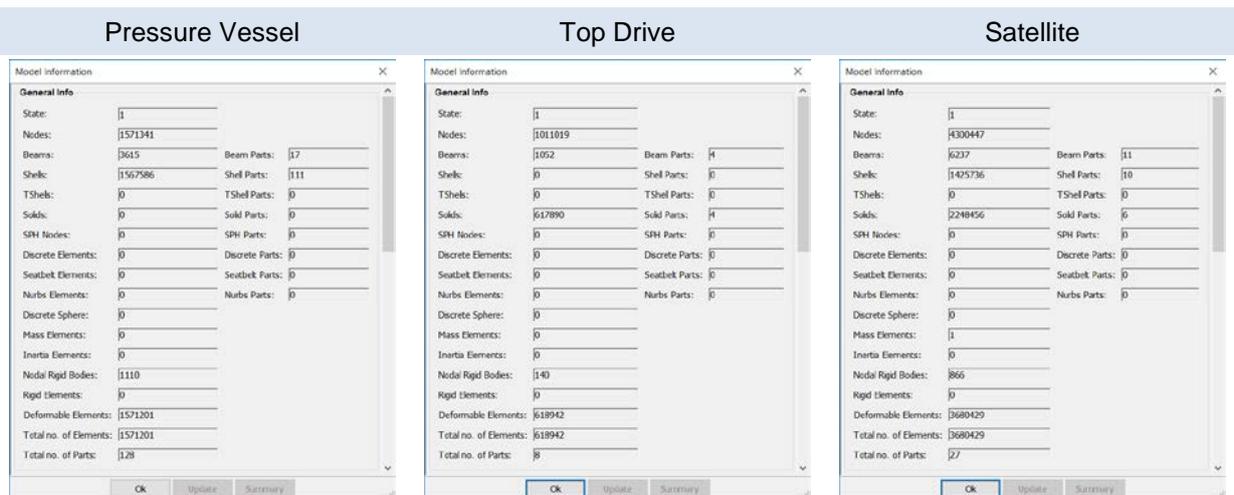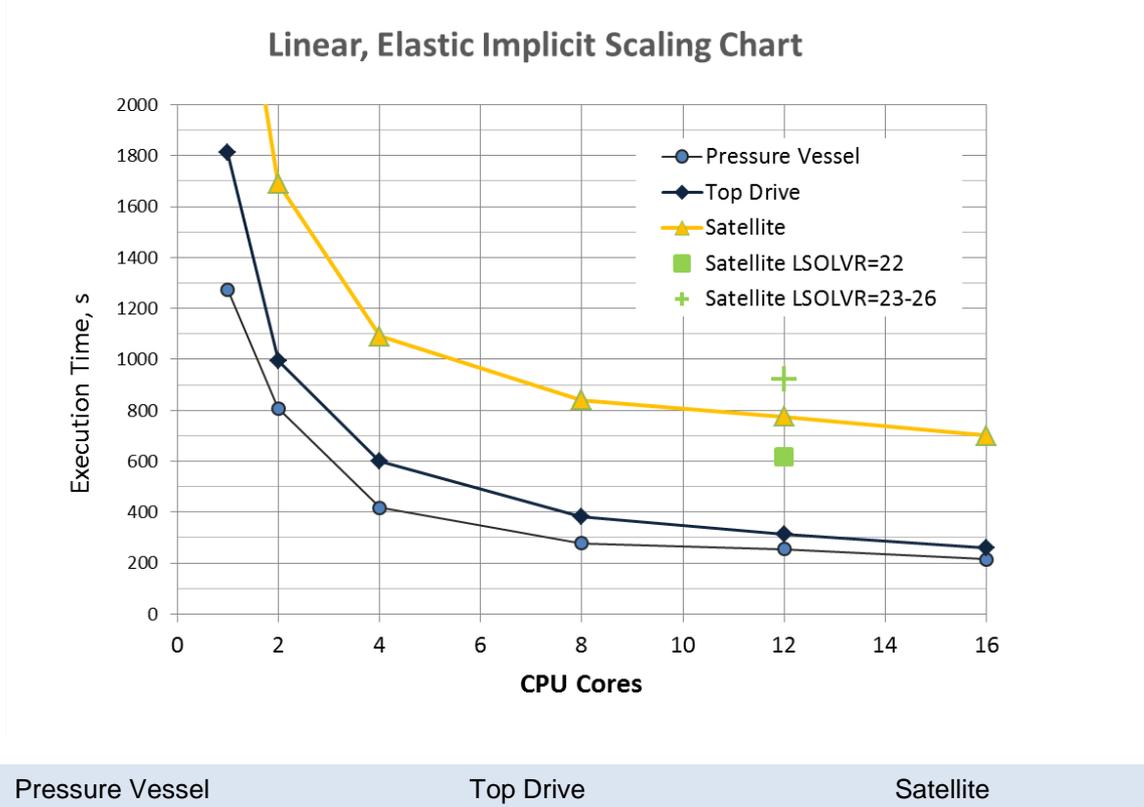
Fig. 12: Scaling chart for linear, elastic finite element models of commercial complexity

4 Nonlinear Implicit Analysis

Analysis examples are presented sequentially from basic plasticity to contact and then combined material plasticity with contact. The example models are very basic to illustrate the functionality of the code and use only the absolute minimal set of Keywords. At the end of this section scaling results are presented for more commercially relevant examples. In all work, the r9.1.0 MPP double-precision solver is used.

4.1   Material Nonlinearity: Plasticity / Rubber

The LS-DYNA implicit code strives to cover every material law in the Keyword Manual Vol. II and in general, the majority of material laws are covered. As a demonstration of the robustness of the implicit solver, Fig. 13 shows the deformation of a rubber bar to element failure and then complete release of the bar due to additional element failure. To allow the simulation to finish its solution sequence, it was run dynamically using *CONTROL_IMPLICIT_DYNAMICS. Otherwise the only other unique commands were: *CONTROL_ACCURACY, _IMPLICIT_AUTO and _IMPLICIT_SOLUTION. Details on the use of these Keywords are given in Section 4.3 since they are also used in the other nonlinear analysis sequences.
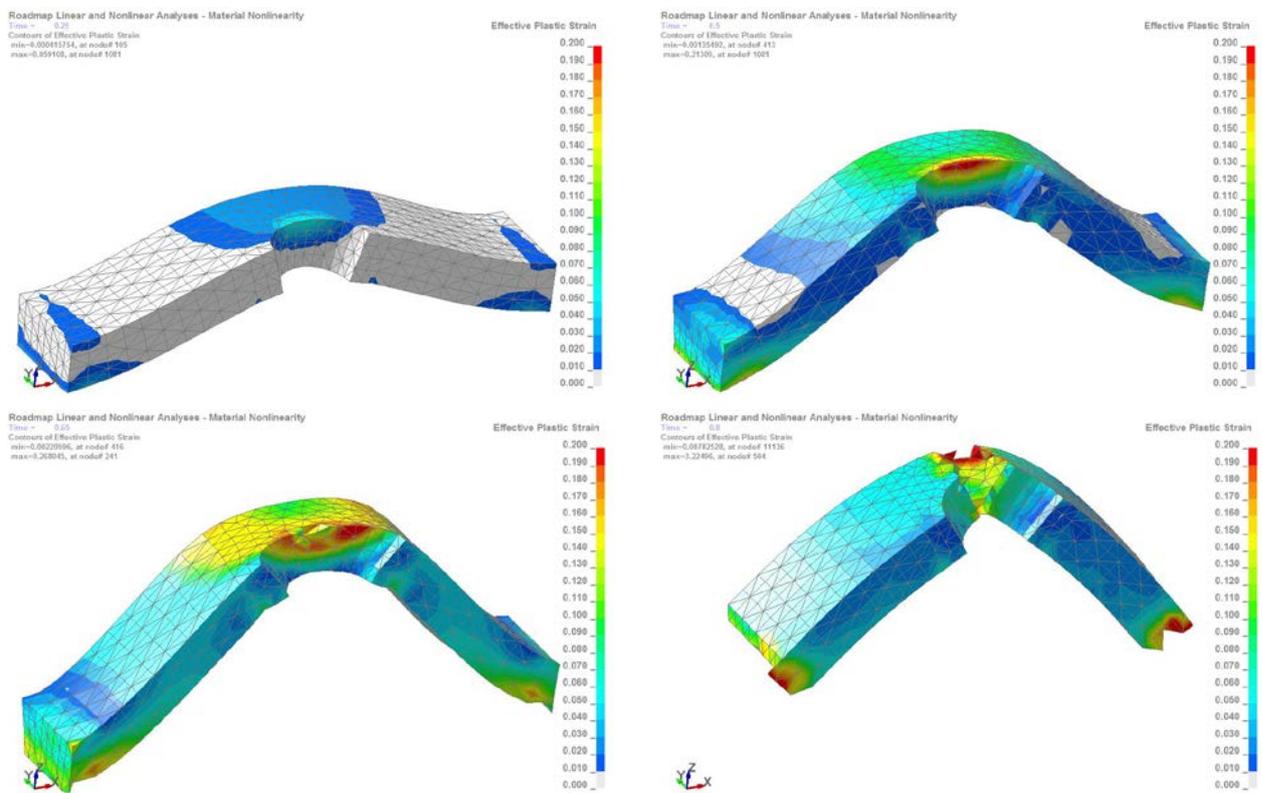
Fig. 13: Implicit analysis of rubber failure in half-symmetric four-point bend test

4.2   Geometric Nonlinearity: Contact

This is the classic crushing of a beer can done using LS-DYNA implicit. Fig. 14 presents a sequence of the crushing behavior. The can is supported on its base such that a Nastran linear buckling analysis can be performed. In LS-DYNA the can will start to buckle around a load factor of 0.19 while interestingly enough the Nastran linear buckling analysis indicates a load factor of 0.18. The material model is *MAT_ELASTIC and to handle the self-intersecting contact, the default settings were used for *CONTACT_AUTOMATIC_SINGLE_SURFACE_MORTAR [14, 15]. It should be emphasized that _MORTAR contact is very robust and rarely does one need to stray from the default settings. Although developed for implicit it is also applicable to explicit models and its higher computational cost (~15% [14]) is often an acceptable tradeoff given lower analysts' cost of model debugging.

As shown in Fig. 14, once the can starts to buckle, the applied force quickly crushes the can. What keeps the simulation stable is using *CONTROL_IMPLICIT_DYNAMICS. Although it adds some computational cost it tends to facilitate the convergence of simulations that normally would never run. Although one could also attempt to run this simulation without _DYNAMICS (aka, line search), the simplicity of using a quasi-static approach outweighs any computational cost.

On a buckling mechanics side, there was no need to simulate imperfections (e.g., *PERTURBATION) since just numerical noise and general rounding of nodal positions provides enough random initiation to start the buckling process.
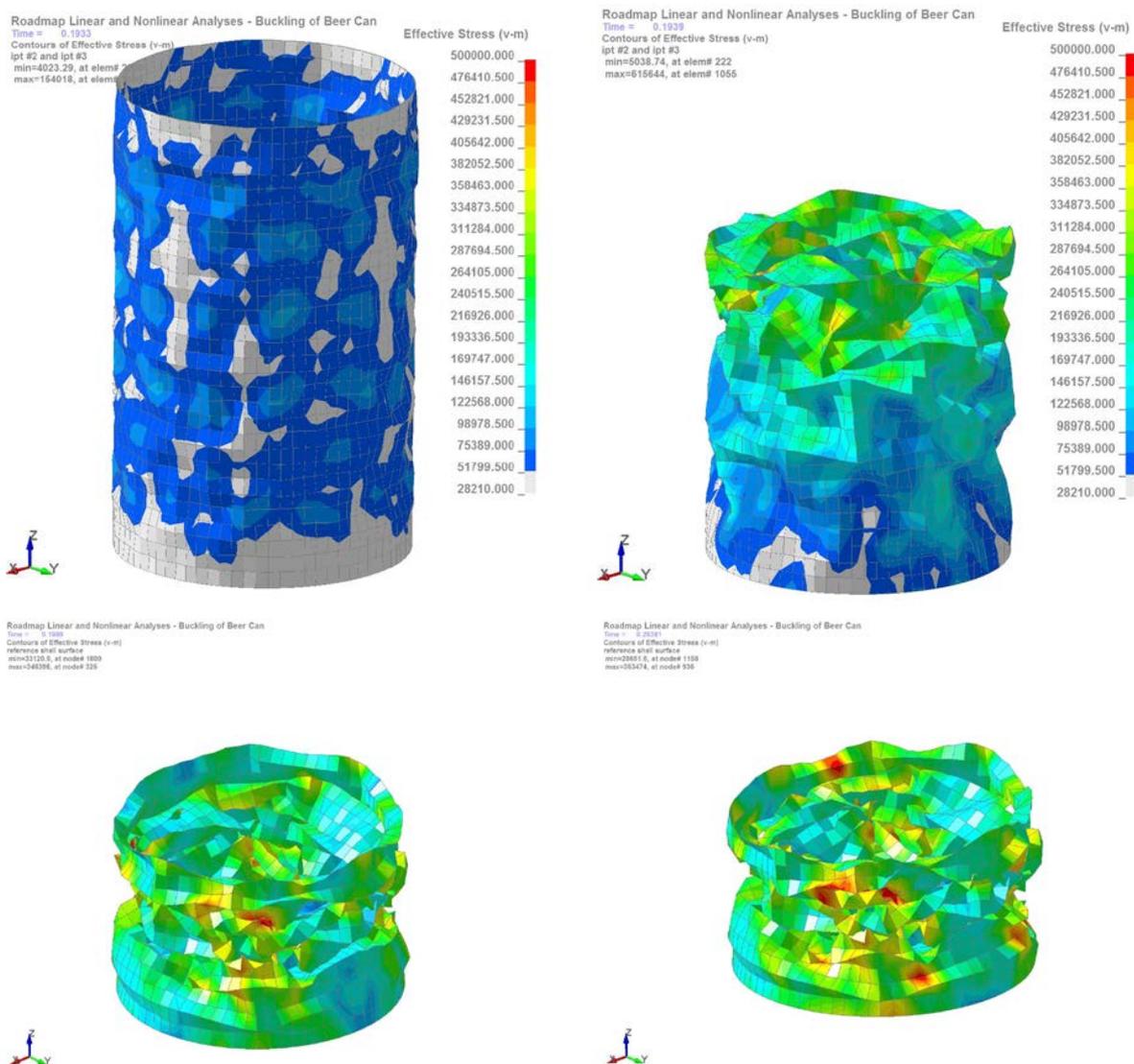
Fig. 14: LS-DYNA implicit analysis of the buckling of a thin walled cylinder (aka aluminum beer can)

4.3   General Combined Nonlinearity (Contact, Plasticity and Failure)

This example is presented as a compact problem that provides a platform to demonstrate the simplicity of setting up a full-on, nonlinear implicit LS-DYNA analysis. Fig. 15 shows the model and it final deformed shape. The analysis sequence has the bolts preload (Time=0.20) and then a pressure load ramped up along the top edge of the L-Beam. Since the pressure load follows the elements faces as the beam bends over, we have a "follower load". The analysis then covers preload, geometric nonlinearity, contact and material failure. It should be noted that the model uses _IMPLICIT_DYNAMICS to allow the part to completely fail and still converge. For the material model, the L-bracket uses *MAT_98 and for failure *MAT_ADD_EROSION is employed to set a failure criteria that differentiates between tensile and compressive failure modes (e.g., effeps=0.40 (global) while mxeps=0.20 (pure tensile)).

Bolt Pre-Load	Ramp Up of Pressure Load on Upper Edge

Material Failure as L-Beam Folds Over	L-Beam Snaps Through and Bounces Upward
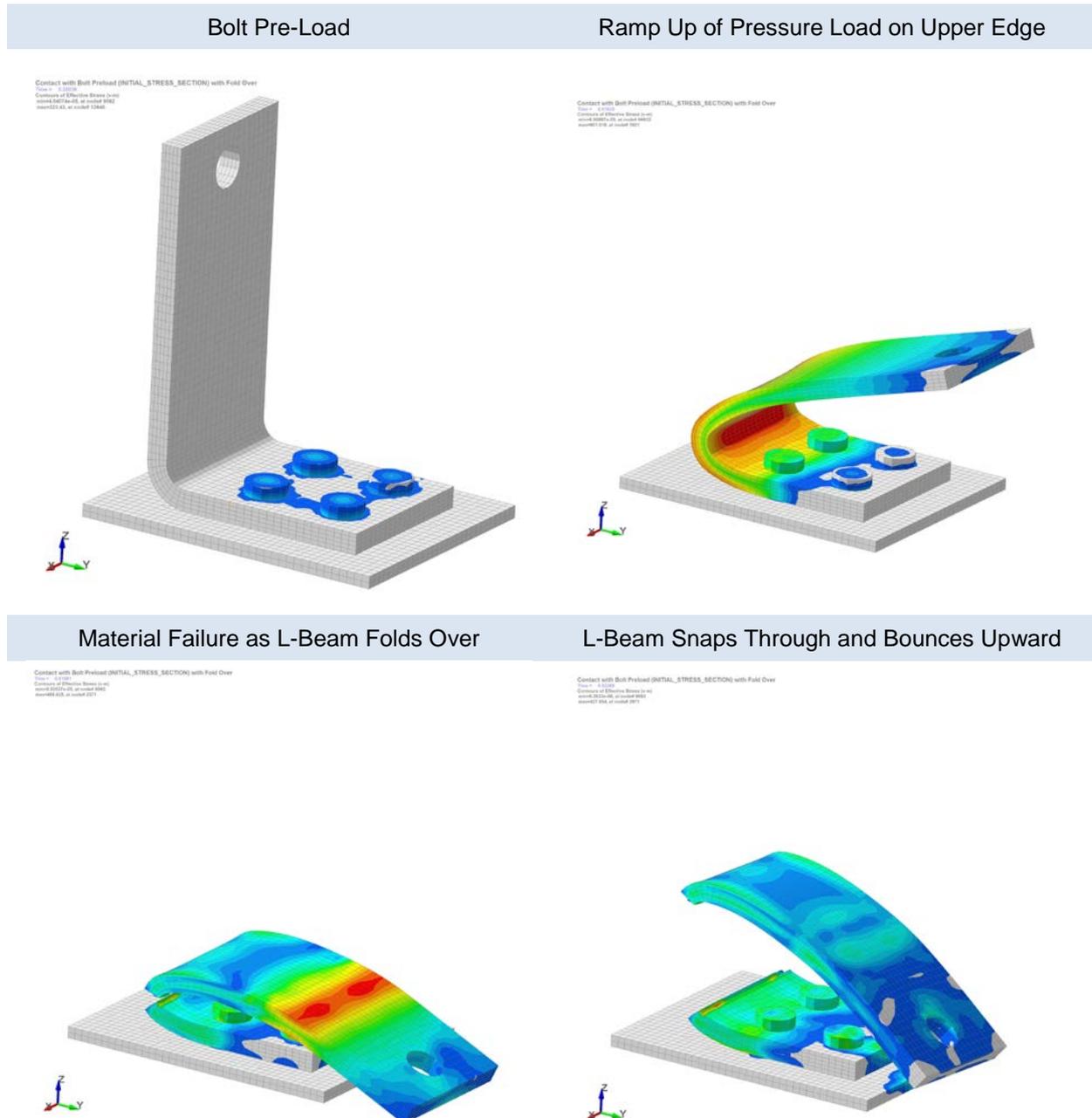
Fig. 15: Simple L-Bracket model for nonlinear analysis (courtesy of DYNAmore, Nordic)

4.4 Nonlinear Analysis LS-DYNA Keywords

The complete LS-DYNA keyword decks for these nonlinear models can be found at www.DYNAexamples.com.

4.4.1 Contact

*CONTACT_AUTOMATIC_SINGLE_SURFACE_MORTAR with ignore=1 or -1 {Mortar contact rarely benefits from non-default settings. The use of ignore handles small penetrations which often exists in the best of models. It is the authors' recommendation to strive to use only one contact definition for the complete model. Given the robustness and accuracy of _MORTAR to handle beam-on-beam, edge-on-surface, etc. this is often times entirely possible.}
{Recommended implicit practices for tied contacts is Appendix P: Implicit Solver within the LS-DYNA Keyword User's Manual Vol 1 (see R9.0.1 or later) and the reader is encouraged to read this section a few times since it has many gems of sound modeling advice.}

4.4.2 Control Section

*CONTROL_ACCURACY with osu=1 and iacc=1 {standard practice for implicit with the iacc providing specialized controls for implicit analysis.}
*CONTROL_IMPLICIT_DYNAMICS with imass=1 and gamma=0.6 and beta=0.38 {imass activates implicit dynamics while setting gamma=0.6 and beta=0.38 damps the dynamic solution toward a quasi-static behavior.}
*CONTROL_IMPLICIT_GENERAL with imflag=1 and dt0=? {implicit analysis and your initial time step is dependent on the severity of the model's nonlinearity or how lucky you are feeling}
*CONTROL_IMPLICIT_SOLUTION all defaults {Although one can run successfully with all defaults, one might consider performing a sensitivity check and run the model with abstol=1e-20. The authors have noted in the past that with the default convergence settings, the model may not be converged.} For a practical and useful discussion on solution convergence parameters, see Appendix P, LS-DYNA Keyword User's Manual, Volume 1.
*CONTROL_IMPLICIT_SOLVER with the lsolvr setting up for debate. As a background, the _IMPLICIT_SOLUTION nsolvr option sets the solution strategy from linear (nsolvr=1) to the default for nonlinear implicit (nsolvr=12) but the type of solver used to decompose the implicit matrix is controlled by the lsolvr setting. The default is lsolvr=5 for MPP and works well. Newer solvers are currently available from 22 through 26 and were investigated in this work. The results are mixed and no concrete recommendation can be made. It was observed that the transmission model (mostly solid elements) solved more effectively with lsolvr=22 while with the default solver it struggled to converge while in the case of the automotive seat (mostly shells), the new solvers failed and the default solver was required. Thus for the time being, the default solvers are recommended as the starting point but one should also explore lsolvr=22 for mostly solid element models.
*CONTROL_OUTPUT with solsig=2 {linear stress extrapolation from the solid element integration points}
*CONTROL_SHELL with isort=2 {Switches triangular elements from ELFORM 16 to ELFORM 17}

4.4.3 Database

*DATABASE_EXTENT_BINARY with neips=-3, beamip=9 and nintsld=8 {write out all shell, beam and solid element integration points}

4.4.4 Section

*SECTION_SOLID with elform=16 {standard 10-node tetrahedral}
*SECTION_SOLID with elform=-1 {Improved selectively reduced integration solid}
*SECTION_SHELL with elform=-16 and nip=7 {Better formulation and higher resolution of near surface stresses}
*SECTION_BEAM with elform=4 and qr/irid=4 {Belytschko-Schwer full cross-section integration with 3x3 Lobatto quadrature}

4.4.5 LS-PrePost Commands

Standard post-processing as one would for a general explicit analysis.

4.5   Nonlinear Scaling Results

The three examples given in Fig. 16 provide a broad overview of what can be solved using LS-DYNA's nonlinear implicit solver. The first model is of transmission composed of five ductile iron castings that are bolted together (bolt preload) and then shifted into first gear under maximum torque (engine at full 3,000 HP output). Internal to the model are a series of shafts (beam elements) and bearings (rings of solid elements that are in contact with the transmission housing). The second model shows progressive failure of a large composite shipping container under internal pressure. This model has been discussed in earlier work by Jensen and Laird [7]. The last example is from Satish and Borrvall [3] of an automotive seat. Scaling results and model information are given in Fig. 17.

Bolt Preload and Contact of Eight-Speed, Dual-Shaft 3,000 HP Transmission
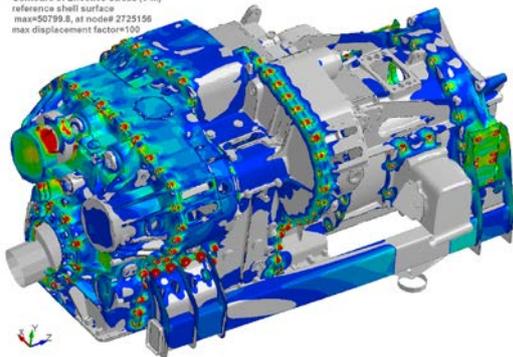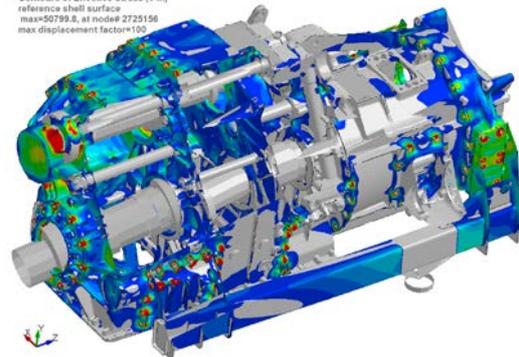
Aerospace Composite Transportation Container
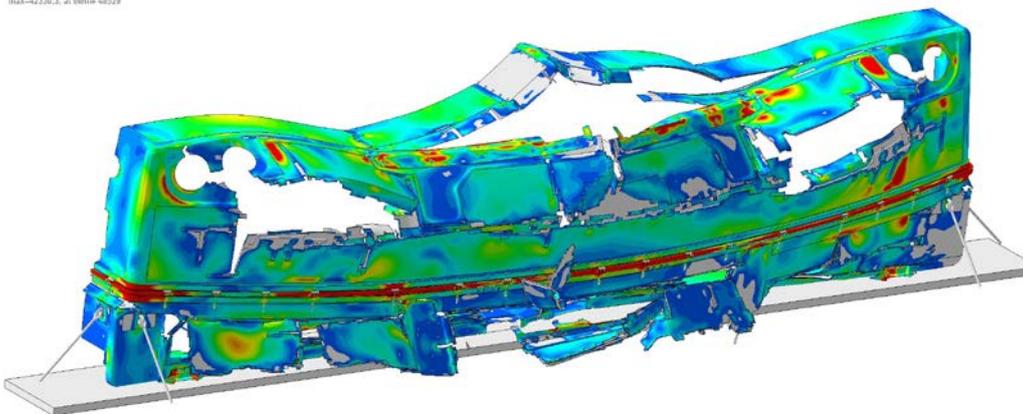
Automotive Seat Analysis
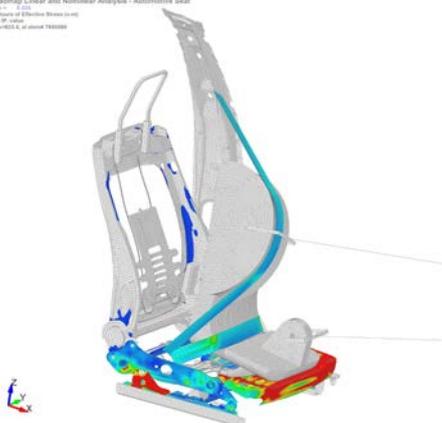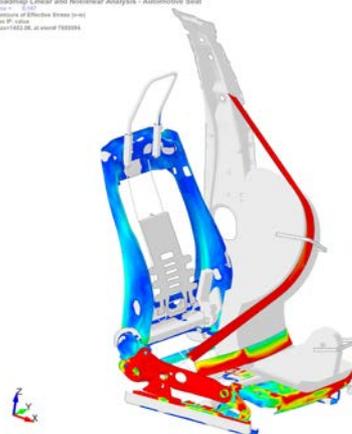
Fig. 16: Nonlinear scaling models of commercial complexity

Fig. 17 shows that for large-scale nonlinear implicit analyses, scaling is dependent upon many factors other than just node count. The first scaling graph on the upper, left-hand side shows execution time in minutes versus CPU-cores. The transmission and composite container models scale as might be expected while that for the automotive seat has negative scaling. Since the transmission model swamps the two other curves, another graph is given on the upper, right-hand side showing percentage of improvement between CPU-cores. This graph shows positive scaling improvement from two to sixteen CPU-cores for the composite container and more or less good scaling for the transmission up to twelve CPU-cores. As for the automotive seat model, scaling only occurs from two to four CPU-cores and then the scaling decreases and reverses. This was unexpected and the reason for such poor scaling is cause for future investigation.

As Fig. 17 indicates, scaling of large, nonlinear implicit models is not on par with what one would expect in an explicit analysis. Such implicit scaling performance represents an opportunity for LS-DYNA in the coming years.

3,000 HP Transmission	Composite Container	Automotive Seat
lsolvr=22	*lsolvr=5 (default)*	*lsolvr=5 (default)*

Fig. 17: Nonlinear implicit analysis scaling results for commercial simulation models

5 Linear and Nonlinear Implicit Observations

⇒ LS-DYNA linear, implicit is fully capable and generates standard and expected answers to linear mechanics problems for beams, shells and solid elements along with standard connection technologies (e.g., Nastran multi-point constraint elements (RBE's));

⇒ LS-DYNA shows good linear implicit scaling across a wide-range of commercial sized models up to 16 CPU-cores;

⇒ LS-DYNA nonlinear implicit analysis is robust and requires only a few specialized cards to activate and moreover, is directly compatible with the explicit solver facilitating implicit to explicit analysis sequences without the necessity of performing a restart;

⇒ LS-DYNA nonlinear implicit scales in most cases with large commercial sized models. A notable exception was observed for the automotive seat model and is an open topic for program development;

⇒ _MORTAR contact could well be the new default standard for all LS-DYNA analyses requiring contact regardless of whether it is implicit or explicit due its robustness, ease of setup and reasonable solution speed;

⇒ LSTC and DYNAmore have vastly improved their documentation, white papers and Guidelines for performing implicit analyses. With a bit of research, a wealth of information is available on-line to performing your first LS-DYNA implicit analysis.

6 Conclusion

This paper addresses two key questions about performing a LS-DYNA implicit analysis: (i) Can LS-DYNA accurately generate standard linear elastic implicit answers? And (ii) can LS-DYNA scale commercial sized models with linear and nonlinear analyses sequences? In the first case the answer is a simple *yes* while in the second case it is more complicated. Scaling is well noted for linear models but more variable when the analysis is nonlinear. This is known challenge to the LS-DYNA community and we expect to see improvements in the coming years.

The one take away comment about LS-DYNA implicit is that yesterday's experiences with the code should not be leveraged forward. The implicit code of today is different than what was available a few years ago and important changes are added almost monthly.

7 Acknowledgements

8 Literature

[1] Jonsson, A. and Lilja, M.: "Some Guidelines for Implicit Analyses Using LS-DYNA", DYNAmore Nordic, Göteborg, Sweden, Rev 10, 2017, 62 pages

[2] Lilja, M.: "Benchmark of LS-DYNA for Off-shore Applications according to DNV Recommended Practice C208", 13ᵗʰ International LS-DYNA Users Conference, 2014, 10 pages

[3] Pathy, S. and Borrvall, T.: "Quasi-Static Simulations using Implicit LS-DYNA", 14ᵗʰ International LS-DYNA Users Conference, 2016, 8 pages

[4] Hallquist, J.: "LS-DYNA Keyword User's Manual, Volume I – Appendix P: Implicit Solver", LSTC, Livermore, CA, USA, 2017, 2824 pages.

[5] Hirth, A., Gromer, A. and Borrvall, T.: "Dummy Positioning Using LS-DYNA Implicit", 10ᵗʰ European LS-DYNA Conference, Wurzburg, 2015, 42 pages

[6] Laird, G.: "Large Scale Normal Modes and PSD Analysis with Nastran and LS-DYNA", 12ᵗʰ International LS-DYNA Users Conference, 2012, 12 pages

[7] Jensen, A. and Laird, G: "Broad-Spectrum Stress and Vibration Analysis of Large Composite Container", 14ᵗʰ International LS-DYNA Users Conference, 2016, 23 pages

[8] Zienkiewicz, O.C. and Zhu, J.Z.: "The Superconvergent Patch Recovery and A Posteriori Error Estimates. Part I: The Recovery Technique", Intl. J. Numerical Methods in Engr.,Vol.33,1992, 33 pages

[9] BheemReddy, V., Jensen, A. and Laird, G.: "Composite Laminate Modeling", Predictive Engineering, 2015, 106 pages

[10] Wade B., Feraboli, P. and Rassaian, M.: "LS-DYNA MAT54 for Simulation Composite Crash Energy Absorption", JAMS, 2011, 25 pages

[11] Wade. B. and Feraboli, P.: "Composite Damage Material Modeling for Crash Simulation: MAT54 & the Efforts of the CMH-17 Numerical Round Robin", JAMS, 2014, 21 pages

[12] Day, J., Erhart, T. and Hartmann, S.: "LSTC Composite Notes for MAT54_MAT55", 2015, 7 pages

[13] Laird, G.: "LS-DYNA: Observations on Composite Modeling", FEA Information, Vol. 4, Issue 09, 2015, 5 pages

[14] Borrvall, T.: "Mortar Contact for Implicit Analysis", LS-DYNA Forum, DYNAmore, Ulm, Germany, 2012, 8 pages.

[15] Borrvall, T.: "A Guide to Solving Implicit Mortar Contact Problems in LS-DYNA", Technical Note DYNAmore Nordic AB, Linköping, Sweden Technical Note, 2104, 13 pages.